



Dokuments sagatavots Eiropas Sociālā fonda projekta „Profesionālajā izglītībā iesaistīto vispārizglītojošo mācību priekšmetu pedagogu kompetences paaugstināšana” (2009/0274/1DP/1.2.1.1.2/09/IPIA/VIAA/003) ietvaros. Dokumentu vai tā daļas nedrīkst reproducēt, pārrakstīt, tulkot kādā citā valodā, uzglabāt elektroniskā meklēšanas sistēmā vai pārraidīt, ja nav panākta rakstiska vienošanās ar Latvijas Universitāti. Visas tekstā izmantotās tirdzniecības zīmes pieder to īpašniekiem un ir izmantotas tikai kā atsauces.

© Latvijas Universitāte, 2011. Visas tiesības aizsargātas

## Satura rādītājs

1. Pārbaudes funkcijas teorētiskajiem jautājumiem.....	8
1.1. Funkcija IsMatch.....	8
1.2. Funkcija IsLengthAtLeast.....	10
1.3. Funkcija IsLength.....	11
1.4. Funkcija If.....	11
2. Pārbaudes funkcijas Word vai Writer vidē.....	12
2.1. Funkcija IsText.....	14
2.2. Funkcija IsTextExactly.....	15
2.3. Funkcija ContainsText.....	16
2.4. Funkcija IsOption.....	17
2.5. Funkcija IsFont.....	18
2.6. Funkcija IsFontSize.....	19
2.7. Funkcija IsTextFormatted.....	20
2.8. Funkcija IsFontColor.....	21
2.9. Funkcija IsShadingColor.....	22
2.10. Funkcija IsStyle.....	23
2.11. Funkcija IsParagraphFormatted.....	24
2.12. Funkcija IsDocumentFormatted.....	25
2.13. Funkcija IsTextInPage.....	26
2.14. Funkcija IsHeader.....	27
2.15. Funkcija IsFooter.....	27
2.16. Funkcija IsTable.....	28
2.17. Funkcija IsTableDimensions.....	28
2.18. Funkcija IsCellText.....	29
2.19. Funkcija IsColumnWidth.....	30
2.20. Funkcija IsRowHeight.....	30
2.21. Funkcija IsCellMargin.....	31
2.22. Funkcija IsCellBorder.....	31
2.23. Funkcija IsCellShading.....	32
2.24. Funkcija IsObject.....	33
2.25. Funkcija IsObjectSize.....	33
2.26. Funkcija IsMailMerge.....	34
2.27. Funkcija IsTextRepeated.....	35
3. Pārbaudes funkcijas Excel un Calc vidē.....	36

3.1.	Funkcija IsWorksheetOption .....	37
3.2.	Funkcija IsWorksheet.....	38
3.3.	Funkcija IsFormattedNumber .....	39
3.4.	Funkcija IsFormattedText .....	39
3.5.	Funkcija IsFormattedDate .....	39
3.6.	Funkcija IsFormattedCurrency.....	39
3.7.	Funkcija IsFormattedSpecial.....	39
3.8.	Funkcija IsFormattedAllignement.....	40
3.9.	Funkcija IsFormattedFont .....	41
3.10.	Funkcija IsFormattedBorder.....	42
3.11.	Funkcija IsFormattedPattern.....	43
3.12.	Funkcija IsFormattedProtected.....	43
3.13.	Funkcija IsRowHeight.....	44
3.14.	Funkcija IsColumnWidth .....	44
3.15.	Funkcija IsColumnWideEnough .....	45
3.16.	Funkcija IsRowHighEnough .....	45
3.17.	Funkcija IsCellNumber .....	46
3.18.	Funkcija IsCellText .....	47
3.19.	Funkcija IsCellDate .....	47
3.20.	Funkcija IsCellFormula .....	48
3.21.	Funkcija IsCellFormulaSubstring.....	49
3.22.	Funkcija IsWorksheetHeader .....	50
3.23.	Funkcija IsWorksheeFooter.....	50
3.24.	Funkcija IsCellFormulaEQ.....	51
3.25.	Funkcija IsCellEmpty .....	51
3.26.	Funkcija IsFrozen .....	52
3.27.	Funkcija IsRangeAscending .....	52
3.28.	Funkcija IsRangeDescending .....	53
3.29.	Funkcija IsFormulaValue .....	53
3.30.	Funkcija IsCellFormulaValue .....	54
3.31.	Funkcija IsChart .....	56
3.32.	Funkcija IsChartType .....	56
3.33.	Funkcija IsChartData.....	57
3.34.	Funkcija IsChartTitle.....	57
3.35.	Funkcija IsChartValueColor.....	58

4. Pārbaudes funkcijas PowerPoint un Impress vidē .....	59
4.1. Funkcija IsDesignTemplate .....	61
4.2. Funkcija IsOption .....	61
4.3. Funkcija IsPageSetup .....	62
4.4. Funkcija IsSlides .....	62
4.5. Funkcija IsSlideTitle .....	63
4.6. Funkcija IsNotSlideTitle .....	64
4.7. Funkcija IsSlideLayout .....	64
4.8. Funkcija IsSlideBackground .....	66
4.9. Funkcija IsSlideHidden .....	66
4.10. Funkcija IsNoteText .....	67
4.11. Funkcija IsSlideTransition.....	67
4.12. Funkcija IsSlidesNumbered.....	68
4.13. Funkcija IsFooter .....	68
4.14. Funkcija IsFooterDate .....	68
4.15. Funkcija IsElementPosition.....	69
4.16. Funkcija IsElementSize .....	70
4.17. Funkcija IsElementInFront.....	71
4.18. Funkcija IsElementAnimation.....	72
4.19. Funkcija ContainsTextBoxes.....	72
4.20. Funkcija IsTextBox .....	73
4.21. Funkcija ContainsText.....	73
4.22. Funkcija IsParagraphText.....	74
4.23. Funkcija IsParagraphFormatted.....	75
4.24. Funkcija IsParagraphLineSpacing.....	76
4.25. Funkcija IsTextFormatted.....	77
4.26. Funkcija IsFont .....	78
4.27. Funkcija IsFontSize .....	79
4.28. Funkcija IsFontColor .....	80
4.29. Funkcija ContainsTables .....	81
4.30. Funkcija IsTable .....	81
4.31. Funkcija IsTableExactly .....	82
4.32. Funkcija ContainsPictures .....	83
4.33. Funkcija IsPicture .....	83
4.34. Funkcija IsPictureSize .....	84

4.35.	Funkcija ContainsOrgCharts .....	85
4.36.	Funkcija IsOrgChart .....	85
4.37.	Funkcija ContainsCharts .....	86
4.38.	Funkcija IsChart .....	86
4.39.	Funkcija IsChartType .....	87
4.40.	Funkcija IsChartTitle .....	87
4.41.	Funkcija IsChartValueColor .....	88
4.42.	Funkcija IsChartData .....	89
5.	Pārbaudes funkcijas Access un Base vidē .....	90
5.1.	Funkcija IsTable .....	90
5.2.	Funkcija IsTableColumnCount .....	91
5.3.	Funkcija IsTableColumnType .....	92
5.4.	Funkcija IsTablePrimaryKey .....	93
5.5.	Funkcija IsRowCount .....	93
5.6.	Funkcija IsCellValue .....	94
5.7.	Funkcija IsRelationship .....	95
5.8.	Funkcija IsTableOrdered .....	95
5.9.	Funkcija IsColumnWidth .....	96
5.10.	Funkcija IsQuery .....	96
5.11.	Funkcija IsQueryColumnCount .....	96
5.12.	Funkcija IsQueryColumn .....	97
5.13.	Funkcija IsForm .....	97
5.14.	Funkcija IsFormFieldCount .....	99
5.15.	Funkcija IsFormField .....	100
5.16.	Funkcija IsFormHeaderFieldCount .....	101
5.17.	Funkcija IsFormHeaderField .....	101
5.18.	Funkcija IsFormFooterFieldCount .....	101
5.19.	Funkcija IsFormFooterField .....	101
5.20.	Funkcija Isreport .....	101
5.21.	Funkcija IsReportFieldCount .....	101
5.22.	Funkcija IsReportField .....	102
5.23.	Funkcija IsReportSortGroup .....	103
5.24.	Funkcija IsReportPage .....	104
5.25.	Funkcija IsReportHeaderFieldCount .....	104
5.26.	Funkcija IsReportHeaderField .....	104

5.27.	Funkcija IsReportFooterFieldCount .....	104
5.28.	Funkcija IsReportFooterField .....	104
6.	Pārbaudes funkcijas Windows un Linux vidē .....	105
6.1.	Funkcija IsFile .....	106
6.2.	Funkcija IsDirectory .....	106
6.3.	Funkcija IsFileReadOnly .....	107
6.4.	Funkcija IsFileModifiedAfter .....	107
6.5.	Funkcija IsFileSize .....	107
6.6.	Funkcija IsFileSizeExactly .....	108
6.7.	Funkcija IsFileType .....	109
6.8.	Funkcija IsFileText .....	109
6.9.	Funkcija IsMemorySize .....	109
6.10.	Funkcija IsDiskSize .....	110
6.11.	Funkcija IsProgrammInstalled .....	110
6.12.	Funkcija IsProgrammStarted .....	111
7.	InPārbaudes funkcijas Internet Explorer un Mozilla Firefox vidē .....	112
8.	Pārbaudes funkcijas Outlook un Thunderbird vidē .....	118
8.1.	Funkcija IsMessage .....	118

## 1. Pārbaudes funkcijas teorētiskajiem jautājumiem

Teorētiskie jautājumi parasti tiek pārbaudīti, salīdzinot studenta atbildi ar sistēmā ievadīto pareizo atbildi – pareizo atbildes variantu, kas studentam jāizvēlas, pareizo frāzi, kas jāievada, pareizo vietu attēlā, uz kuras jānoklikšķina.

Atsevišķos gadījumos jautājumus, uz kuriem students atbild ar frāzi vai brīvā tekstā, atbildes pārbaudei ir nepieciešami elastīgāki līdzekļi. Piemēram, uz jautāju: „Kā sauc tiesības izmantot programmatūru?”, pareizā atbilde ir „licence”, un visvienkāršāk ir to noformētu kā jautājumu ar viena varianta izvēle no dotām alternatīvām. Tomēr mēs varam jautājumu padarīt radošāku, nedodot izvēles variantus un noformējot jautājumu kā ar frāzi atbildamu. Tad studentam pašam ir jāieraksta vārds „licence”. Grūtības rada tas, ka students pareizo atbildi var ievadīt dažādos veidos, piemēram: „licence”, „Licence”, „programmatūras licence” vai „tā ir licence”. Šajā gadījumā atbildes pārbaudei nepieciešams lietot pārbaudes funkcijas.

Otrs gadījums, kad var rasties nepieciešamība lietot pārbaudes funkcijas ir jautājumi, uz kuriem jāatbild brīvā tekstā, piemēram: „Aprakstiet atšķirību starp aparatūru un programmatūru!” Students uz šādu jautājumu atbild uzrakstot savas domas, un šādu atbildi, protams, var pārbaudīt tikai pasniedzējs. Tomēr ar pārbaudes funkcijām ir iespējams atsijāt atbildes, kuras acīmredzami nav pareizas. Piemēram, atbilde nevar būt pareiza, ja tā ir īsāka par 30 simboliem, vai arī, ja tajā vispār neparādās vārdi „aparatūra” un „programmatūra”.

Pārbaudes funkcijas, kuras var lietot sarežģītāku teorētisku jautājumu pārbaudīšanai ir aprakstītas nākamajās nodaļās. Visu funkciju pirmais arguments ir studenta dotā atbilde uz jautājumu (ievadītā frāze vai ievadītais teksts). Sistēmā Tests to apzīmē ar &0 (nulltais parametrs). Ja tiek gatavots parametrizējams jautājums, funkcijā var lietot arī parametrus &1, &2... Šos parametrus parasti lieto funkciju otrajā argumentā.

Šajā un visās turpmākajās nodaļās funkcijas tiek pierakstītas, atdalot argumentus ar komatu, un teksta rindas liekot pēdiņās. Sistēmā Tests funkcijas ievada, izmantojot sistēmas izvēlnes, un ne komati, ne pēdiņas nav jālieto.

### 1.1. Funkcija *IsMatch*

Funkcija *IsMatch* pārbauda, vai norādītais teksts (kas parasti ir studenta atbilde), ir „aptuveni tāds”, kā vajadzīgs. „Aptuveni tāds” te nozīmē:

1. ka teksts ir tieši tāds, kā prasīts (burts burtā, tiesa, nešķirojot lielos un mazos burtus), vai arī
2. ka teksts atbilst norādītajai regulārajai izteiksmei.

Funkcijai *IsMatch* ir sekojoši argumenti:

1. pārbaudāmais teksts (parasti – studenta atbilde &0),
2. vajadzīgais teksts vai regulārā izteiksme.

Regulāra izteiksme ir spēcīgs līdzeklis tekstu aptuvenas sakritības pārbaudīšanai. Ar regulārām izteiksmēm var iepazīties grāmatā „Jeffrey E. F. Friedl, Mastering Regular Expressions” vai tiešsaistē:

[msdn2.microsoft.com/en-us/library/1400241x.aspx](http://msdn2.microsoft.com/en-us/library/1400241x.aspx)



Daži vienkārši regulāro izteiksmju piemēri

Nr.	Izteiksme	Komentārs
1.	MS Word	Precīzi uzrakstītais teksts (lielie un mazie burti gan netiek šķīroti)
2.	(MS Word) (Microsoft Word)	Vai nu pirmā frāze „MS Word”, vai arī otrā frāze „Microsoft Word”
3.	(MS) (Microsoft) (Word)	Tas pats, kas iepriekš
4.	.	Jebkurš viens simbols (burts, cipars, pieturas zīme, tuksne...)
5.	a*	Jebkurš skaits burtu „a” pēc kārtas
6.	.*	Jebkurš skaits jebkādu simbolu
7.	.*PowerPoint.*	Vārds „PowerPoint” pirms kura un pēc kura var būt jebkurš skaits jebkādu simbolu
8.	\d	Jebkurš cipars
9.	\d*	Jebkurš skaits jebkādu ciparu
10.	0{6}	Tieši 6 nulles
11.	\d{6}	Tieši 6 cipari (jebkādi)
12.	\d{6}-\d{5}	Tieši 6 cipari, tad mīnus zīme, un vēl 5 cipari
13.	\.	Tieši viens punkts („.” nozīmē jebkuru simbolu)
14.	.*\.{3}	Jebkurš teksts, kas beidzas ar trim punktiem

Funkcijai IsMatch lietošanas piemēri (pēdiņas piemēros lietotas parametru atdalīšanai – sistēmas Tests logos tās nav jāievada)

Jautājums	Funkcijas lietojums	Komentārs
Cik baitu ir kilobaitā?	IsMatch(„&0”, „1024”)	Atbilde var būt tikai „1024” bez jebkādām variācijām. Šādu jautājumu vienkāršāk var pārbaudīt bez formulas – norādot pareizo atbildes variantu tieši
Kā sauc tiesības izmantot programmatūru?	IsMatch(„&0”, „.*licence.*”)	Jebkurš teksts, kurā kaut kur pa vidu ir vārds „licence”. Der, piemēram, „Programmatūras licence”, „Licences”, „Tās ir licences”, „licences!”. Tiesa, der arī „Tās noteikti nav licences”, bet praksē vērā ņemamu seku šādai neprecizitātei nav, kamēr

		neprecizitāte ir par labu studentam
Aprakstiet atšķirību starp aparatūru un programmatūru	IsMatch(„&0”, „(*aparatūr.*programmatūr.*) (*programmatūr.*aparatūr.*)”)	Jebkurš teksts, kurā kaut reizi pieminēta aparatūra un programmatūra – vienalga kādā secībā. Vārdi var būt jebkurā locījumā, jo galotnes nav norādītas
Sameklējiet Google sistēmā jebkuru informāciju par &1 un iekopējiet to atbildes logā	IsMatch(„&0”, „*&1.*”)	Šis ir jautājums ar parametru. Parametra vērtības varētu, piemēram, būt: „ECDL”, „FTP” un „ISDN” (students jautājumā redz vienu no šiem vārdiem, bet dažādiem studentiem var parādīties dažādi vārdi). Par pareizu tiek uzskatīta jebkura atbilde, kurā meklējamais vārds (apzīmēts ar &1) vismaz reiz ir pieminēts (tieši tas vārds, kuru students redzēja jautājumā – funkcija „pielāgojas” jautājumam)

## 1.2. Funkcija *IsLengthAtLeast*

Funkcija *IsLengthAtLeast* pārbauda, vai norādītais teksts (kas parasti ir studenta atbilde), ir vismaz tik garš, kā norādīts. Tiek skaitīti visi simboli, ieskaitot pieturas zīmes un tuksnes starp vārdiem.

Funkcijai *IsLengthAtLeast* ir sekojoši argumenti:

1. pārbaudāmais teksts (parasti – studenta atbilde &0),
2. minimālais garums.

Funkcijai *IsLengthAtLeast* lietošanas piemērs

Jautājums	Funkcijas lietojums	Komentārs
Aprakstiet atšķirību starp aparatūru (hardware) un programmatūru (software)	IsLengthAtLeast(„&0”, 30)	Mēs nevaram atbildi automātiski pārbaudīt, bet varam droši teikt, ka atbilde, kas īsāka par 30 simboliem, nav pareiza

### 1.3. Funkcija IsLength

Funkcija IsLength pārbauda, vai norādītais teksts (kas parasti ir studenta atbilde), ir tieši tik garš, kā norādīts – ne garāks, un ne īsāks. Tiek skaitīti visi simboli, ieskaitot pieturas zīmes un tuksnes starp vārdiem.

Funkcijai IsLength ir sekojoši argumenti:

1. pārbaudāmais teksts (parasti – studenta atbilde &0),
2. sagaidāmais garums.

Funkcijai IsLength lietošanas piemērs

Jautājums	Funkcijas lietojums	Komentārs
Enciklopēdijā lv.wikipedia.org sameklējiet jēdziena „dators” definīciju un iekopējiet to atbildes laukā	IsLengthAtLeast(„&0”, 553)	Mēs varējām salīdzināt arī tekstu (ar funkciju IsMatch), bet praktiskiem mērķiem pilnīgi pietiek, ja salīdzina garumu. Tā kā interneta saturs mainās, ir grūti panākt, lai šāda vaida jautājums būtu ilglaicīgs

### 1.4. Funkcija If

Funkcija If paredzēta saliktu pārbaudes formulu veidošanai.

Funkcijai If ir sekojoši argumenti:

1. teksts1 (parasti jautājuma parametra apzīmējums &1),
2. teksts2 (parasti konkrēta parametra vērtība),
3. teksts3 (parasti studenta atbildes apzīmējums &0),
4. teksts4 (parasti konkrēts atbildes variants).

Funkcija veic pārbaudi: „Ja teksts1 vienāds ar teksts2, tad arī teksts3 jābūt vienādam ar teksts4”. Tā atbild „Jā” divos gadījumos: 1) ja teksts1=teksts2 un teksts3=teksts4 un 2) teksts1 nav vienāds ar teksts2 (tad teksts3 un teksts4 nav svarīgi).

Jautājums	Funkcijas lietojums
Jautājums ar parametru aun atbildes variantiem: „Kura no dotajām programmām paredzēta, lai &1?” Parametra &1 vērtības ir: „sagatavotu tekstuālus dokumentus”, „veidotu datu bāzes”, „sūtītu e-pasta ziņojumus”. Atbildes varianti ir: „Outlook”, „Access”, „Word”.	If(&1, „sagatavotu tekstuālus dokumentus”, &0, „Word”) & If(&1, „veidotu datu bāzes”, &0, „Access”) & If(&1, „sūtītu e-pasta ziņojumus”, &0, „Outlook”) &

## 2. Pārbaudes funkcijas Word vai Writer vidē

Praktiskie uzdevumi teksta redaktora Word vai Writer vidē ir tādi, kas liek studentam kaut ko izdarīt ar dokumentu (piemēram, noformēt vārdu „ECDL” kursīvā). Pēc tam, kad students ir beidzis uzdevuma pildīšanu, dokuments tiek saglabāts. Pārbaudīt uzdevuma pareizību nozīmē pārbaudīt, vai saglabātais dokuments ir tāds, kā prasa uzdevums (piemēram, vai vārds „ECDL” tik tiešām ir kursīvā). Šo pārbaudi realizē ar pārbaudes funkcijām.

Vairumā gadījumu pārbaudes funkcijas Microsoft Word dokumentiem ir tādas pat, kā Open Office Writer dokumentiem, tāpēc ir iespējams veidot uzdevumus, kurus var vienlīdz labi izpildīt kā Word tā Writer vidēs. Nelielās atšķirības ir atrunātas attiecīgo funkciju aprakstos, tomēr no šīm atšķirībām ir ieteicams izvairīties.

Pārbaude, ko realizē funkcija, parasti neattiecas uz visu dokumentu, tā attiecas uz kādu dokumenta fragmentu (piemēram, uz vienu vārdu). Vairumam funkciju pirmie divi argumenti nosaka dokumenta fragmentu, uz kuru funkcija attiecas.

Pirmais arguments nosaka fragmenta veidu. Iespējamās vērtības ir:

1. „Document” – viss dokumenta teksts,
2. „Paragraph” – paragrāfs (rindkopa),
3. „Word” – vārds MS Word (nevis Writer) izpratnē. Tā kā Word un Writer vārda interpretācija nedaudz atšķiras, uzdevumu sagatavotājiem vajadzētu izvairīties no situācijām, kad šai atšķirībai ir nozīme uzdevuma pareizības pārbaudē. To viegli izdarīt, ar taustiņu kombināciju Ctrl-Right saskaitot vārdus kā Word tā Writer vidēs.
4. „Text” – jebkurš teksta fragments (varbūt vārda daļa, varbūt vairāki vārdi).

Otrais arguments precizē, par kuru paragrāfu, vārdu vai teksta fragmentu ir runa. Šo argumentu sauc par meklēšanas kritēriju. Meklēšanas kritērijs var būt:

1. skaitlis, kas norāda, kurš paragrāfs vai vārds ir domāts (paragrāfus un vārdus numurē no dokumenta sākuma), vai
2. teksts, kas norāda to paragrāfu vai vārdu, kas satur norādīto tekstu. Ar šo meklēšanas kritēriju identificē arī „Text” tipa fragmentus. Teksta meklēšana notiek no dokumenta sākuma, un no vairākiem vienādiem fragmentiem tiek atrasts pirmais. Meklējot lielle un mazie burti netiek ņemti vērā.

Sekojoši piemēri ilustrē vajadzīgu fragmentu meklēšanu. Tajos tiek izmantota funkcija IsFont, kura pārbauda, vai teksta fragments ir noformēts vajadzīgajā fontā. Šeit un turpmāk piemēri izmanto sekojošu Word dokumentu.

Labdien, Kalniņa kungs!

Priecājamies pavēstīt, ka Jūs esat ar labām sekmēm nokārtojis pēdējo, septīto ECDL testu, un Jums piešķirts Eiropas Datorprasmes sertifikāts. Sertifikātu Jūs varat saņemt Latvijas Universitātes Datorikas fakultātē, Raiņa bulvārī 19.

### Dokumenta fragmenta norādīšanas piemēri

Nr.	Funkcija	Komentārs
1.	IsFont(„Paragraph”, 3, „Arial”)	Vai dokumenta 3. paragrāfs ir fontā Arial? Par citiem paragrāfiem nekas netiek prasīts. Jāņem vērā, ka paragrāfi tiek skaitīti pēc Word likumiem, un tukša rindiņa arī ir paragrāfs. Mūsu piemērā šī pārbaude attiecas uz paragrāfu, kas sākas ar vārdiem „Priecājamies pavēstīt”, jo otrais paragrāfs ir tukša rindiņa.
2.	IsFont(„Documents”, „Arial”)	Vai viss dokuments ir fontā Arial? Otrais arguments nav norādīts (divi komati pēc kārtas), jo meklēšanas kritērijs nav jānorāda
3.	IsFont(„Paragraph”, „ECDL”, „Arial”)	Vai tas paragrāfs, kurā ir teksts „ECDL”, ir fontā Arial? Vārds „ECDL” tiek izmantots tikai paragrāfa identifikācijai, fontā Arial ir jābūt visam paragrāfam, ne tikai šim vārdam
4.	IsFont(„Word”, 4, „Arial”)	Vai dokuments 4. vārds ir fontā Arial? Piemērā tas ir vārds „kung” (to var pārbaudīt, četras reizes nospiežot Ctrl-Right).
5.	IsFont(„Word”, „kung”, „Arial”)	Vai pirmais vārds dokumentā, kurā ir burti „kung” ir fontā Arial? Šādu konstrukciju var lietot, lai izvairītos no vārdu locīšanas problēmas
6.	IsFont(„Text”, „Eiropas Datorprasmes sertifikāts”, „Arial”)	Vai teksts „Eiropas Datorprasmes sertifikāts” ir fontā Arial?

„Document” tipa fragmentiem meklēšanas kritēriju pielietojot nav jēgas – tas vienmēr nozīmē visu dokumentu. Tāpat nav jēgas fragmenta tipam „Text” meklēšanas kritēriju norādīt kā skaitli.

Ja funkcija nevar dokumentā atrast norādīto paragrāfu vai vārdu, tad pārbaudes rezultāts vienmēr ir negatīvs (atbilde ir: „Nē”). Tā piemēram, funkcija IsFont(„Paragraph”, 99, „Arial”) atbildēs „Nē”, ja dokumentā (studenta darba rezultātā) nemaz nebūs deviņdesmit deviņu paragrāfu. Tāpat arī funkcija IsFont(„Text”, „ICDL”, „Arial”) atbildēs „Nē”, ja dokumentā nebūs teksta „ICDL”.

Pārbaudot, vai fragmentam ir kāda īpašība, vienmēr tiek pārbaudīts viss fragments, katrs tā simbols, tomēr fragmentam apkārt esošās tuksnes tajā ieskaitītas netiek, tāpat arī paragrāfa beigās esošā neredzamā paragrāfa zīme netiek ieskaitīta paragrāfā. Šāda pārbaudes funkciju definīcija nodrošina, ka šaubas tiek tulkotas par labu studentam – atbilde būs pareiza arī tad, ja tuksnes un paragrāfu zīmes nebūs noformētas tā, kā prasīts.

Turpmākajās nodaļās dots atsevišķu funkciju apraksts. Apraksts nav veidots kā rokasgrāmata – tas ir lasāms no sākuma līdz beigām, jo informācija, kas attiecas uz daudzām funkcijām, ir dota tikai pirmās funkcijas parakstā, un varbūt pat piemēros.

## 2.1. Funkcija IsText

Funkcija IsText pārbauda, vai norādītais dokumenta fragments satur norādīto tekstu un tikai to.

Funkcijai IsText ir sekojoši argumenti:

1. fragmenta veids,
2. meklēšanas kritērijs,
3. teksts.

Teksta fragmentu salīdzinot ar trešo argumentu, lielie un mazie burti netiek atšķirti (nav „case sensitive”). Tiek prasīta tikai teksta sakrišana – formatējuma zīmes dokumentā (sk. Edit->Find->Special) netiek ņemtas vērā.

### Funkcijai IsText lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsText(„Paragraph”, 1, „Labdien, Kalniņa kungs!”)	Vai dokumenta 1. paragrāfs ir „Labdien, Kalniņa kungs!”?
2.	IsText(„Paragraph”, 1, „Labdien, &1 kungs!”)	Šāda konstrukcija būtu lietojama praktiskā uzdevumā, kas liek studentam nomainīt vēstules adresātu. Lai uzdevumi būtu daudzveidīgāki, adresāts tiek noformēts kā uzdevuma parametrs &1. Parametra vērtības varētu būt „Lejiņa kungs”, „Zariņa kungs”, „Ozoliņa kungs”. Uzdevuma formulējums tad ir: „Nomainiet vēstules adresātu uz &1”, un parametrs &1 katrā konkrētā uzdevumā tiek aizstāts ar konkrētu vērtību kā uzdevuma formulējumā, tā pārbaudes formulā.
3.	IsText(„Paragraph”, „Labdien”, „Labdien, Kalniņa kungs!”)	Vai tas paragrāfs, kurā ir teksts „Labdien”, ir „Labdien, Kalniņa kungs!”?
4.	IsText(„Word”, 4, „kungs”)	Vai 4. vārds dokumentā ir „kungs”?
5.	IsText(„Text”, „kungs!”, „kungs!”)	Formāli atļauts, bet bezjēdzīgs funkcijas lietojums. Rezultāts ir „Nē” vienīgi tad, ja dokumentā nemaz nav teksta „kungs!”, tomēr teksta esamību dokumentā vienkāršāk ir pārbaudīt ar funkciju ContainsText

## 2.2. Funkcija *IsTextExactly*

Funkcija *IsTextExactly* pārbauda, vai norādītais dokumenta fragments satur norādīto tekstu un tikai to. Funkcija darbojas tāpat, kā *IsText* (sk. 2.1), tikai prasa precīzu teksta sakrišanu – salīdzinot trešo argumentu ar dokumenta fragmentu, lielie un mazie burti tiek uzskatīti par atšķirīgiem, un formatējuma zīmes ignorētas netiek. Funkcijas trešajā argumentā līdz ar tekstu var norādīt arī formatējuma zīmes `^t` un `^p` (tabulācija un rindkopas beigas). Šāda precīzāka salīdzināšana attiecas tikai uz trešā argumenta salīdzināšanu ar dokumenta fragmentu. Dokumenta fragmenta meklēšanas likumi (pirmie divi argumenti) visām funkcijām ir vienādi – meklēšana nav „case sensitive”, un formatējums tiek ignorēts.

### Funkcijai *IsTextExactly* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsTextExactly(„Paragraph”, 1, „Labdien, Kalniņa kungs!”)</code>	Vai dokumenta 1. paragrāfs ir „Labdien, Kalniņa kungs!”? Atbilde būs „Nē”, ja, piemēram, vārds „Kalniņa kungs” būs rakstīts ar mazo burtu, vai arī, ja starp vārdiem „Labdien,” un „Kalniņa kungs!” būs nevis viena tuksne, bet divas.

### 2.3. Funkcija **ContainsText**

Funkcija **ContainsText** pārbauda, vai norādītais dokumenta fragments satur norādīto tekstu. Bez norādītā teksta dokumenta fragments drīkst saturēt arī citu tekstu.

Funkcijai **ContainsText** ir sekojoši argumenti:

1. fragmenta veids,
2. meklēšanas kritērijs,
3. teksts.

Teksta fragmentu salīdzinot ar trešo argumentu, lielie un mazie burti netiek atšķirti (nav „case sensitive”). Tiek prasīta tikai teksta sakrišana – formatējuma zīmes dokumentā (sk. Edit->Find->Special) netiek ņemtas vērā.

#### Funkcijai **ContainsText** lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>ContainsText(„Document” , „ECDL”)</code>	Vai dokumentā jebkurā vietā ir burti „ECDL”?
2.	<code>!ContainsText(„Document” , „ ”)</code>	Funkcijas trešais arguments ir divas tuksnes. Šādu funkciju ar nolieguma zīmi var lietot, lai pārbaudītu dokumenta sagatavotāja stilu – divas tuksnes pēc kārtas parasti norāda uz rakstītāja paviršību.
3.	<code>ContainsText(„Paragraph” , „!” , „Labdien”)</code>	Vai tajā paragrāfā, kurā ir izsaukuma zīme, ir arī vārds „Labdien”? Jāatzīmē, ka meklēšana pēc konteksta vienmēr atrod pirmo sakritību, skatoties no dokumenta sākuma. Mūsu piemērā izsaukuma zīme ir arī paragrāfā „Apsveicam!”, tomēr šis paragrāfs atrasts netiks.
4.	<code>!ContainsText(„Document” , „Apsveicam!”)</code>	Praktiskais uzdevums varētu būt „Izdzēst no dokumenta rindkopu „Apsveicam!”. Formula ar negāciju atbildēs „Jā”, ja šāda teksta dokumentā nebūs.



## 2.4. Funkcija IsOption

Funkcija IsOption pārbauda, vai dokuments ir saglabāts ar norādītajām opcijām.

Funkcijai IsOption ir sekojoši argumenti:

1. opcijas nosaukums,
2. vērtība.

Opcijas nosaukums var būt vai nu „View”, vai „Zoom”. Ja opcijas nosaukums ir „View”, tad vērtības var būt „Normal”, „Web layout”, „Print layout” utt. (sk. izvēlni View). Ja opcijas nosaukums ir „Zoom”, tad vērtības var būt skaitļi (piemēram, 50 nozīmē Zoom 50%). Lai izvairītos no kļūdām, kas saistītas ar izmēru mērīšanu centimetros un collās, kā arī ar dažādām ekrānu izšķirtspējām, Zoom skaitliskajai vērtībai tiek pieļauta +/-5% liela kļūda.

### Funkcijai IsOption lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsOption(„View”, „Print layout”)	Vai dokumentu saglabājot ir bijusi ieslēgta opcija View Print layout?
2.	IsOption(„Zoom”, 10)	Vai dokumentu saglabājot ir bijusi ieslēgta opcija Zoom 10%, lai pārliecinātos par pareizu dokumenta sadalījumu pa lapām?
3.	IsOption(„Zoom”, &1)	Līdzīgs iepriekšējam piemēram, tikai ar parametru. Uzdevums varētu būt formulēts: „Ieslēdziet opciju Zoom &1% un saglabājiet dokumentu”, un parametra &1 vērtības varētu būt 100, 125, 150.

Bez „View” un „Zoom” funkcija var pārbaudīt vēl dažas opcijas, kas attiecas uz visus sistēmu, nevis vienu dokumentu, tomēr šo opciju vērtības netiek glabātas dokumentā, un tāpēc attiecīgās pārbaudes darbojas tikai uzdevuma izpildes brīdī – pēc tam rezultātu pārbaudīt nav iespējams. Tādas opcijas var izmantot treniņuzdevumos, bet ne testos, kur rezultāta pārbaude pēc testa pabeigšanas ir obligāta. Papildus opcijas ir: Options.View.FormattingMarks (ar vērtībām Tab characters, Spaces... All – sk. izvēlni Tools/Options); Options.UserInformation.UserName; Options.UserInformation.Initials; Options.FileLocation.Documents.

Lai testā pārliecinātos, vai students prot lietot šīs sistēmas līmeņa opcijas, uzdevumi ir jāformulē netieši un jāizmanto teorētiskie jautājumi. Piemēram, lai pārliecinātos, vai students prot lietot opciju Tools – Options – Show formating marks, var veidot ar frāzi atbildamu jautājumu: „Cik tabulācijas zīmju ir pirms vārda ”Apsveicam””. Ja dokumentā pirms šī vārda ieliek trīs tuksnes un vienu tabulācijas zīmi, tad, nelietojot opciju Show formating marks, uz šo jautājumu atbildēt ir grūti.

Var jautāt arī par šo opciju vērtībām, piemēram, „Kāda ir Word noklusētā mērvienība?” (ar atbildes variantiem „Centimetri” un „Collas”).

Pārbaudot zināšanas par opcijām, jāņem vērā arī iespēja, ka students var nejauši uzstādīt tādas opcijas, kas padara tālāko testa uzdevumu pildīšanu neiespējamu. No šī viedokļa drošāk ir zināšanas par opcijām pārbaudīt ar teorētiskiem jautājumiem.

## 2.5. Funkcija IsFont

Funkcija IsFont pārbauda, vai norādītais dokumenta fragments ir noformēts norādītajā fontā.

Funkcijai IsFont ir sekojoši argumenti:

1. fragmenta veids,
2. meklēšanas kritērijs,
3. fonta nosaukums.

Par burtu lielumu nekas netiek prasīts. Tāpat arī nekas netiek prasīts par fontu lietojumu pārējā dokumentā (tas, ka funkcija neko nepārbauda ārpus norādītā fragmenta, attiecas uz visām funkcijām).

### Funkcijai IsFont lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsFont(„Document”, „Arial”)	Vai viss dokuments ir fontā Arial?
2.	IsFont(„Paragraph”, 3, „Times New Roman”) & !IsFont(„Paragraph”, 1, „Arial”) & !IsFont(„Paragraph”, 2, „Arial”) & !IsFont(„Paragraph”, 4, „Arial”) & !IsFont(„Paragraph”, 5, „Arial”)	Ja uzdevums ir bijis nomainīt trešajam paragrāfam fontu uz Arial, ir dabiski pārbaudīt, vai pārējiem paragrāfiem fonts nav mainījies. Saskaņā ar principu „šaubas – par labu studentam”, tas, kādā fontā noformēta paragrāfa zīme aiz 3 paragrāfa, netiek pārbaudīts (tāpat netiek pārbaudīts arī pārējo paragrāfa zīmju noformējums).

## 2.6. Funkcija *IsFontSize*

Funkcija *IsFontSize* pārbauda, vai norādītais dokumenta fragments ir noformēts ar norādītā lieluma fontu.

Funkcijai *IsFontSize* ir sekojoši argumenti:

1. fragmenta veids,
2. meklēšanas kritērijs,
3. fonta izmērs.

Par fonta nosaukumu nekas netiek prasīts. Fonta izmēru norāda kā skaitli. Var norādīt arī zīmes aiz komata. Fonta izmērā tiek pieļauta +/-2% kļūda.

### Funkcijai *IsFontSize* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsFontSize(„Document”, , 12)</code>	Vai viss dokuments ir 12. izmēra fontā?
2.	<code>IsFont(„Document”, „Arial”) &amp; IsFontSize(„Document”, , 12)</code>	Vai viss dokuments ir 12. izmēra Arial fontā?
3.	<code>IsFontSize(„Paragraph”, „apsv”, 18)</code>	Vai paragrāfs, kas satur burtus „apsv”, ir 18. lieluma fontā? Identificējot paragrāfu, mēs varam nenorādīt pilnu tā tekstu, un varam arī nerūpēties par lielajiem un mazajiem burtiem (kas gan nav labs formulu rakstīšanas stils).

## 2.7. Funkcija *IsTextFormatted*

Funkcija *IsTextFormatted* pārbauda, vai norādītais dokumenta fragments ir noformēts ar norādītā lieluma formatējumu: Bold, Italic, Underline, Subscript, Superscript.

Funkcijai *IsTextFormatted* ir sekojoši argumenti:

1. fragmenta veids,
2. meklēšanas kritērijs,
3. formatējums.

Tiek pārbaudīts tikai norādītais formatējums – citi formatējumi var būt, un var nebūt. Tāpat funkcija nepārbauda arī fonta nosaukumu un lielumu.

### Funkcijai *IsTextFormatted* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsTextFormatted(„Word”, „ECDL”, „Italic”)</code>	Vai vārds „ECDL” ir kursīvā?
2.	<code>IsTextFormatted(„Word”, „ECDL”, „Italic”) &amp; ! IsTextFormatted(„Word”, „septīto”, „Italic”) &amp; ! IsTextFormatted(„Word”, „testu”, „Italic”)</code>	Vai vārds „ECDL” ir kursīvā? Ja uzdevums ir bijis noformēt kursīvā vienu vārdu, ir lietderīgi pārbaudīt, vai arī blakus vārdi nejauši nav noformēti kursīvā. Ideāli būtu pārbaudīt visu pārējo dokumentu, bet arī ar blakus vārdu pārbaudi praktiskām vajadzībām pietiek.
3.	<code>IsFont(„Document”, „Arial”) &amp; IsFontSize(„Document”, 12) &amp; !IsTextFormatted(„Document”, „Bold”) &amp; !IsTextFormatted(„Document”, „Italic”) &amp; !IsTextFormatted(„Document”, „Underline”)</code>	Vai viss dokuments ir 12. izmēra Arial fontā bez liekiem formatējumiem?

## 2.8. Funkcija *IsFontColor*

Funkcija *IsFontColor* pārbauda, vai norādītais dokumenta fragments ir noformēts norādītajā krāsā.

Funkcijai *IsFontColor* ir sekojoši argumenti:

1. fragmenta veids,
2. meklēšanas kritērijs,
3. krāsa.

Tiek pārbaudīta fonta krāsa (nevis fona krāsa). Krāsa var būt: balts, melns, pelēks, sarkans, zaļš, zils. Krāsu salīdzināšana ir ļoti liberāla, visas šaubas tiek tulkotas par labu studentam. Tā, piemēram, par baltu tiks atzīta arī viegli pelēka krāsa, bet par sarkanu – arī sarkanbrūna. Ja students būs izmantojis krāsu „Authomatic”, tad sistēma tehnisku ierobežojumu dēļ, uzdevumu pārbaudot, uzskatīs, ka lietota melna krāsa, kaut faktiski „Authomatic” krāsa ir atkarīga no fona krāsas. Tāpēc uzdevumus sagatavojot jālieto tādas fona krāsas, lai „Authomatic” atbilstu melnai krāsai arī MS Word izpratnē.

### Funkcijai *IsFontColor* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsFontColor(„Document”, „Melns”)</code>	Vai viss dokuments ir melnas krāsas fontā? Fonta krāsa „Authomatic” tiek saprasta kā melna.
2.	<code>IsFont(„Document”, „Arial”) &amp; IsFontSize(„Document”, 12) &amp; !IsTextFormatted(„Document”, „Bold”) &amp; !IsTextFormatted(„Document”, „Italic”) &amp; !IsTextFormatted(„Document”, „Underline”) &amp; IsFontColor(„Document”, „Melns”)</code>	Vai viss dokuments ir 12. izmēra Arial fontā bez liekiem formatējumiem un melnā krāsā?

## 2.9. Funkcija *IsShadingColor*

Funkcija *IsShadingColor* pārbauda, vai norādītais dokumenta fragmentam ir norādītā ēnojuma krāsa. Funkcija ir tāda pat, kā *IsFontColor* (sk. 2.8), tikai pārbauda nevis fonta, bet ēnojuma krāsu.

### Funkcijai *IsShadingColor* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsShadingColor(„Document”, „Pelēks”)</code>	Vai viss dokuments ir ar pelēku ēnojumu?
2.	<code>IsFont(„Document”, „Arial”) &amp; IsFontSize(„Document”, 12) &amp; !IsTextFormatted(„Document”, „Bold”) &amp; !IsTextFormatted(„Document”, „Italic”) &amp; !IsTextFormatted(„Document”, „Underline”) &amp; IsFontColor(„Document”, „Meln”) &amp; IsShadingColor(„Document”, „Balts”)</code>	Vai viss dokuments ir 12. izmēra Arial fontā bez liekiem formatējumiem un melnā krāsā uz balta fona?

## 2.10. Funkcija *IsStyle*

Funkcija *IsStyle* pārbauda, vai norādītais dokumenta fragments ir noformēts norādītajā stilā.

Funkcijai *IsStyle* ir sekojoši argumenti:

1. fragmenta veids,
2. meklēšanas kritērijs,
3. stila nosaukums.

Tiek pārbaudīts tikai stila nosaukums, nevis pats noformējums – tā piemēram, no diviem identiskiem stiliem ar dažādiem nosaukumiem, par pareizu tiks atzīts tikai funkcijā norādītais. Nekādas modifikācijas netiek pieļautas (tā „Normal + 11,9pt” nav „Normal”, neskatoties uz to, ka fonta izmēra maiņa ir pieļaujamās kļūdas robežās).

### Funkcijai *IsStyle* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsStyle(„Document”, „Normal”)</code>	Vai viss dokuments ir stilā „Normal”? Šī pārbaude ir saprātīga alternatīva nodaļas 2.9 otrajam piemēram – daudz īsāka un precīzāka. Tā gan ir atkarīga no stila „Normal” definīcijas konkrētajā situācijā.

## 2.11. Funkcija IsParagraphFormatted

Funkcija IsParagraphFormatted pārbauda, vai norādītajā dokumenta fragmentā ietilpstošajiem paragrāfiem ir norādītais noformējums.

Funkcijai IsParagraphFormatted ir sekojoši argumenti:

1. fragmenta veids,
2. meklēšanas kritērijs,
3. formatējums,
4. lielums (skaitlis).

Formatējums ir: Left, Right, Centered, Justified, Line spacing x, Spacing before x, Spacing after x, Left indent x, Right indent x, Hanging x, First line x, Left tab x, Right tab x, Centered tab x, Decimal tab x, Numbered, Bulleted, Bordered x. Formatējumiem, kuriem būtisks lielums (x), to norāda kā ceturto argumentu.

Formatējumi Left, Right, Centered, Justified, Line spacing x, Spacing before x, Spacing after x, Left indent x, Right indent x, Hanging x, First line x atbilst Word izvēlnes Format -> Paragraph vērtībām. Left tab x, Right tab x, Centered tab x, Decimal tab x pārbauda, vai paragrāfā ar attiecīgo nobīdi (centimetros) ir uzstādīta attiecīgā veida tabulācija. Numbered un Bordered (Format -> Bullets and numbering) pārbauda, vai paragrāfs ir numurēta (ar jebkādiem numerācijas uzstādījumiem) vai ar speciālu simbolu izdalīta (ar jebkuru simbolu un jebkuriem uzstādījumiem) saraksta daļa. Bordered x pārbauda, vai paragrāfam ir rāmis Format -> Borders and shading izpratnē (x norāda līnijas treknumu (Width) punktus, nekādi citi līnijas atribūti pārbaudīti netiek).

Skaitliskā vērtība x vienmēr (izņemot Border) norāda attālumu centimetros (nevis collās vai punktus, vai rindiņās). Tiek pieļauta +/-2% kļūda.

### Funkcijai IsParagraphFormatted lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsParagraphFormatted(„Document”, „Justified”)	Vai visos paragrāfos teksts ir izlīdzināts gar kreiso un labo malu?
2.	IsParagraphFormatted(„Paragraph”, 1, „Left”) & IsParagraphFormatted(„Paragraph”, 3, „Justified”) & IsParagraphFormatted(„Paragraph”, 5, „Left”)	Vai pamatteksts (sk. piemēru) ir izlīdzināts, bet virsraksts un atvadišanās – kreisajā malā (vizuāli nav redzams, kamēr teksts nepārsniedz vienu rindiņu)?
3.	IsParagraphFormatted(„Document”, „Spacing after”, 0,21)	Vai visam dokumentā atstarpe starp paragrāfiem ir 0,21cm (aptuveni 6pt)?



## **2.12. Funkcija IsDocumentFormatted**

Funkcija IsDocumentFormatted pārbauda, vai dokumentam ir norādītais noformējums.

Funkcijai IsDocumentFormatted ir sekojoši argumenti:

1. formatējums,
2. lielums 1 (skaitlis),
3. lielums 2 (skaitlis).

Formatējums ir: Portrait, Lanscepe, Page size x y, Top margin x, Bottom margin x, Left margin x, Right margin x. Formatējumiem, kuriem būtisks lielums (x, y), to norāda kā otro un trešo argumentu (norāda centimetros). Skaitliskajiem lielumiem tiek pieļauta +/-2% kļūda.

Pārbaude vienmēr attiecas uz visu dokumentu. Ja dokumentā, piemēram, ir vairākas sekcijas, un tajās ir dažādi noformējumi, tad pārbaudītas tiks visas sekcijas, un funkcijas atbilde visdrīzāk būs „Nē”.

### **Funkcijai IsDocumentFormatted lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsDocumentFormatted(„Landscape”)	Vai dokuments sagatavots ainavorientētam papīram?
2.	IsDocumentFormatted(„Page size”, 21, 29,7)	Vai papīra izmērs ir A4 (21x29,7cm)?

### **2.13. Funkcija *IsTextInPage***

Funkcija *IsTextInPage* pārbauda, vai norādītais dokumenta fragments ir norādītajā lappusē.

Funkcijai *IsTextInPage* ir sekojoši argumenti:

1. fragmenta veids,
2. meklēšanas kritērijs,
3. lappuse (skaitlis).

Visam norādītajam fragmentam jāatrodas norādītajā lappusē.

#### **Funkcijai *IsTextInPage* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<code>IsTextInPage(„Document”, , 1)</code>	Vai viss dokuments ietilpst vienā lappusē?
2.	<code>IsTextInPage(„Paragraph”, „Labdien”, 1) &amp; IsTextInPage(„Paragraph”, „Apsveicam”, 2)</code>	Vai paragrāfs „Labdien” ir pirmajā lappusē, bet paragrāfs „Apsveicam”- otrajā?
3.	<code>IsTextInPage(„Paragraph”, 4, 2) &amp; !IsTextInPage(„Paragraph”, 3, 1) &amp; !IsTextInPage(„Paragraph”, 3, 2)</code>	Ja ceturtais paragrāfs ir otrajā lappusē, bet trešais paragrāfs nav ne pirmajā, ne otrajā, tas nozīmē, ka lappuses mainās trešajam paragrāfam pa vidu.

## **2.14. Funkcija IsHeader**

Funkcija IsHeader pārbauda, vai dokumentam ir galvene ar norādīto tekstu.

Funkcijai IsHeader ir viens arguments:

1. teksts.

Tiek pārbaudīts tikai parasts teksts, ne lappuses numurs vai drukāšanas datums. Arī teksta noformējums netiek pārbaudīts.

Pārbaude attiecas uz visu dokumentu – uz katru tā lappusi

### **Funkcijai IsHeader lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsHeader(„Sistēma Tests. Lietotāja ceļvedis”)	Vai viss dokuments ietilpst vienā lappusē?

Lai pārbaudītu, vai students prot lietot galvenē sarežģītākas konstrukcijas, piemēram, lappuses numuru, jautājumu jāformulē netieši un jāpārbauda bez formulām. Piemēram, tas varētu būt ar jā/nē atbildams jautājums: „Vai dokumenta galvenē ir lietota automātiskā lappuses numura parādīšana” (un, lai pareizā atbilde būtu „Nē”, lappuses numuri varētu būt ievadīti ne automātiski, bet tieši tekstā).

## **2.15. Funkcija IsFooter**

Funkcija IsFooter pārbauda, vai dokumentam ir kājenē ar norādīto tekstu. Funkcija ir tāda pati, kā IsHeader (sk. 2.14), tikai pārbauda nevis galveni, bet kājeni.

## 2.16. Funkcija *IsTable*

Funkcija *IsTable* pārbauda, vai dokumentā ir tieši viena tabula.

Funkcijai *IsTable* argumentu nav.

### Funkcijai *IsTable* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsTable</i>	Vai dokumentā ir tieši viena, vienalga, kāda, tabula?
2.	<i>!IsTable</i>	Pārbauda tieši pretējo, vai tabulas tiešām nav. Šāda pārbaude būtu lietojama uzdevumā „Izdzēsiet no dokumenta tabulu” (kas nemaz nav triviāls).

## 2.17. Funkcija *IsTableDimensions*

Funkcija *IsTableDimensions* pārbauda, vai dokumentā ir tieši viena tabula, un tai ir norādītais rindu un kolonu skaits.

Funkcijai *IsTableDimensions* ir sekojoši argumenti:

1. rindu skaits,
2. kolonu skaits.

Funkcija atbild „Nē”, ja dokumentā nav nevienas tabulas, dokumentā ir vairākas tabulas, vai arī, ja dokumentā ir viena tabula, bet tai nav vajadzīgais rindu un kolonu skaits. Šāds princips attiecas arī uz citām funkcijām, kas pārbauda tabulas.

### Funkcijai *IsTableDimensions* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsTableDimensions</i> (4, 3)	Vai dokumentā ir viena tabula, un vai tai ir 4 rindas un 3 kolonas?
2.	<i>IsTableDimensions</i> (4, 3) & <i>!ContainsText</i> („Document”, „4.”) & <i>!ContainsText</i> („Document”, „5.”) & <i>!ContainsText</i> („Document”, „6.”)	Šādu formulu varētu lietot pārbaudot uzdevumu „Izdzēst pēdējās 2 tabulas rindas”. Mēs pārbaudām, vai tabulai ir vajadzīgās dimensijas, un, pieņemot, ka pirmā kolona satur rindu numurus, heuristiski pārbaudām, vai izdzēstas ir pēdējās rindas.

## 2.18. Funkcija *IsCellText*

Funkcija *IsCellText* pārbauda, vai tabulas norādītajā ailē ir vajadzīgais teksts.

Funkcijai *IsCellText* ir sekojoši argumenti:

1. rinda,
2. kolona,
3. teksts.

Funkcija darbojas tāpat, kā *IsText* (2.1), tikai dokumenta fragmenta vietā tiek pārbaudīta tabulas aile. Pārbaude attiecas uz vienīgo tabulu dokumentā. Ja tabulā šādas ailes nav, atbilde ir „Nē”. Šāds princips attiecas arī uz citām funkcijām, kas pārbauda tabulas

### Funkcijai *IsCellText* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsTableDimensions(2, 2) &amp; IsCellText(1, 1, „Stiprās puses”) &amp; IsCellText(1, 2, „Vājās puses”) &amp; IsCellText(2, 1, „Iespējas”) &amp; IsCellText(2, 2, „Draudi”)</code>	Uzdevums varētu būt bijis sagatavot 2x2 tabulu SVID analīzei.
2.	<code>IsTableDimensions(4, 3) &amp; IsCellText(1, 1, „Nr.”) &amp; IsCellText(2, 1, „1.”) &amp; IsCellText(3, 1, „2.”) &amp; IsCellText(4, 1, „3.”)</code>	Alternatīvs, bet vienalga aptuvens, risinājums 2.17 otrajam piemēram.

## **2.19. Funkcija *IsColumnWidth***

Funkcija *IsColumnWidth* pārbauda, vai tabulas norādītajai kolonai ir vajadzīgais platums.

Funkcijai *IsColumnWidth* ir sekojoši argumenti:

1. kolona,
2. platums.

Platums tiek norādīts centimetros. Salīdzinot platumu, tiek pieļauta +/-2% kļūda.

### **Funkcijai *IsColumnWidth* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsColumnWidth</i> (1, 1,2) & <i>IsColumnWidth</i> (2, 7) & <i>IsColumnWidth</i> (3, 7)	Tabula ar vienu šauru kolonnu numuram un divām platākām - tekstam.

## **2.20. Funkcija *IsRowHeight***

Funkcija *IsRowHeight* pārbauda, vai tabulas norādītajai rindai ir vajadzīgais augstums. Funkcija ir tāda pati, kā *IsColumnWidth* (sk. 2.19), tikai pārbauda nevis kolonas platumu, bet rindas augstumu. Augstums 0 nozīmē, ka tabulas rindai augstums nav norādīts un mainās atkarībā no teksts.

### **Funkcijai *IsRowHeight* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsRowHeight</i> (1, 0) & <i>IsRowHeight</i> (2, 0) & <i>IsRowHeight</i> (3, 0)	Formula varētu būt papildus pārbaude kādā tabulas aizpildīšanas uzdevumā, jo rindas augstuma mainīšana ir tipiska iesācēju kļūda.

## 2.21. Funkcija *IsCellMargin*

Funkcija *IsCellMargin* pārbauda, vai norādītajai ailei ir norādītā izmēra labā, kreisā, augšējā vai apakšējā apmale.

Funkcijai *IsCellMargin* ir sekojoši argumenti:

1. rinda,
2. kolona,
3. apmales tips,
4. apmales lielums.

Apmales tips ir: Left, Right, Top, Bottom. Apmales lielums tiek norādīts centimetros. Salīdzinot apmales lielumu, tiek pieļauta +/-2% kļūda.

### Funkcijai *IsCellMargin* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsCellMargin</i> (1, 1, „Left”, 0,19) & <i>IsCellMargin</i> (1, 1, „Right”, 0,19) & <i>IsCellMargin</i> (1, 1, „Top”, 0) & <i>IsCellMargin</i> (1, 1, „Bottom”, 0)	Pārbauda, vai apmaļu lielumam ir noklusētā vērtība.

## 2.22. Funkcija *IsCellBorder*

Funkcija *IsCellBorder* pārbauda, vai norādītajai ailei ir norādītā veida rāmis.

Funkcijai *IsCellBorder* ir sekojoši argumenti:

1. rinda,
2. kolona,
3. līnijas tips,
4. līnijas treknums.

Līnijas tips norāda, vai līnija ir nepārtraukta, pārtraukta, punktēta (iespējamās vērtības numurētas saskaņā ar attiecīgo Word izvēlni). Apmales treknums tiek norādīts punktos. Salīdzinot apmales treknumu, tiek pieļauta +/-2% kļūda.

### Funkcijai *IsCellBorder* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsCellBorder</i> (1, 1, 1, 0,5) & <i>IsCellBorder</i> (1, 2, 1, 0,5) & <i>IsCellBorder</i> (2, 1, 1, 0,5) & <i>IsCellBorder</i> (2, 2, 1, 0,5)	2x2 tabula ar visparastāko rāmi.

### **2.23. Funkcija IsCellShading**

Funkcija IsCellShading pārbauda, vai norādītajai ailei ir norādītais iekrāsojums.

Funkcijai IsCellShading ir sekojoši argumenti:

1. rinda,
2. kolona,
3. krāsa.

Krāsa tiek norādīta tāpat, kā funkcijā IsShadingColor (sk. 2.9).

#### **Funkcijai IsCellShading lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsTableDimensions(3, 3) & IsCellShading(1, 1, „Pelēks”) & IsCellShading(1, 2, „Pelēks”) & IsCellShading(1, 3, „Pelēks”) & IsCellShading(2, 1, „Balts”) & IsCellShading(2, 2, „Balts”) & IsCellShading(2, 3, „Balts”) & IsCellShading(3, 1, „Balts”) & IsCellShading(3, 2, „Balts”) & IsCellShading(3, 3, „Balts”)	Pirmās, virsraksta rindas ēnojums ir pelēks, bet pārējās tabulas – No Fill.



## 2.24. Funkcija *IsObject*

Funkcija *IsObject* pārbauda, vai dokumentā ir tieši norādītais skaits objektu – attēlu, grafiku u.tml.

Funkcijai *IsObject* ir sekojoši argumenti:

1. skaits.

Ne ar šo, ne ar citām funkcijām nav iespējams noteikt objektu tipu (attēls, diagramma...).

### Funkcijai *IsObject* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsObject</i> (0)	Pēc uzdevuma „Izdzēst no dokumenta grafiku” pārbauda, vai tas ir izdarīts.
2.	<i>IsObject</i> (1)	Sākotnējā dokumentā ir divi attēli, viens otram tieši blakus, un tādi, kas vizuāli veido vienu veselu. Liekot studentam izdzēst vienu no tiem, tiek netieši pārbaudīts, vai viņš prot attēlu iezīmēt.

## 2.25. Funkcija *IsObjectSize*

Funkcija *IsObjectSize* pārbauda, vai dokumentā ir tieši viens objekts, un vai tam ir norādītais platums un augstums.

Funkcijai *IsObjectSize* ir sekojoši argumenti:

1. platums,
2. augstums.

Platums un augstums tiek norādīti centimetros. Tiek pieļauta +/-2% kļūda.

### Funkcijai *IsObjectSize* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsObject</i> (1) & <i>IsObjectSize</i> (9, 12)	Uzdevumā varētu būt prasīts ielikt dokumentā attēlu c:\tests\photo.jpg un uzstādīt attēla mērogošanu 100%. Izmēri netieši norāda, ka ielikts ir tieši prasītais attēls.

## 2.26. Funkcija IsMailMerge

Funkcija IsMailMerge pārbauda, vai dokuments ir operācijas MailMerge rezultāts.

Funkcijai IsMailMerge ir sekojoši argumenti:

1. platums,
2. augstums,
3. aiļu saturs.

Platums un augstums tiek norādīti centimetros. Tiek pieļauta +/-2% kļūda.

Aiļu saturs tiek norādīts kā viens teksta lauks (potenciāli ļoti garš), kurā pēc kārtas parādīts visu aiļu saturs, atdalot vienu aili no nākamās ar semikolu.

Funkcija pārbauda:

1. Vai dokuments sastāv no vienas tabulas, un vai tajā bez šīs tabulas nav neviena cita paragrāfa.
2. Vai visas tabulas ailes ir ar tādu izmēru, kā norādīts.
3. Vai tabulas aiļu saturs pēc kārtas ir tāds, kā norādīts trešajā argumentā. Teksta salīdzināšana ir ļoti liberāla – nebūtiskās tuksnes, paragrāfa beigu zīmes un citi formatējuma elementi netiek salīdzināti.

Šāda pārbaude vēl pilnīgi droši neliecina, ka lietota operācija MailMerge, tomēr praktiskai lietošanai tā ir pietiekama.

### Funkcijai IsMailMerge lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
2.	IsMailMerge(7, 4,23, „Jānis Kalniņš Brīvības 1-12 Rīga; Pēteris Lejiņš Rīgas 1-12 Liepāja”)	Izmērs 7x4,23 atbilst uzlīmēm Herma 5054, un jau tas vien gandrīz droši liecina, ka lietota operācija MailMerge. Aiļu satura pārbaude parāda, ka lietots pareizais datu avots.

## 2.27. Funkcija *IsTextRepeated*

Funkcija *IsTextRepeated* pārbauda, vai dokumentā norādītais teksts parādās tieši norādīto skaitu reižu.

Funkcijai *IsTextRepeated* ir sekojoši argumenti:

1. teksts,
2. skaits.

### Funkcijai *IsMailMerge* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsTableDimensions(648, 3) &amp; IsColumnWidth(1, 7) &amp; IsColumnWidth(3, 7) &amp; IsColumnWidth(2, 7) &amp; IsTextRepeated(„Rīga”, 1938)</code>	Formula var tikt lietota ļoti apjomīga <i>MailMerge</i> pārbaudei. Mēs zinām, ka ir 1938 datu ieraksti, tāpēc vārdam „Rīga” būtu jāparādās 1938 reizes (pieņemot, ka tiek veidotas adrešu uzlīmes, un, ka visas adreses ir Rīgā). Kolonu platumu nosaka uzlīmju tips ( šeit - Herma 5054), bet rindu skaitu tabulā – ierakstu skaits datu avotā. Visi faktori kopumā liecina, ka operācija <i>MailMerge</i> ir lietota – lielā datu apjoma dēļ manuāli šādi tabulu iegūt nebūtu iespējams.
2.	<code>IsTextRepeated(„Kalniņa kungs”, 20)</code>	Ja mērķis ir pārbaudīt, vai students prot lietot <i>Copy</i> un <i>Paste</i> , uzdevums var būt „Nokopēt paragrāfu „Labrīt, Kalniņa kungs!”20 reizes”. Nezinot <i>Copy</i> un <i>Paste</i> , šo uzdevumu reālā laikā izpildīt nav iespējams.

### 3. Pārbaudes funkcijas Excel un Calc vidē

Praktiskie uzdevumi izklājlapu veidošanā tiek pildīti un pārbaudīti līdzīgi kā uzdevumi Word vidē – studenta darba rezultāts ir Excel vai Calc izklājlapa (.xls fails), un rezultāta pārbaude notiek ar funkciju palīdzību.

Tāpat, kā Word gadījumā, pārbaude, ko realizē funkcija, parasti neattiecas uz visu izklājlapu, tā attiecas uz kādu tās apgabalu – šūnu vai šūnu grupu. Vairumam funkciju pirmais arguments nosaka izklājlapas apgabalu, uz kuru funkcija attiecas, bet pārējie argumenti nosaka, kas tiek pārbaudīts.

Apgabals tiek norādīts MS Excel sintaksē – sistēma to pārveido Calc sintaksē, ja nepieciešams. Piemēram, funkcija IsCellText(„C2”, „Izdevumi”) pārbauda, vai šūnā C2 ir ierakstīts teksts „Izdevumi”, bet IsFormatted(„C1:C3”, „Arial”, „Bold”) pārbaude, vai šūnas C1, C2 un C3 ir noformētas fontā Arial Bold.

Funkcijā norādītā pārbaude attiecas arī uz paslēptām rindām un kolonām. Ja nav norādīta darba lapa, tad tiek ņemta faila pirmā darba lapa. Ja norādītās darba lapas failā nav, tad visas funkcijas uzreiz atbild „Nē”.

Faila vārdu apgabālā norādīt nav atļauts. Tiek pieņemts, ka sistēma zina faila vārdu un mapīti, kurā tam jāatrodas. Ja pēc kādas nekorektas studenta darbības fails nav atrodams, visas funkcijas atbild „Nē”.

Funkcijās visi izmēri tiek norādīti centimetros (interpretācija nav atkarīga no Excel/Calc uzstādījumiem). Krāsas tiek norādītas tāpat, kā Word funkcijās (sk. 2.8). Funkciju argumentos norādītie teksti nedrīkst saturēt speciālos simbolus: apostrofu, pēdiņu, semikolu, konjunkcijas zīmi un slīpsvītru \. Ja vienā argumentā jānorāda vairākas vērtības, tad tās atdala ar semikolu.

Turpmākajos piemērus tiek izmantota sekojoša izklājlapa (tiek pieņemts, ka darba lapas vārds ir „Izdevumi”).

Izdevumi kancelejas preču iegādei				
Nr.	Prece	Cena	Skaitis	Summa
1.	Pildspalva	0,72	10	7,20
2.	Zīmulis	0,50	20	10,00
3.	Piezīmju papīrs	1,30	5	6,50
4.	Klade	1,85	5	9,25
5.	Lineāls	3,20	1	3,20
			<b>Kopā:</b>	<b>36,15</b>

### 3.1. Funkcija *IsWorksheetOption*

Funkcija *IsWorksheetOption* pārbauda, vai darba lapai ir uzstādīta vajadzīgā opcija.

Funkcijai ir sekojoši argumenti:

1. darba lapas vārds,
2. opcijas nosaukums,
3. vērtība.

Opcijas un vērtības var būt šādas:

1. „Zoom” ar skaitlisku vērtību x pārbauda, vai izklājlapa tiek skatīta mērogā x%. Mērogam tiek pieļauta kļūda +/-5%, lai izslēgtu starpgadījumus, kas saistīti ar dažādām ekrāna izšķirtspējām, izmēriem collās u.tml.
2. „Page. Orientation” ar vērtību „Portrait” vai „Landscape”. Pārbauda, vai dokuments sagatavots drukāšanai attiecīgā režīmā. Šī un visas turpmākās opcijas, ko pārbauda funkcija *IsWorksheetOption*, ir uzstādāmas Excel izvēlnē „File -> Page Setup”.
3. „Sheet. Row to repeat at top” ar tekstuālu vērtību (piemēram, „\$3:\$3”) pārbauda, vai ir uzstādīta katras lappuses sākumā drukājamā rindiņa.
4. „Page. Fit to pages wide” ar skaitlisku vērtību n. Pārbauda, vai izklājlapa tiks drukāta n lappušu platumā.
5. „Page. Fit to pages tall” ar skaitlisku vērtību n. Pārbauda, vai izklājlapa tiks drukāta n lappušu garumā.
6. „Sheet. Gridlines” ar tekstuālu vērtību „Yes” vai „No”. Pārbauda, vai tiks (netiks) izdrukāts šūnu režģis.
7. „Sheet. Row and column headings” ar tekstuālu vērtību „Yes” vai „No”. Pārbauda, vai tiks (netiks) izdrukāti rindu un kolonu virsraksti.
8. Opcijas „Margins. Top”, „Margins. Bottom”, „Margins. Left” un „Margins. Right” ar skaitlisku vērtību x. Pārbauda, vai ir uzstādīta attiecīgā lapas apmale. Apmales lielumam tiek pieļauta kļūda +/-5%.

#### Funkcijai *IsWorksheetOption* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsWorksheetOption</i> („Izdevumi”, „Zoom”, 100)	Vai izklājlapa uz ekrāna tiks rādīta 100% mērogā?
2.	<i>IsWorksheetOption</i> („Izdevumi”, „Sheet. Row to repeat at top”, „\$3:\$3”)	Trešajā argumentā tiek ierakstīta precīzi tā pati formula, kas attiecīgajā Excel izvēlnē.
3.	<i>IsWorksheetOption</i> („Izdevumi”, „Page. Fit to pages wide”, 1) & <i>IsWorksheetOption</i> („Izdevumi”, „Page. Fit to pages tall”, 1)	Vai izklājlapa tiks izdrukāta tieši uz vienas lapas (kā platumā, tā garumā)

Funkcija `IsWorksheetOption` dod iespēju pārbaudīt tikai tās opcijas, kas tiek saglabātas kopā ar izklājlapu. Lai pārbaudītu, vai students prot uzstādīt „globālās” Excel opcijas, jāizmanto netieši uzdevumi vai teorētiski jautājumi.

Piemēram, lai pārbaudītu, vai students saprot, kā uzstāda noklusētu failu glabāšanas mapi (Default file location), var veidot praktisku uzdevumu Excel vidē, uz kuru jāatbild ar frāzi, un uzdevumu formulēt šādi:

„Ierakstiet atbildē, kāds ir Microsoft Excel noklusētā failu glabāšanas mape (Default file location).”

Pārbaudes funkcija tad būtu šāda:

```
IsMatch(&0, „C:\Documents and Settings\*\My Documents”)
```

Studenta atbilde tiek pārbaudīta ar teorētisko jautājumu pārbaudes funkciju `IsMatch` (sk. 1), un pēc būtības tiek lietota regulāra izteiksme (simbols \* apzīmē jebkuru tekstu), jo taka uz mapi ir atkarīga no Windows lietotāja vārda.

### 3.2. Funkcija *IsWorksheet*

Funkcija `IsWorksheet` pārbauda, vai fails satur norādītu darba lapu.

Funkcijai ir sekojoši argumenti:

1. darba lapas vārds,
2. darba lapas numurs pēc kārtas.

#### Funkcijai `IsWorksheet` lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsWorksheet(„Izdevumi”, 1)</code>	Vai pirmā darba lapa failā saucas „Izdevumi”?
2.	<code>IsWorksheet(&amp;0, 1)</code>	Varētu būt pārbaude ar frāzi atbildamam jautājuma: „Kā saucas pirmā darba lapa failā?”
3.	<code>IsWorksheet(&amp;1, 1)</code>	Varētu būt pārbaude praktiskam uzdevumam ar parametru: „Nomainiet 1. darba lapas vārdu uz &1”
4.	<code>IsWorksheet(„Izdevumi”, 99)</code>	Funkcija atbildēs „Nē”, ja failā nebūs 99 darba lapu (tāpat arī tad, ja 99. nesauksies „Izdevumi”).

### **3.3. Funkcija *IsFormattedNumber***

Funkcija *IsFormattedNumber* pārbauda, vai šūnām ir norādītais formatējums Excel izvēlnes „Format Cells” sadaļas „Number” izpratnē.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. formatējums.

Formatējumu norāda kā šabloni, piemēram, „Ls # ##0;-Ls # ##0”. Šablonus biežāk lietojamajiem formatējumiem var iegūt Excel vidē, vispirms uzstādot šūnai interesējošo formatējumu (piemēram, Currency), bet pēc tam mainot šo formatējumu uz Custom – Currency atbilstošais šablons tiek parādīts šablona ievadlaukā.

#### **Funkcijai *IsFormattedNumber* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsFormattedNumber</i> („C4:C8”, „# ##0,00”)	Vai šūnas C4, C5, C6, C7 un C8 noformētas kā skaitļi ar 2 zīmēm aiz komata?
2.	<i>IsFormattedNumber</i> („D9”, „@”)	Vai šūna D9 noformētas kā teksts (Text)?
3.	<i>IsFormattedNumber</i> („D4:D8”, „dd.mm.yyy”)	Vai šūnas D4:D8 noformētas kā datums?

### **3.4. Funkcija *IsFormattedText***

Pašreizējā sistēmas Tests realizācijā funkcija sakrīt ar *IsFormattedNumber*, tomēr, lai uzlabotu formulu lasāmību, teksta formatējumu ieteicams pārbaudīt ar šo funkciju.

### **3.5. Funkcija *IsFormattedDate***

Pašreizējā sistēmas Tests realizācijā funkcija sakrīt ar *IsFormattedNumber*, tomēr, lai uzlabotu formulu lasāmību, datuma formatējumu ieteicams pārbaudīt ar šo funkciju.

### **3.6. Funkcija *IsFormattedCurrency***

Pašreizējā sistēmas Tests realizācijā funkcija sakrīt ar *IsFormattedNumber*, tomēr, lai uzlabotu formulu lasāmību, naudas vienību formatējumu ieteicams pārbaudīt ar šo funkciju.

### **3.7. Funkcija *IsFormattedSpecial***

Pašreizējā sistēmas Tests realizācijā funkcija sakrīt ar *IsFormattedNumber*.

### 3.8. Funkcija *IsFormattedAllignement*

Funkcija *IsFormattedAllignement* pārbauda, vai šūnām ir norādītais formatējums Excel izvēlnes „Format Cells” sadaļas „Allignement” izpratnē.

Funkcijai ir sekojoši argumenti (sākot ar otro tie atbilst attiecīgajiem Excel lodziņiem):

1. apgabals,
2. horizontal: „General”, „Center”, „Fill”, „Justify” vai „Right”,
3. vertical: „Top”, „Center” vai „Bottom”,
4. indent: skaitlis,
5. orientation degrees: skaitlis,
6. wrap: „Jā” vai „Nē”,
7. shrink: „Jā” vai „Nē”,
8. merge: „Jā” vai „Nē”,
9. text direction: „Context”, „Left-to-right” vai „Right-to-left”.

Jebkuru no argumentiem 2.-9. var nenorādīt, tādā gadījumā attiecīgā pārbaude netiek veikta.

Piezīme. Funkcijas nosaukums tiešām ir *IsFormattedAllignement* (ar diviem l), nevis *IsFormattedAlignement* (ar vienu l), kā tas būtu gramatiski pareizi.

#### Funkcijai *IsFormattedAllignement* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsFormattedAllignement</i> („A3:E3”, „Center”, „Center”, „Jā”)	Vai šūnas A3:E3 ir gan horizontāli, gan vertikāli centrētas, un vai tām norādīta automātiska pārvešana uz jaunu rindu (wrap)? 4. un 5. parametrs ir izlaisti.
2.	<i>IsFormattedAllignement</i> („D9”, „Right”) & <i>IsFormattedFont</i> („D9”, „Arial”, „Bold”) & <i>IsCellText</i> („D9”, „Kopā:”)	Vai šūna D9 („Kopā:”) ir izlīdzināta pa labi un noformēta treknā fontā?



### 3.9. Funkcija **IsFormattedFont**

Funkcija **IsFormattedFont** pārbauda, vai šūnām ir norādītais fonts un formatējums Excel izvēlnes „Format Cells” sadaļas „Font” izpratnē.

Funkcijai ir sekojoši argumenti (sākot ar otro tie atbilst attiecīgajiem Excel lodziņiem):

1. apgabals,
2. font: teksts – fonta nosaukums (piemēram, „Arial”),
3. font style: „Regular”, „Italic”, „Bold” vai „Bold Italic”,
4. size: skaitlis (fonta izmērs),
5. underline: „Jā” vai „Nē”,
6. krāsa: balts, melns, pelēks, sarkans, zaļš, zils (par krāsu pārbaudi sk. 2.8),
7. effects: „Strikethrough”, „Superscript” vai „Subscript”.

Jebkuru no argumentiem 2.-7. var nenorādīt, tādā gadījumā attiecīgā pārbaude netiek veikta.

#### **Funkcijai **IsFormattedFont** lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<code>IsFormattedFont(„A1”, „Arial”, „Bold”, 12, „Meln” )</code>	Vai šūna A1 ir Arial 12 Bold un melnā krāsā?
2.	<code>IsFormattedFont(„A1”, „Arial”, „Bold”, 12) &amp; IsFormattedAllignement(„A1”, „Left”) &amp; IsFormattedAllignement(„A1” „”, „Nē”) &amp; IsCellText(„A1”, „Izdevumi kancelejas preču iegādei”)</code>	Vai šūnai A1 ir tipisks virsraksta noformējums? Trešajā formulā lietota negācija, jo šūnai nedrīkst būt norādīts „Wrap” režīms. Turpat pieci komati pēc kārtas norāda, ka četri argumenti ir izlaisti un netiek pārbaudīti.

### 3.10. Funkcija **IsFormattedBorder**

Funkcija **IsFormattedBorder** pārbauda, vai šūnām ir norādītais rāmis Excel izvēlnes „Format Cells” sadaļas „Border” izpratnē.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. līnijas tips
3. līnijas stils,
4. līnijas krāsa.

„līnijas tips” norāda pārbaudāmās līnijas vietu: horizontālā augšējā, horizontālā apakšējā, slash diagonāle, vertikālā kreisā, vertikālā vidējā, vertikālā labā, backslash diagonāle. Lai pārbaudītu vairākus līniju tipus, lieto vairākas funkcijas **IsFormattedBorder**, kas savienotas ar „un” zīmi.

„līnijas stils” norāda līnijas veidu: vienkārša vai dubulta vai līnijas nav nemaz (citus līniju veidus norādīt nav iespējams)

„līnijas krāsa” – balts, melns, pelēks, sarkans, zaļš, zils (par krāsu pārbaudi sk. 2.8).

Ja kāds parametrs nav norādīts, attiecīgo formatējumu nepārbauda.

#### **Funkcijai **IsFormattedBorder** lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<code>IsFormattedBorder(„A3:E3”, „horizontālā augšējā”, „dubulta”, „Melns”) &amp; IsFormattedBorder(„A3:E3”, „horizontālā apakšējā”, „dubulta”, „Melns”) &amp; IsFormattedBorder(„A3:E3”, „vertikālā kreisā”, „dubulta”, „Melns”) &amp; IsFormattedBorder(„A3:E3”, „vertikālā labā”, „dubulta”, „Melns”) &amp; IsFormattedBorder(„A3:E3”, „vertikālā vidējā”, „vienkārša”, „Melns”)</code>	Tabulas virsraksta rindas noformējums: ārpusē dubults rāmis, iekšpusē vienkāršs rāmis.

### 3.11. Funkcija *IsFormattedPattern*

Funkcija *IsFormattedPattern* pārbauda, vai šūnām ir norādītais tonējums (fona krāsa un raksts) Excel izvēlnes „Format Cells” sadaļas „Patterns” izpratnē.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. fona krāsa: balts, melns, pelēks, sarkans, zaļš, zils (par krāsu pārbaudi sk. 2.8).

Pārbaudīta tiek tikai fona krāsa - šablons un šablona krāsa šajā sistēmas Tests versijā pārbaudīta netiek.

#### Funkcijai *IsFormattedPattern* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsFormattedPattern</i> („A3:E3”, „Pelēks”)	Vai kolonu virsraksti A3:E3 ir iekrāsoti pelēki?
2.	<i>IsFormattedPattern</i> („A1:E9”, „Balts”)	Formula pieprasa, lai iekrāsojums nebūtu lietots.

### 3.12. Funkcija *IsFormattedProtected*

Funkcija *IsFormattedProtected* pārbauda, vai šūnas ir slēgtas un ar paslēptām formulām Excel izvēlnes „Format Cells” sadaļas „Protection” izpratnē.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. locked: „Jā” vai „Nē” (vai šūnas ir slēgtas),
3. protected: „Jā” vai „Nē” (vai ir paslēptas formulas).

Otro un trešo argumentus var nenorādīt, tādā gadījumā attiecīgā pārbaude netiek veikta.

#### Funkcijai *IsFormattedProtected* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsFormattedProtected</i> („E4:E9”, ; „Jā”)	Vai formulas kolonā E4:E9 ir aizsargātas? Otrais parametrs ir izlaists, tāpēc pārbaude, vai šūnas ir slēgtas, netiek izdarīta.

### **3.13. Funkcija IsRowHeight**

Funkcija IsRowHeight pārbauda, vai šūnām ir norādītais augstums.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. augstums punktos.

Augstumam tiek pieļauta kļūda +/-2%. Norādītais augstums attiecas uz visām rindām, kas krusto norādīto apgabalu.

#### **Funkcijai IsRowHeight lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsRowHeight(„A1:E9”, 12,75)	Vai rindām ir noklusētais augstums (10. lieluma fontam). Mūsu 3 piemērā atbilde būs „Nē”, jo pirmā rinda ir lielākā fontā, un tāpēc augstāka.

### **3.14. Funkcija IsColumnWidth**

Funkcija IsColumnWidth pārbauda, vai šūnām ir norādītais platums.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. platums (standartfonta burtu platuma vienībās).

Platumam tiek pieļauta kļūda +/-2%. Norādītais platums attiecas uz visām kolonām, kas krusto norādīto apgabalu.

#### **Funkcijai IsColumnWidth lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsColumnWidth(„C:C”, 5,56)	Vai kolonas C platums ir 5,56?

### **3.15. Funkcija *IsColumnWideEnough***

Funkcija *IsColumnWideEnough* pārbauda, vai šūnām ir satura parādīšanai nepieciešamais platums.

Funkcijai ir viens arguments:

1. apgabals.

Funkcija pārbauda, vai visu apgabalā esošo šūnu platums ir pietiekams, lai parādītu šūnu saturu (netiek rādīta zīme #####, un ir vismaz „Auto fit” platums). Platumam nav jābūt vienādam visām šūnām (kolonām).

#### **Funkcijai *IsCellWideEnough* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsColumnWideEnough</i> („A4:E9”)	Mūsu 3.2.9 piemērā visai tabulai A1:E9 pārbaude būtu neveiksmīga, jo šūna A1 ir šaurāka par „Auto fit”.

### **3.16. Funkcija *IsRowHighEnough***

Funkcija *IsRowHighEnough* pārbauda, vai šūnām ir satura parādīšanai nepieciešamais augstums.

Funkcijai norādītais ir viens arguments:

1. apgabals.

Funkcija pārbauda, vai visu apgabalā esošo šūnu augstums ir pietiekams, lai parādītu šūnu saturu (ir vismaz „Auto fit” augstums). Augstumam nav jābūt vienādam visām šūnām (rindām).

#### **Funkcijai *IsCellHighEnough* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsRowHighEnough</i> („A1:E9”)	

### 3.17. Funkcija IsCellNumber

Funkcija IsCellNumber pārbauda, vai šūnas satur vajadzīgos skaitļus.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. vērtības – skaitļi.

Funkcijas otrajā argumentā tiek norādīti tik daudzi skaitļi, cik pirmajā argumentā norādītajā apgabalā ir šūnu. Apgabala šūnu saturam, apskatot tās pa rindām, sākot no kreisā augšējā stūra, ir jāsakrīt ar skaitļiem.

Ja šūnā ir ierakstīta formula, funkcija IsCellNumber salīdzina formulas vērtību, nevis pašu formulu.

Ja kādā no pārbaudāmajām šūnām nav skaitliska vērtība, funkcija uzreiz atbild "Nē". Funkcija atšķir skaitļus no teksta šūnām. Tā skaitlis 1,2 nesakrīt ar tekstu šūnā „1,2”, kaut vizuāli vērtības var izskatīties vienādas.

#### Funkcijai IsCellNumber lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsCellNumber(„A4:A8”, „1; 2; 3; 4; 5”)	Pārbauda, vai kolonā A4:A8 pareizi ierakstīti numuri pēc kārtas. Pārbaudot nav nozīmes, vai numuri izrēķinātu ar formulu, vai ierakstīti „ar roku”.
2.	IsCellNumber(„C4”, „&1”)	Šī formula pārbauda uzdevumu ar parametru: „Ierakstiet šūnā C4 skaitli &1”.
3.	IsCellNumber(„E9”, „36,15”)	Mūsu piemērā šūna E9 satur ar formulu izrēķinātu kolonas summu („Kopā:”). Ja uzdevums ir bijis: „Ievadīt šūnā E9 formulu summas aprēķināšanai”, mums ir grūtības pārbaudīt rezultātu, jo pareizā formula var izskatīties dažādi (piemēram, „=SUM(E4:E8)” vai „=E4+E5+E6+E7+E8”). Tomēr, mēs varam pārbaudīt formulas izrēķināto summu, un, ja summējamo skaitļu ir pietiekami daudz, tad studentam bez formulas sasummēt nebūs iespējams.

### 3.18. Funkcija IsCellText

Funkcija IsCellText pārbauda, vai šūnas satur vajadzīgo tekstu.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. tekstuālas vērtības.

Funkcija darbojas tāpat kā IsCellNumber, tikai ar tekstuālām vērtībām. Otrajā argumentā tekstuālās vērtības atdala ar semikolu, tāpēc semikols nevar būt tekstuālās vērtības daļa.

#### Funkcijai IsCellText lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsCellText(„B4:B8”, „Pildspalva”, „Zīmulis”, „Piezīmju papīrs”, „Klade”, „Lineāls”)	Pārbauda, vai kolonā B4:B8 pareizi ierakstīti preču nosaukumi.
2.	IsCellText(„C5”, „0,72”)	Šī formula mūsu piemērā atbildēs „Nē”, jo šūnā C5 ir ierakstīts skaitlis (un tad jau vienalga, kāds skaitlis).
3.	IsCellText(„E9”, „=SUM(E4:E8)”)	Šī formula mūsu piemērā atbildēs „Nē”, jo netiks salīdzināta formula, bet gan tās rezultāts.

### 3.19. Funkcija IsCellDate

Funkcija IsCellDate pārbauda, vai šūnas satur vajadzīgo datumu.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. datumi.

Funkcija darbojas tāpat kā IsCellNumber, tikai ar datumiem. Datumus norāda formātā dd.mm.yyyy

#### Funkcijai IsCellDate lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsCellDate(„B3”, „31.12.2007”)	

### 3.20. Funkcija IsCellFormula

Funkcija IsCellFormula pārbauda, vai šūnas satur vajadzīgo formulu.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. formula.

Argumentus „formula” var arī nenorādīt, tādā gadījumā formula pārbauda, vai šūnās ir ierakstīta jebkāda formula. Ja šūnā ir ierakstīta vērtība, nevis formula, funkcija atbild "Nē". Pat triviāla formula tiek atšķirta no vērtības (piemēram "=12" nav 12).

Formulas tiek salīdzinātas kā simbolu rindas (lielie un mazie burti tiek uzskatīti par vienādiem), tāpēc funkcija IsCellFormula atzīs par atšķirīgām tekstuāli dažādas formulas, kas veic vienu uzdevumu. Par dažādām, piemēram, tiks atzītas formulas "=E4+E5+E6", "=E6+E5+E4" un "=SUM(E4:E6)". Šī iemesla dēļ dažkārt labāk pārbaudīt, vai šūnā vispār ir formula, un vai izrēķinātā vērtība ir tāda, kā paredzēts.

#### Funkcijai IsCellFormula lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsCellFormula(„E9”, „=SUM(E4:E8)”)	Varbūt ne ideāls, bet pieņemams pārbaudes veids, jo šajā gadījumā visas iedomājamās alternatīvas (kā "= E4+E5+E6+E7+E8") ir ļoti sliktā stila paraugi. Ja summējamo šūnu ir vēl vairāk, tad formulai saprātīgu alternatīvu nav.
2.	IsCellFormula(„E9”) & IsCellNumber("E9", 36,15) & !IsCellFormula(„E9”, "=36,15")	Korektāks pārbaudes veids. Grūti iedomāties, kāda cita formula, neskaitot triviālo, vēl varētu dot pareizo skaitlisko rezultātu.
3.	IsCellFormula(„A1", "=SIN(A2)")	Formula, kuru citādi pierakstīt nav iespējams (ja neskaita tīšas ļaunprātības).



### 3.21. Funkcija **IsCellFormulaSubstring**

Funkcija **IsCellFormulaSubstring** pārbauda, vai šūnās ir ierakstītas formulas, kurās kā fragmenti ietilpst norādītie teksts.

Funkcijai ir sekojoši argumenti:

1. apgabals,
2. teksts.

Funkcija darbojas tāpat, kā **IsCellFormula** (sk. 3.20), tikai neprasa pilnīgu šūnā ierakstītās formulas un argumenta sakritību, bet tikai to, lai formula jebkurā vietā saturētu argumentā norādīto tekstu.

#### Funkcijai **IsCellFormulaSubstring** lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsCellFormulaSubstring(„E4”, „\$B\$2”) &amp; IsCellNumber("E4", 128)</code>	Ja uzdevumā prasīts kas tāds, kas pēc būtības prasa šūnu absolūto adresāciju, šādi mēs varam pārbaudīt, vai absolūtā adresācija vispār ir lietota. Tā kā formulu ekvivalence dēļ visu formulu mēs salīdzināt nevaram, tad tiek papildus pārbaudīts, vai aprēķina rezultāts ir pareizs.

### 3.22. Funkcija *IsWorksheetHeader*

Funkcija *IsWorksheetHeader* pārbauda, vai darba lapai ir norādīta vajadzīgā galvene.

Funkcijai ir sekojoši argumenti:

1. darba lapas vārds,
2. galvenes kreisā daļa,
3. galvenes centrālā daļa,
4. galvenes labā daļa.

Argumenti 2-4 atbilst Excel loga Custom Header laukiem Left Section, Center Section un Right Section. Argumentos tiek norādīts tikai attiecīgās galvenes daļas teksts (formatējums netiek pārbaudīts). Argumentos var norādīt arī speciālos simbolus, piemēram, "&[Page]. lappuse".

#### Funkcijai *IsWorksheetHeader* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsWorksheetHeader(„Izdevumi”, „Izdevumi kancelejas preču iegādei”, "" "&amp;[Page]. lappuse ")</code>	Galvenei norādīta kreisā un labā daļa, centrālajai daļai jābūt tukšai.
2.	<code>IsWorksheetHeader(„Izdevumi”, „”, "&amp;1", "&amp;[Page]")</code>	Uzdevumā ar parametru varētu būt prasīts: "Noformējiet galveni tā, lai labajā malā tiktu parādīts lappuses numurs, bet centrā - teksts "&1"". Pārbaudot &1 tiek aizstāts ar parametra vērtību, bet &[Page], saprotams, ne ar ko netiek aizstāts.

### 3.23. Funkcija *IsWorksheeFooter*

Funkcija *IsWorksheeFooter* pārbauda, vai darba lapai ir norādīta vajadzīgā kājene.

Funkcijai ir sekojoši argumenti:

1. darba lapas vārds,
2. kājenes kreisā daļa,
3. kājenes centrālā daļa,
4. kājenes labā daļa.

Funkcija darbojas tāpat, kā *IsWorksheeHeader* (sk. 3.22), tikai pārbauda kājeni.

### 3.24. Funkcija IsCellFormulaEQ

Funkcija IsCellFormulaEQ pārbauda, vai šūnās ir ierakstītās formulas ir vienādas ar precizitāti līdz šūnu relatīvajai adresācijai.

Funkcijai ir viens arguments:

1. apgabals.

Funkcijas veikto pārbaudi varam iedomāties tā, ka visas šūnas pēc kārtas ar Ctrl-C un Ctrl-V tiek pārkopētas uz vienu šūnu (piemēram, apgabala kreiso augšējo šūnu). Ja formulās tiek lietotas formulas ar relatīvo adresāciju, pārkopēšanas brīdī tās izmainās. Ja pārkopēšanas rezultāts vienmēr būs viena un tā pati formula, tad sākotnējās formulas būs bijušas ekvivalentas un atšķirušās tikai relatīvās adresācijas dēļ. Funkcija tādā gadījumā atbildēs "Jā". Ja kāda formula pēc pārkopēšanas atšķirsies no citām, funkcija atbildēs "Nē".

Ar funkciju var pārbaudīt, piemēram, vai students prot ar peli izplatīt formulu pa visu tabulas kolonu.

#### Funkcijai IsCellFormulaEQ lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsCellFormula(„E4”, "C4*D4") & IsCellFormulaEQ („E4:E8”)	Tiek pārbaudīts, vai šūnā E4 ir pareizā formula, un vai visas pārējās formulas ir ekvivalentas.

### 3.25. Funkcija IsCellEmpty

Funkcija IsCellEmpty pārbauda, vai norādītās šūnās ir tukšas - bez formulām un bez vērtībām.

Funkcijai ir viens arguments:

1. apgabals.

#### Funkcijai IsCellEmpty lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsCellEmpty(„A10:IV65536”) & IsCellEmpty(„F1:IV9”)	Formula pārbauda, vai ārpus tabulas A1:E9 šūnās nekas nav ierakstīts ("ieķēpāts"). Šāda pārbaude darbosies lēni lielā šūnu skaita dēļ.
2.	IsCellEmpty(„A10:O48”) & IsCellEmpty(„F1:O9”)	Tā pati pārbaude, tikai praktiskāka. Ja kaut kas nejauši būs ierakstīts, tad visdrīzāk jau izklājlapas sākumā.

### 3.26. Funkcija IsFrozen

Funkcija IsFrozen pārbauda, vai tieši aiz norādītās rindiņas ir iesaldēšanas robeža (View -> Freeze panes).

Funkcijai ir viens arguments:

1. rindiņas numurs - skaitlis.

#### Funkcijai IsFrozen lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsFrozen(3)	Mūsu piemērā loģiski būtu "iesaldēt" pirmās trīs rindiņas.

### 3.27. Funkcija IsRangeAscending

Funkcija IsRangeAscending pārbauda, vai norādītās šūnās ir sakārtotas vērtību pieaugšanas secībā.

Funkcijai ir viens arguments:

1. apgabals,
2. kārtība.

Apgabals parasti ir viena kolons tabulā. Ja apgabals sarežģītāks, tad ar argumentu „kārtība” norāda, kādā kārtībā jāapstaigā apgabala šūnas –pa rindām vai pa kolonām. Ja kārtība norādīta „pa rindām”, tad šūnas apgabalā tiek salīdzinātas, sākot no apgabala kreisā augšējā stūra, vispirms ejot pa rindu pa labi, un tad – uz leju, uz nākamo rindu.

Funkcija paredzēta, lai pārbaudītu, vai students prot lietot Excel izvēlni "Data -> Sort".

#### Funkcijai IsRangeAscending lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsRangeAscending(„A4:A8")	
2.	IsRangeAscending(„A1:E5")	Ar uzdevumu: "Sakārtot augošā secībā šūnas A1:A5" varētu pārbaudīt, vai students prot lietot taustiņus Ctrl-C un Ctrl-V, jo, ja šūnās ir ierakstīti pietiekami lieli skaitļi, citādi šis uzdevums praktiski nav izpildāms.

### **3.28. Funkcija *IsRangeDescending***

Funkcija *IsRangeDescending* pārbauda, vai norādītās šūnās ir sakārtotas vērtību dilšanas secībā.

Funkcijai ir viens arguments:

1. apgabals,
2. kārtība.

Funkcija darbojas tāpat, kā *IsRangeAscending* (sk.3.27).

### **3.29. Funkcija *IsFormulaValue***

Funkcija *IsFormulaValue* pārbauda, vai, ierakstot dotajā šūnā ierakstot norādīto formulu, tiktu iegūts vajadzīgais skaitlis.

Funkcijai ir sekojoši arguments:

1. šūna,
2. formula,
3. vērtība (skaitlis).

Funkciju var izmantot, lai viegli pārlicinātos, vai tabulas fragments ir aptuveni tāds, kā vajadzīgs.

#### **Funkcijai *IsFormulaValue* lietošanas piemērs**

Pieņemsim, ka uzdevumā studentam bija jāaizpilda šūnas A1:A999 ar skaitļiem 1, 2, 3... 999. Aizpildāmo šūnu skaits ir liels, lai piespiestu studentu ierakstīt pirmos divus skaitļus, bet pārējos „aizvilkt ar peli”. Diemžēl lielais šūnu skaits liedz pārbaudīt katru šūnu, tomēr ar *IsFormulaValue*(B1, „SUM(A1:A999)”, 499500) mēs varam pārbaudīt ievadīto skaitļu summu, un iegūt pārlicību, ka uzdevums izpildīts.

### 3.30. Funkcija IsCellFormulaValue

Funkcija IsCellFormulaValue dod iespēju uz piemēriem pārbaudīt, vai studenta ierakstītā formula darbojas pareizi.

Funkcijai ir sekojoši argumenti:

1. šūna 1,
2. šūna 2,
3. vērtība.

Formāli funkcija darbojas tā, ka formula no šūnas "šūna 1" ar Ctrl-C un Ctrl-V palīdzību tiek pārcelta uz šūnu "šūna 2" (pie tam relatīvās adresācijas dēļ formula var izmainīties). Jaunajā atrašanās vietā formula izrēķina kaut kādu vērtību, kas atkarīga no apkārtējām šūnām. Ja šī vērtība sakrīt ar 3. argumentā norādīto, funkcija atbild "Jā", ja atšķiras, "Nē".

Funkcija paredzēta, lai pārbaudītu uzdevumus, kuros studentam jāizveido sarežģīta formula ar absolūto un relatīvo adresāciju, vairākām funkcijām u.tml. Formulu ekvivalences dēļ šādu formulu mēs nevaram pārbaudīt tekstuāli ar IsCellFoemula. Ar IsCellNumber mēs varam pārbaudīt, vai formula dod skaitliski pareizu rezultātu, bet pareizo atbildi students var būt izrēķinājis galvā un ierakstījis bez formulas (vai izrēķinājis ar nepietiekami universālu formulu).

Funkcija IsCellFormulaValue dod mums iespēju izveidot speciālu testēšanas darba lapu, kuru mēs studentam nerādām, bet izmantojam tikai viņa izveidoto formulu pārbaudei. Ar IsCellFormulaValue mēs tad varam studenta izveidoto formulu "pārnest" uz šo testa lapu, un pārbaudīt, kādas vērtības tā rēķina atšķirīgā kontekstā.

#### Funkcijai IsCellFormulaValue lietošanas piemērs

Pieņemsim, ka sekojošajā tabulā mēs esam studentam lūguši aizpildīt šūnas E6:E10 ar formulām summas ar atlaidi aprēķināšanai (mēs sagaidām aptuveni šādu formulu " $=C6*D6*(1-SC\$3)$ ") un aprēķināt summu "Kopā:".

Izdevumi kancelejas preču iegādei				
Atlaide:		5%		
Nr.	Prece	Cena bez atlaide	Skaitis	Summa ar atlaidi
3.	Pildspalva	0,72	10	6,84
4.	Zīmulis	0,50	20	9,50
5.	Piezīmju papīrs	1,30	5	6,18
6.	Klade	1,85	5	8,79
7.	Lineāls	3,20	1	3,04
			<b>Kopā:</b>	<b>34,34</b>

Mums ir grūti pārliecināties par formulas pareizību, jo students var būt ierakstījis formulā SC\$3 kā konstanti 0,05, var būt SC\$3 vietā lietojis C3, var būt =SUM(E6:E11) ierakstījis =SUM(E1:E11) un rezultāts vienalga būs skaitliski pareizs (Visas trīs ir tipiskas iesācēju kļūdas).

Lai uzdevumu pārbaudītu, mēs izveidojam speciālu darba lapu (sauksim to "Tests") ar citiem datiem.

<b>Izdevumi kancelejas preču iegādei</b>				
Atlaide:		3%		
<b>Nr.</b>	<b>Prece</b>	<b>Cena bez atlaide</b>	<b>Skaitis</b>	<b>Summa ar atlaidi</b>
3.	Pildspalva	0,72	10	6,98
4.	Zīmulis	0,50	20	9,70
5.	Piezīmju papīrs	1,30	5	6,31
6.	Klade	1,85	5	8,97
7.	Lineāls	3,20	1	3,10
8.	Adata	0,01	1	0,01
			<b>Kopā:</b>	<b>35,08</b>

Tagad uzdevumu mēs varam pārbaudīt ar divām funkcijām:

IsCellFormulaValue("E6", "Tests!E6", 6,98) &

IsCellFormulaValue("E11", "Tests!E12", 35,08) &

### 3.31. Funkcija IsChart

Funkcija IsChart pārbauda, vai norādītajā darba lapā ir tieši viens grafiks (chart).

Funkcijai ir viens arguments:

1. darba lapas vārds.

Šī un sekojošās funkcijas, kas attiecas uz grafikiem, dod visai ierobežotas iespējas uzdevumu pārbaudei, tāpēc tās ir jākombinē ar netiešiem jautājumiem (arī teorētiskiem jautājumiem un jautājumiem, kuros izmantoti attēli). Piemēram:

1. Kāda ir tāda-un-tāda parametra vērtība iepriekš sagatavotā grafikā (ar frāzi atbildams jautājums, kurā tiek izmantots Excel fails).?
2. Kurā laika periodā valūtas kursa svārstības ir bijušas lielākās (Uzdevumā tiek dots liels datu apjoms, ko bez grafika izanalizēt ir praktiski neiespējams)?
3. Kurai no divām kompānijām ienākumu izaugsmes prognoze ir optimistiskāka (Atkal ar lielu datu apjomu un izteiktu tendenci)?
4. Izveidojiet pēc dotajiem datiem līnijas (Line) tipa grafiku ar noklusētajiem uzstādījumiem. Kāda ir maksimālā vērtība, ko var attēlot uz y ass (Scale -> Maximum)?

Tā kā funkciju klāsts ir ierobežots, uzdevumi bieži jāveido tādi, ka students izdara izmaiņas jau esošā grafikā, jo pilnvērtīgi pārbaudīt jauna grafika izveidi nav iespējams.

#### Funkcijai IsChart lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsChart(„Sheet1“)	Uzdevums varētu būt, izdzēst vienu grafiku no diviem.

### 3.32. Funkcija IsChartType

Funkcija IsChartType pārbauda, vai norādītajā darba lapā ir tieši viens grafiks (chart), un tam ir norādītais tips.

Funkcijai ir sekojoši argumenti:

1. darba lapas vārds.
2. grafika tips.

Funkcija pārbauda, ka darba lapā ir tieši viens grafiks, un šī grafika tips ir tāds, kā norādīts. Grafika tips atbilst Excel izvēlnes „Standard Types” sadaļai „Chart type”. Tipi drīkst būt: ColumnNormal, BarNormal, PieNormal, Line. Apakštipi netiek pārbaudīti.

#### Funkcijai IsChartType lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsChartType(„Sheet1”, "ColumnNormal")	Uzdevums varētu būt nomainīt tipu esošam grafikam.



### 3.33. Funkcija *IsChartData*

Funkcija *IsChartData* pārbauda, vai norādītajā darba lapā ir tieši viens grafiks (chart), un vai šim grafikam dati tiek ņemti no norādītā apgabala.

Funkcijai ir sekojoši argumenti:

1. darba lapas vārds.
2. apgabals.

Datu apgabals atbilst MS Chart izvēlnei Data Range.

#### Funkcijai *IsChartData* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsChartData(„Izdevumi”, "E4:E9")</code>	Uzdevums varētu būt izlabot esošu grafiku tā, lai dati tiktu ņemti nevis no kolonas "Cena", bet no kolonas "Summa".

### 3.34. Funkcija *IsChartTitle*

Funkcija *IsChartTitle* pārbauda, vai norādītajā darba lapā ir tieši viens grafiks (chart), un vai šim grafikam ir vajadzīgais virsraksts.

Funkcijai ir sekojoši argumenti:

1. darba lapas vārds.
2. virsraksta teksts.

Virsraksta teksts atbilst MS Chart izvēlnes Chart Options sadaļas Titles laukam Chart title.

#### Funkcijai *IsChartTitle* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsChartTitle(„Izdevumi”, "Kancelejas preču izmaksas")</code>	Uzdevums varētu būt: "Izlabot esoša grafika virsrakstu uz "Kancelejas preču izmaksas"".
2.	<code>IsChartTitle(„Izdevumi”, "&amp;1")</code>	Uzdevums varētu būt: "Izlabot esoša grafika virsrakstu uz "&1"".
3.	<code>IsChartTitle(„Izdevumi”, "&amp;0")</code>	Varētu būt ar frāzi atbildams jautājums, kas izmanto Excel failu: "Kāds ir grafika virsraksts?". Grafiks varētu būt noformēts tā, ka virsraksts uzreiz nav redzams.

### **3.35. Funkcija IsChartValueColor**

Funkcija IsChartValueColor pārbauda, vai norādītajā darba lapā ir tieši viens grafiks (chart), un vai šajā grafikā norādītā vērtība ir attēlota norādītajā krāsā.

Funkcijai ir sekojoši argumenti:

1. darba lapas vārds,
2. vērtības numurs
3. krāsa.

Krāsa atbilst MS Chart izvēlnes Format data series sadaļas Patterns laukam Color. Krāsa tiek norādīta tāpat, kā Word funkcijā IsShadingColor (sk. 2.9). Ar vienu funkcijas izsaukumu var pārbaudīt tikai viena rādītāja (datu elementa) krāsu - nākamais datu elements jāpārbauda ar citu funkcijas izsaukumu. Datu elementi tiek numurēti izvēlnes Source data sadaļas Series loga Series kārtībā.

#### **Funkcijai IsChartValueColor lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsChartValueColor(„Sheet1”, 1, "Sarkans") & IsChartValueColor(„Sheet1”, 2, "Zaļš")	Uzdevums varētu būt izlabot esoša grafika krāsas.

## 4. Pārbaudes funkcijas PowerPoint un Impress vidē

Praktiskajiem uzdevumiem PowerPoint un Impress vidē studenta darba rezultāts ir prezentācija, fails ar paplašinājumu .ppt. Šī faila pareizība tiek pārbaudīta ar zemāk aprakstītajām pārbaudes funkcijām.

Funkcijās tiek izmantots, ka prezentācija sastāv no slaidiem, un ka slaidi ir numurēti ar skaitļiem 1, 2, 3... Vairumam funkciju pirmais arguments norāda, uz kuru slaidu attiecas pārbaude. Piemēram, funkcija IsSlideTitle(2, "Kompānijas struktūra") pārbauda virsrakstu otrajam slaidam prezentācijā.

Pārbaudes funkcijas, kurās tam ir saturīga jēga, var pārbaudīt ne tikai prezentācijas slaidu, bet arī Slide Master (sk. izvēlni View/ Slide Master). Tādā gadījumā kā pirmo argumentu, slaida numuru, norāda 0, bet pārējos argumentus norāda tāpat, kā pārbaudot parasto slaidu. Piemēram, IsSlideBackground(1, "Pelēks") pārbauda pirmā slaida fona krāsu, bet IsSlideBackground(0, "Pelēks") - Slide Master fona krāsu. Visās šajās funkcijās tiek pieņemts, ka prezentācijā ir tikai viens Slide Master. Ja to ir vairāki, tad pārbaudes rezultāts vienmēr ir negatīvs.

Funkcijās tiek uzskatīts, ka katrs slaidis sastāv no elementiem: teksta rāmjiem, attēliem, diagrammām..Elements atbilst tam, kas rodas vai nu no Placeholder (Slide Layout izpratnē), vai arī tiek ielikts slaidā ar izvēlni Insert (Text box, Picture, Diagramm). Elementi tiek numurēti ar skaitļiem 0, 1, 2... Numurs 0 ir tam teksta rāmim, kuru View Outline uztver kā virsrakstu (Title). Pārējie elementi tiek numurēti ar numuriem sākot no 1. Daudzās funkcijas pirmais arguments ir slaida numurs, bet otrais - elementa numurs. Piemēram, IsTextBox(1, 2) pārbauda, vai pirmā slaida otrais elements ir teksta rāmis (zemāk dotajā piemērā tas ir teksta rāmis "Kompānija "Jānis un dēli"").

Elementu numurus piešķir PowerPoint brīdī, kad elements tiek izveidots, un uzdevumu sagatavotājam nav iespējas šos numurus mainīt. Numurus var apskatīt, piemēram, piešķirot elementam animāciju (Slide show/ Custom animation), un, kad numurs apskatīts, animāciju atkal noņemot. Parasti gan tas nav nepieciešams, jo elementi tiek numurēti to izveidošanas secībā.

Elementu numuri nav saistīti ar to grafisko novietojumu slaidā - teorētiski dulltais elements var atrasties arī slaida labajā apakšējā stūrī.

Visās funkcijās, kur notiek teksta salīdzināšana vai meklēšana, tā notiek studentam maksimāli labvēlīgi: nav case sensitive, nebūtiskie tukšumi un citi formatējuma elementi tiek izlaisti. Izņēmums ir tikai funkcijā IsSlideTitle, kurā teksta salīdzināšana ir precīza.

Visas funkcijas, kuru argumentos norādāms slaida numurs un elementa numurs, atbild „Nē”, ja slaida ar norādīto numuru prezentācijā nav, vai arī ja elementa ar šādu numuru norādītajā slaidā nav.

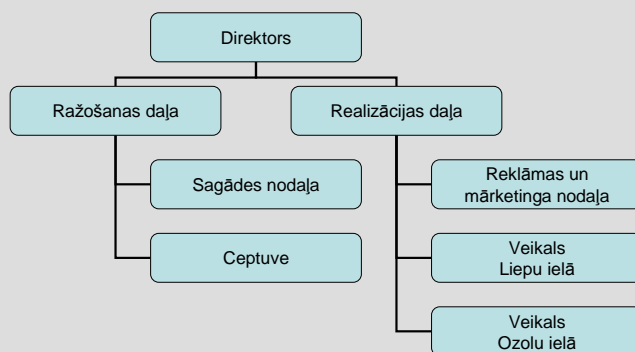
Funkcijas atbild "Nē" arī tad, ja elementa tips neatbilst funkcijā prasītajam. Piemēram, ContainsText(2, 1, "Direktors") atbildēs "Nē", ja otrā slaida pirmais elements būs nevis teksta rāmis, bet diagramma, jo funkcija ContainsText nemeklē tekstu diagrammās un grafikos.

Turpmākajās nodaļās, aprakstot pārbaudes funkcijas, piemēros parasti tiks izmantota sekojoša prezentācija.

## Pasākumi maizes pārdošanas apjoma palielināšanai

Kompānija Jānis un dēli

### Kompānijas struktūra



### Plānotie pārdošanas veicināšanas pasākumi

- Slēgt abus kompānijas veikalus,
- Uzsākt maizes piegādi lielveikalu ķēdei "Pēteris un dēli",
- Uzsākt konditorijas izstrādājumu piegādi kafejnīcai "Līga un meitas",
- Palielināt konditorijas izstrādājumu īpatsvaru ražošanā un pārdošanā.

#### **4.1. Funkcija *IsDesignTemplate***

Funkcija *IsDesignTemplate* pārbauda, vai prezentācija ir veidota ar norādīto šablonu (sk. izvēlni *Format/ Slide design*).

Funkcijai ir viens arguments:

1. šablona vārds.

Funkcija pārbauda visai prezentācijai norādīto šablonu, neņemot vērā, ka atsevišķi slaidi prezentācijā var būt noformēti ar atšķirīgu šablonu. Prezentācijai norādīto šablonu PowerPoint vidē var uzzināt ar izvēlni *File/ Properties*.

##### **Funkcijai *IsDesignTemplate* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsDesignTemplate</i> („Default Design”)	Vai prezentācijai lietots noklusētais šablons?

#### **4.2. Funkcija *IsOption***

Funkcija *IsOption* pārbauda, vai prezentācijai ir uzstādīta norādītā opcija.

Funkcijai ir sekojoši argumenti:

1. opcijas nosaukums: "Zoom" vai "View",
2. opcijas vērtība.

Ja opcijas nosaukums ir "Zoom", tad vērtība ir apskates mērogs procentos - skaitlis, kura vērtība tiek pārbaudīta ar precizitāti +/-10%. Tiek pārbaudīts mērogs PowerPoint pamat logam.

Ja opcijas nosaukums ir "View", tad vērtība var būt vai nu "Normal", vai "Slide sorter".

##### **Funkcijai *IsOption* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsOption</i> („Zoom", 80)	Vai mērogs ir 80%?

### **4.3. Funkcija *IsPageSetup***

Funkcija *IsPageSetup* pārbauda, vai prezentācijai ir uzstādīts vajadzīgais papīra lapas veids.

Funkcijai ir sekojoši argumenti:

1. opcijas nosaukums: "Orientation" (šajā sistēmas Tests versijā citas opcijas netiek pārbaudītas),
2. opcijas vērtība: "Portrait" vai "Landscape".

#### **Funkcijai *IsPageSetup* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsPageSetup</i> („Orientation”, "Landscape")	Vai uzstādīts Landscape papīra formāts?

### **4.4. Funkcija *IsSlides***

Funkcija *IsSlides* pārbauda, vai prezentācijai ir norādītais slaidu skaits.

Funkcijai ir viens arguments:

1. slaidu skaits.

Tiek skaitīti visi prezentācijas slaidi, ieskaitot arī paslēptos. Slide Master netiek uzskatīts par atsevišķu slaidu.

#### **Funkcijai *IsSlides* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsSlides</i> (3)	Vai prezentācijā ir tieši trīs slaidi?

## 4.5. Funkcija **IsSlideTitle**

Funkcija **IsSlideTitle** pārbauda, vai slaidam ir norādītais virsraksts.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. slaida virsraksts.

Par slaida virsrakstu tiek uzskatīts teksts, kas atrodas teksta rāmī ar numuru 0. Parasti šis teksta rāmis rodas no placeholder "Click to add title". Ja slaidam nav virsraksta (piemēram, tajā nav vispār neviena elementa), funkcija atbild "Nē".

Teorētiski prezentācijā var būt vairāki slaidi ar vienu virsrakstu, tomēr, sagatavojot uzdevumus, no šādas situācijas būtu jāizvairās, ja virsrakstu ir plānots lietot slaidu identifikācijai.

Slaida virsrakstam ar funkcijā norādīto argumentu ir jāsakrīt precīzi - tiek ņemti vērā arī nenozīmīgi tukšumi un paragrāfa zīmes. Lielie un mazie burti tiek uzskatīti par dažādiem. Šī ir vienīgā starp PowerPoint funkcijām, kurā teksta sakrišanai jābūt precīzai - visās citās šādas nelielas atšķirības tiek tulkotas par labu studentam.

### Funkcijai **IsSlideTitle** lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsSlideTitle(2, „&amp;1”)</code>	Ir bijis parametrizēts uzdevums: "Nomainiet otrā slaida virsrakstu uz &1"
2.	<code>IsSlideTitle(&amp;1, „Pārdošanas veicināšana”)</code>	Ir bijis parametrizēts uzdevums: "Nomainiet slaidam numur &1 virsrakstu uz „Pārdošanas veicināšana”" (ar &1 vērtībām 1 un 3)
3.	<code>IsSlideTitle(2, „Plānotie pārdošanas veicināšanas pasākumi”) &amp;</code> <code>IsSlideTitle(3, „Kompānijas struktūra”)</code>	Ir bijis uzdevums: "Apmainiet vietām otro un trešo slaidu". Virsraksts tiek lietots slaida identifikācijai
4.	<code>IsSlideTitle(1, „Pasākumi maizes pārdošanas apjoma palielināšanai”) &amp;</code> <code>IsSlideTitle(2, „Kompānijas struktūra”) &amp;</code> <code>IsSlides(2)</code>	Ir bijis uzdevums: "Izdzēst trešo slaidu". Tiek pārbaudīts, vai pirmie divi slaidi ir savā vietā, un vai slaidu skaits ir samazinājies par vienu.  Lielākā prezentācijā šāda veida pārbaude var būt apgrūtināša - īsāku pārbaudi šai situācijai sk. nodaļā 4.6

#### 4.6. Funkcija *IsNotSlideTitle*

Funkcija *IsNotSlideTitle* pārbauda, vai starp prezentācijas slaidiem nav neviena slaida ar norādīto virsrakstu.

Funkcijai ir sekojoši argumenti:

1. slaida virsraksts.

Funkcija atbild "Jā", ja prezentācijā *nav* slaida ar norādīto virsrakstu.

#### Funkcijai *IsNotSlideTitle* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsNotSlideTitle(„Plānotie pārdošanas veicināšanas pasākumi”) &amp; IsSlides(2)</code>	Ir bijis uzdevums: "Izdzēst trešo slaidu". Tiek pārbaudīts, vai izdzēšamā slaida tiešām nav, un vai slaidu skaits ir samazinājies par vienu.
2.	<code>IsNotSlideTitle(„&amp;1 ") &amp; IsSlides(2)</code>	Ir bijis parametrizēts uzdevums: "Izdzēst slaidu ar virsrakstu &1".

#### 4.7. Funkcija *IsSlideLayout*

Funkcija *IsSlideLayout* pārbauda, vai norādītajam slaidam ir vajadzīgais izklājuma formāts (sk. izvēlni *Format/ Slide Layout*).

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. izklājuma formāta nosaukums.

PowerPoint vidē izklājuma formāta nosaukums tiek rādīts kā iznirstošs teksts *Slide layout* logā (logā, no kura tiek paņemti izklājuma formāti). Izklājuma formāti var būt:

1. Blank slide,
2. Title only,
3. Title slide,
4. Title and object,
5. Title and 4 objects,
6. Title, text and object,
7. Custom.

#### Funkcijai *IsSlideLayout* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsSlideLayout(1, „Title slide")</code>	Vai pirmā slaida izklājuma formāts ir "Title Slide". Šī ir prezentācijas labā stila pārbaude, jo vajadzīgā izskata virsraksta slaidu var izveidot arī no cita izklājuma formāta, tomēr tādā gadījumā pazūd iespēja nomainīt noformējuma



		šablonu visai prezentācijai.
--	--	------------------------------

#### **4.8. Funkcija *IsSlideBackground***

Funkcija *IsSlideBackground* pārbauda, vai norādītajam slaidam ir vajadzīgais fona krāsa.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. krāsa.

Funkcija var pārbaudīt kā parastus slaidus, tā arī Slide Master (pirmais arguments var būt 0).

Krāsa var būt: balts, melns, pelēks, sarkans, zaļš, zils. Tā tiek norādīta un salīdzināta tāpat, kā Word funkcijā *IsShadingColor* (sk. 3.2.8.9). Aizpildīšanas veidi (Fill effects) netiek pārbaudīti.

#### **Funkcijai *IsSlideBackground* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsSlideBackground</i> (1, „Pelēks”)	Vai pirmā slaida fons ir pelēks?
2.	<i>IsSlideBackground</i> (0, „Pelēks”)	Vai Slide Master fons ir pelēks?

#### **4.9. Funkcija *IsSlideHidden***

Funkcija *IsSlideHidden* pārbauda, vai norādītais slaidi ir paslēpts (sk. izvēlni Slide show/Hide slide).

Funkcijai ir viens arguments:

1. slaida numurs.

Neskatoties uz to, ka slaidi ir paslēpti, tas tiek ņemts vērā slaidu numerācijā, un to var pārbaudīt ar visām PowerPoint funkcijām.

#### **Funkcijai *IsSlideHidden* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsSlideHidden</i> (2)	Ir bijis uzdevums paslēpt slaidu „Kompānijas struktūra”
2.	! <i>IsSlideHidden</i> (12) & ! <i>IsSlideHidden</i> (93)	Ir bijis uzdevums noņemt paslēpšanas pazīmi visiem slaidiem prezentācijā. Ja prezentācija ir liela, tad šī uzdevuma izpildei ir nepieciešama prasme iezīmēt visus slaidus (Edit/ Select All slaidu pārskata logā)

#### **4.10. Funkcija IsNoteText**

Funkcija IsNoteText pārbauda, vai norādītajam slaidam prezentācijas veidotāja piezīmēs (logā, kas uz PowerPoint ekrāna ir apakšā, zem slaida loga) ir norādītais teksts un tikai tas.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. teksts.

Tiek pārbaudītas slaida piezīmes, kas PowerPoint ekrānā ir apakšā, zem slaida loga (nevis komentāri, kas attiecas uz atsevišķiem slaida elementiem).

Piezīmēs jābūt norādītajam tekstam un tikai tam. Teksts tiek salīdzināts studentam labvēlīgi (sk. 4).

#### **Funkcijai IsNoteText lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsNoteText(3, „Uzsvērt, ka slēgto veikalu darbinieki tiks nodarbināti ceptuvē”)	
2.	IsNoteText(3, ””)	Ir bijis uzdevums: "Izdzēst trešā slaida piezīmes"

#### **4.11. Funkcija IsSlideTransition**

Funkcija IsSlideTransition pārbauda, vai visiem slaidiem ir norādīta vajadzīgā slaidu pāreja (izvēlne Slide show/ Slide transition).

Funkcijai ir viens arguments:

1. slaidu pārejas nosaukums.

Pāreja var būt:

1. No transition,
2. Fade Thru Black,
3. Wipe Down,
4. Dissolve.

Visiem slaidiem prezentācijā jābūt lietotai vienai un tai pašai slaidu pārejai - tai, kura norādīta kā arguments. Slaidu pārejas parametri pārbaudīti netiek.

#### **Funkcijai IsSlideTransition lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsSlideTransition(„Fade Through Black”)	
2.	IsSlideTransition(„%1”)	Ir bijis parametrizēts uzdevums.

#### **4.12. Funkcija *IsSlidesNumbered***

Funkcija *IsSlidesNumbered* pārbauda, vai visiem slaidiem tiek rādīts slaida numurs (sk. izvēlni *Insert/ Slide number*).

Funkcijai argumentu nav.

##### **Funkcijai *IsSlidesNumbered* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsSlidesNumbered</i> ()	

#### **4.13. Funkcija *IsFooter***

Funkcija *IsFooter* pārbauda, vai visiem slaidiem ir norādīta vajadzīgais kājenes teksts (sk. izvēlni *View/ Header and footer/ Footer*).

Funkcijai ir sekojoši argumenti:

1. kājenes teksts.

##### **Funkcijai *IsFooter* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsFooter</i> („© Jānis un dēli“)	

#### **4.14. Funkcija *IsFooterDate***

Funkcija *IsFooterDate* pārbauda, vai visiem slaidiem kājenē tiek rādīts datums (sk. izvēlni *View/ Header and footer/ Date and time*).

Funkcijai argumentu nav.

Datuma izvēlnē jābūt norādītam "Update authomatically", valoda un kalendāra tips netiek pārbaudīti.

##### **Funkcijai *IsFooterDate* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsFooterDate</i> ()	

#### 4.15. Funkcija *IsElementPosition*

Funkcija *IsElementPosition* pārbauda, vai norādītais elements slaidā grafiski atrodas vajadzīgajā vietā.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. horizontālais attālums,
4. horizontālais atskaites punkts,
5. vertikālais attālums.
6. vertikālais atskaites punkts

Attālumi un atskaites punkti tiek interpretēti tieši tā, kā izvēlnes *Format placeholder* (vai *Format textbox*) sadaļā *Position*.

Horizontālais un vertikālais attālums ir attālums centimetros. Horizontālais un vertikālais atskaites punkts var būt vai nu "Top left corner", vai "Center". Attālums tiek mērīts no atskaites punkta līdz elementa kreisajam augšējam stūrim.

Attālumu pārbaudot tiek pieļauta kļūda +/-10% no slaida kopējā platuma (horizontālajam attālumam) vai augstuma (vertikālajam attālumam).

#### Funkcijai *IsElementPosition* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsElementPosition</i> (1, 2, 0, "Top left corner", 0, "Top left corner") & <i>IsPicture</i> (1, 2)	Ir bijis uzdevums prezentācijas pirmā slaida kreisajā augšējā stūrī ielikt kompānijas logo.
2.	<i>IsElementPosition</i> (0, 6, 0, "Top left corner", 0, "Top left corner") & <i>IsPicture</i> (1, 2)	Tas pats, ja logo jāieliek Slide Master (parasti Slide Master jau satur elementus 0-5).
3.	<i>IsElementPosition</i> (1, 2, 20, "Top left corner", 0, "Top left corner") & <i>IsPicture</i> (1, 2)	Tas pats, ja logo jāieliek labajā augšējā stūrī, pieņemot, ka logo attēls ir 5cm plats.
4.	<i>IsElementPosition</i> (1, 1, 0, "Top left corner", 0, "Top left corner") & <i>IsElementPosition</i> (1, 2, 12,7, "Top left corner", 0, "Top left corner") & <i>IsElementPosition</i> (1, 3, 0, "Top left corner", 9,5, "Top left corner") & <i>IsElementPosition</i> (1, 4, 12,7, "Top left corner", 9,5, "Top left corner")	Pārbaude uzdevumam, kurā no četriem saskaņotiem attēliem ¼ slaida lielumā jāsabīda viens attēls (objektu iezīmēšanas un pārvietošanas pārbaude). Tiek uzskatīts, ka parasta slaida izmērs ir aptuveni 25,4x19 (+/- 10% kļūda precizitāti neprasa).

#### 4.16. Funkcija *IsElementSize*

Funkcija *IsElementSize* pārbauda, vai norādītā slaida norādītajam elementam ir vajadzīgais izmērs.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. horizontālais izmērs,
4. vertikālais izmērs.

Izmērus norāda centimetros. Salīdzinot tiek pieļauta +/- 5% kļūda.

#### Funkcijai *IsElementSize* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsSlideLayout</i> (1, "Title Slide") & <i>IsElementSize</i> (1, 0, 21,6, 4,08) & <i>IsElementPosition</i> (1, 0, 1,91, "Top left corner", 5,92, "Top left corner")	Formula pārbauda, vai pirmais slaidis ir veidots ar "Title Slide" šablonu, un vai pēc tam virsraksta teksta rāmis nav pārvietots, samazināts vai palielināts (formulā norādīti šī teksta rāmja sākotnējie izmēri un novietojums).
2.	<i>IsElementSize</i> (1, 2, 5, 3) & <i>IsElementPosition</i> (1, 2, 0, "Top left corner", 0, "Top left corner") & <i>IsPicture</i> (1, 2)	Ir bijis uzdevums prezentācijas pirmā slaida kreisajā augšējā stūrī ielikt kompānijas logo. Pēc izmēriem 5x3cm mēs netieši varam spriest, ka ielikts pareizais attēls.
3.	<i>IsElementSize</i> (1, 2, 6, 4,5)	Ir bijis uzdevums prezentācijā jau esošam attēlam, par kuru uzdevuma sastādītājs zina, ka tā izmērs ir 9x12cm, mērogu nomainīt no 100% uz 50%.

#### **4.17. Funkcija *IsElementInFront***

Funkcija *IsElementInFront* pārbauda, vai no norādītā slaida diviem norādītajiem elementiem pirmais ir priekšā otrajam (izvēlnes *Order/ Bring to front* izpratnē).

Funkcijai ir sekojoši argumenti:

1. slaida numurs vai 0,
2. elementa numurs 1,
3. elementa numurs 2.

Funkcija neprasa, lai pirmais elements būtu priekšā visiem citiem elementiem šajā slaidā, tā tikai salīdzina divu elementu relatīvās pozīcijas.

#### **Funkcijai *IsElementInFront* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<code>IsElementInFront(1, 2, 1)</code>	Ir bijis uzdevums sameklēt mazu attēlu, kas paslēpts zem lielāka, un novietot to priekšā.
2.	<code>IsPicture(0, 6) &amp;</code> <code>IsElementInFront(0, 6, 0) &amp;</code> <code>IsElementInFront(0, 6, 1) &amp;</code> <code>IsElementInFront(0, 6, 2) &amp;</code> <code>IsElementInFront(0, 6, 3) &amp;</code> <code>IsElementInFront(0, 6, 4) &amp;</code> <code>IsElementInFront(0, 6, 5) &amp;</code> <code>IsElementSize(0, 6, 25,4, 19) &amp;</code> <code>IsElementPosition(0, 6, 0, "Top left corner", 0, "Top left corner")</code>	Ir bijis uzdevums ielikt Slide Master fona attēlu un mērogot to pa visu slaidu.

#### **4.18. Funkcija IsElementAnimation**

Funkcija IsElementAnimation pārbauda, vai norādītā slaida norādītajam elementam ir norādīta animācija.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs.

Šajā sistēmas Tests versijā tiek pārbaudīts tikai, vai animācija ir vispār – animācijas veids un parametri netiek pārbaudīti.

#### **Funkcijai IsElementAnimation lietošanas piemēri**

Nr.	Funkcijas lietojums	Komentārs
1.	IsElementAnimation(3, 1)	

#### **4.19. Funkcija ContainsTextBoxes**

Funkcija ContainsTextBoxes pārbauda, vai norādītajā slaidā ir tieši norādītais skits teksta rāmju.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. rāmju skaits.

Funkcija skaita tikai teksta rāmjus, tas, vai slaidā bez teksta rāmjiem ir vēl arī attēli, diagrammas vai citi elementi, rezultātu neietekmē.

#### **Funkcijai ContainsTextBoxes lietošanas piemēri**

Nr.	Funkcijas lietojums	Komentārs
1.	ContainsTextBoxes(2, 0) & ContainsOrgCharts(2, 1)	Ir bijis uzdevums izdzēst slaida virsrakstu (slaidā paliek otrs elements – diagramma).
2.	ContainsTextBoxes(1, 2) & ContainsText(1, 1, „Kompānija Jānis un dēli”)	Ir bijis uzdevums papildināt virsraksta slaidu (kas jau satur vienu teksta rāmi, virsrakstu) ar otru teksta rāmi. Visu pareizi izdarot otrā teksta rāmja numurs būs 1 (numerācija sākas ar 0).
3.	ContainsTextBoxes(0, 5)	Visparastākajā Slide Master ir 5 teksta rāmji (virsraksts, pamatteksts un trīs kājenes).



#### **4.20. Funkcija *IsTextBox***

Funkcija *IsTextBox* pārbauda, vai norādītā slaida norādītais elements ir teksta rāmis.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs.

#### **Funkcijai *IsTextBox* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsTextBox</i> (1, 0) & <i>IsTextBox</i> (2, 0) & <i>IsTextBox</i> (3, 0)	Normāli katra slaida nulltais elements ir teksta rāmis (virsraksts)
2.	<i>!IsTextBox</i> (3, 1)	Ir bijis uzdevums izdzēst trešā slaida otro teksta rāmi.

#### **4.21. Funkcija *ContainsText***

Funkcija *ContainsText* pārbauda, vai norādītā slaida norādītais elements ir teksta rāmis vai tabula, kas satur norādīto tekstu.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. teksts.

Teksta rāimim vai tabulai jā satur norādīto tekstu. Teksts var atrasties jebkurā vietā, un, bez norādītā teksta šis teksta rāmis vai tabula drīkst saturēt arī citu tekstu.

#### **Funkcijai *ContainsText* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>ContainsText</i> (3, 1, „Slēgt”) & <i>ContainsText</i> (3, 1, „Uzsākt”) & <i>ContainsText</i> (3, 1, „Palielināt”)	Ir bijis uzdevums ievēdot trešajā slaidā teksta rāmi ar veicināšanas pasākumu sarakstu. Vienkāršības labad mēs nepārbaudām visu ievadāmo tekstu, bet tikai dažus atslēgas vārdus.
2.	<i>ContainsText</i> (0, 2, „Jānis un dēli”)	Ir bijis uzdevums ielikt Slide Master vidējā kājenē (teksta rāmis ar numuru 2) kompānijas nosaukumu.

#### **4.22. Funkcija IsParagraphText**

Funkcija IsParagraphText pārbauda, vai norādītā slaida norādītais elements ir teksta rāmis, kas satur norādīto rindkopu.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. rindkopas numurs,
4. teksts.

Rindkopas teksta rāmī tiek numurētas, sākot ar 1. Tukšas rindkopas tiek skaitītas. Norādītā rindkopa drīkst saturēt tikai norādīto tekstu (ar precizitāti līdz formatējuma elementiem, nebūtiskiem tukšumiem un lielajiem/mazajiem burtiem).

#### **Funkcijai IsParagraphText lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsParagraphText(3, 1, 1, „Slēgt”) & IsParagraphText(3, 1, 2, „Uzsākt”) IsParagraphText(3, 1, 3, „Uzsākt”) & IsParagraphText(3, 1, 4, „Palielināt”)	Ir bijis uzdevums ievēdot trešajā slaidā teksta rāmi ar veicināšanas pasākumu sarakstu. Vienkāršības labad mēs nepārbaudām visu ievadāmo tekstu, bet tikai dažus atslēgas vārdus. Līdzīga, bet mazāk precīza pārbaude ir parādīta arī 4.21 piemēros.

### **4.23. Funkcija IsParagraphFormatted**

Funkcija IsParagraphFormatted pārbauda, vai norādītā slaida norādītā teksta rāmja norādītais paragrāfs ir noformēts tā, kā prasīts.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. teksts,
4. noformējums: Left, Center, Right, Numbered, Bulleted.

Rindkopa trešajā argumentā tiek identificēta ar teksta fragmentu (nevis ar numuru, kā IsParagraphText). Ja teksta rāmī ir vairāki paragrāfi, kas satur norādīto tekstu, tiek pārbaudīts pirmais. Ja teksts nav atrodams nevienā paragrāfā, atbilde ir „Nē”.

#### **Funkcijai IsParagraphFormatted lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsParagraphFormatted(1, 0, „Pasākumi”, „Center”)	Vai pirmā slaida virsraksts ir centrēts. .
2.	IsParagraphFormatted(3, 1, „Slēgt”, „Bulleled”) & IsParagraphFormatted(3, 2, „Uzsākt maizes”, „Bulleled”) & IsParagraphFormatted(3, 3, „Uzsākt konditorijas”, „Bulleled”) & IsParagraphFormatted(3, 4, „Palielināt”, „Bulleled”)	Vai visi četri trešā slaida otrā teksta rāmja paragrāfi (pasākumu saraksts) ir noformēti Bulleted.

#### **4.24. Funkcija IsParagraphLineSpacing**

Funkcija IsParagraphLineSpacing pārbauda, vai norādītā slaida norādītā teksta rāmja norādītais paragrāfs ir noformēts ar prasīto starprindu atstarpi.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. teksts,
4. starprindu atstarpe,
5. pirms-paragrāfa atstarpe,
6. pēc-paragrāfa atstarpe.

Rindkopa trešajā argumentā tiek identificēta ar teksta fragmentu (nevis ar numuru, kā IsParagraphText). Ja teksta rāmī ir vairāki paragrāfi, kas satur norādīto tekstu, tiek pārbaudīts pirmais. Ja teksts nav atrodams nevienā paragrāfā, atbilde ir „Nē”.

Starprindu atstarpi norāda rindās (1 vai 1,5, vai 2), pirms- un pēc-paragrāfa atstarpi norāda punktos (piemēram, 6pt). Impress gadījumā sistēma Tests pati pārvērš punktus centimetros (pēc formulas 1cm = 28,346pt), un tiek pieļauta +/-5% kļūda.

#### **Funkcijai IsParagraphLineSpacing lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsParagraphLineSpacing(3, 1, „Slēgt”, 1, 0,2, 0) & IsParagraphFormatted(3, 2, „Uzsākt maizes”, 1, 0,2, 0) & IsParagraphFormatted(3, 3, „Uzsākt konditorijas”, 1, 0,2, 0) & IsParagraphFormatted(3, 4, „Palielināt”, 1, 0,2, 0”)	Vai visi četri trešā slaida otrā teksta rāmja paragrāfi ir ar atstarpēm 1, 0,2, 0.

#### **4.25. Funkcija *IsTextFormatted***

Funkcija *IsTextFormatted* pārbauda, vai norādītā slaida norādītā teksta rāmja norādītais teksta fragments ir noformēts ar norādīto formatējumu.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. teksts,
4. formatējums: Bold, Italic, Underline, Shadow.

Formatējums tiek pārbaudīts tikai norādītajam tekstam, nevis visam paragrāfam. Ja teksta rāmī norādītais teksts parādās vairākkārtīgi, tiek pārbaudīts pirmais. Ja teksts nav atrodams nevienā paragrāfā, atbilde ir „Nē”.

#### **Funkcijai *IsTextFormatted* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<code>IsTextFormatted(3, 1, „Slēgt”, „Bold”) &amp; !IsTextFormatted(3, 1, „abus”, „Bold”)</code>	Ir bijis uzdevums pasākumu sarakstā vārdu „Slēgt” izcelt ar treknu fontu. Otrā pārbaude noskaidro, vai ar treknu fontu kļūdas dēļ nav izcelti blakus esošie vārdi.
2.	<code>IsTextFormatted(3, 1, „&amp;1”, „Bold”)</code>	Ir bijis uzdevums pasākumu sarakstā izcelt ar treknu fontu uzdevuma parametrā norādīto vārdu. Parametra vērtības varētu būt: „Slēgt”, „Palielināt”. Parametra vērtībai konkrētajā gadījumā nevajadzētu būt „Uzsākt”, jo šis teksts pasākumu sarakstā ir sastopams divas reizes.

#### 4.26. Funkcija IsFont

Funkcija IsFont pārbauda, vai norādītā slaida norādītā teksta rāmja norādītais teksta fragments ir noformēts ar norādītajā fontā.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. teksts,
4. fonts.

Formatējums tiek pārbaudīts tikai norādītajam tekstam, nevis visam paragrāfam. Ja teksta rāmī norādītais teksts parādās vairākkārtīgi, tiek pārbaudīts pirmais. Ja teksts nav atrodams nevienā paragrāfā, atbilde ir „Nē”.

#### Funkcijai IsFont lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsFont(1, 0, „Pasākumi maizes pārdošanas apjoma palielināšanai”, „Arial”) & IsTextFormatted(1, 0, „Pasākumi maizes pārdošanas apjoma palielināšanai”, „Bold”) & IsFont(1, 1, „Kompānija Jānis un dēli”, „Arial”) & IsTextFormatted(1, 1, „Kompānija Jānis un dēli”, „Bold”)	Vai pirmajā slaidā abi virsraksti ir fontā Arial Bold?

#### **4.27. Funkcija IsFontSize**

Funkcija IsFontSize pārbauda, vai norādītā slaida norādītā teksta rāmja norādītais teksta fragments ir noformēts ar norādītajā fonta izmērā.

Funkcijai ir sekojoši argumenti:

5. slaida numurs,
6. elementa numurs,
7. teksts,
8. fonta izmērs.

Formatējums tiek pārbaudīts tikai norādītajam tekstam, nevis visam paragrāfam. Ja teksta rāmī norādītais teksts parādās vairākkārtīgi, tiek pārbaudīts pirmais. Ja teksts nav atrodams nevienā paragrāfā, atbilde ir „Nē”.

Salīdzinot fonta izmēru, tiek pieļauta +/-5% kļūda.

#### **Funkcijai IsFontSize lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsFont(1, 0, „Pasākumi maizes pārdošanas apjoma palielināšanai”, „Arial”) & IsFontSize(1, 0, „Pasākumi maizes pārdošanas apjoma palielināšanai”, 44) & IsTextFormatted(1, 0, „Pasākumi maizes pārdošanas apjoma palielināšanai”, „Bold”)	Vai pirmajā slaidā abi virsraksti ir fontā Arial 44 Bold?

#### 4.28. Funkcija *IsFontColor*

Funkcija *IsFontColor* pārbauda, vai norādītā slaida norādītā teksta rāmja norādītais teksta fragments ir noformēts ar norādītajā fonta krāsā.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. teksts,
4. fonta krāsa.

Krāsa var būt: balts, melns, pelēks, sarkans, zaļš, zils. Tā tiek norādīta un salīdzināta tāpat, kā Word funkcijā *IsShadingColor* (sk. 3.2.8.9).

Krāsa tiek pārbaudīta tikai norādītajam tekstam, nevis visam paragrāfam. Ja teksta rāmī norādītais teksts parādās vairākkārtīgi, tiek pārbaudīts pirmais. Ja teksts nav atrodams nevienā paragrāfā, atbilde ir „Nē”.

#### Funkcijai *IsFontColor* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsFont</i> (1, 0, „Pasākumi maizes pārdošanas apjoma palielināšanai”, „Arial”) & <i>IsFontSize</i> (1, 0, „Pasākumi maizes pārdošanas apjoma palielināšanai”, 44) & <i>IsTextFormatted</i> (1, 0, „Pasākumi maizes pārdošanas apjoma palielināšanai”, „Bold”) & <i>IsFontColor</i> (1, 0, „Pasākumi maizes pārdošanas apjoma palielināšanai”, „Melns”)	Vai pirmajā slaidā abi virsraksti ir fontā Arial 44 Bold un melnā krāsā?



### **4.29. Funkcija ContainsTables**

Funkcija ContainsTables pārbauda, vai norādītajā slaidā ir norādītais skaits tabulu.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. tabulu skaits.

#### **Funkcijai ContainsTables lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	ContainsTables(4, 1)	Vai ceturtajā slaidā ir tieši viena tabula?

### **4.30. Funkcija IsTable**

Funkcija IsTable pārbauda, vai norādītā slaida norādītais elements ir tabula ar vajadzīgo izmēru.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. kolonu skaits
4. rindu skaits.

#### **Funkcijai IsTable lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsTable(4, 1, 2, 3)	Vai ceturtajā slaida otrais elements ir tabula ar divām kolonām un trim rindām?

### **4.31. Funkcija *IsTableExactly***

Funkcija *IsTableExactly* pārbauda, vai norādītā slaida norādītais elements ir tabula ar vajadzīgo saturu.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. tabulas saturs

Tabula tiek uzdota pa rindām. Vērtības tiek atdalītas ar semikoliem, rindas ar „\”. Kolonu un rindu skaitam tabulā jābūt tieši tādām, kā norādīts trešā argumenta dotos. Teksta noformējums netiek pārbaudīts.

#### **Funkcijai *IsTableExactly* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsTableExactly</i> (4, 1, „Nodaļa; Darbinieku skaits\ Sagādes nodaļa; 2\ Ceptuve; 6”)	Vai ceturtajā slaida otrais elements ir tabula ar divām kolonām (Nodaļa un Darbinieku skaits) un trim rindām?

### 4.32. Funkcija *ContainsPictures*

Funkcija *ContainsPictures* pārbauda, vai norādītais slaidis satur norādīto attēlu skaitu.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. attēlu skaits.

#### Funkcijai *ContainsPictures* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>ContainsPictures</i> (4, 0)	Ir bijis uzdevums izdzēst attēlu no ceturtā slaida.
2.	<i>ContainsPictures</i> (0, 1)	Ir bijis uzdevums ielikt attēlu Slide Master.
3.	<i>ContainsPictures</i> (4, &0)	Ir bijis ar frāzi atbildams jautājums: „Cik attēlu ir 4. slaidā?” Ja saskaņotus attēlus novieto blakus, ar šādu jautājumu var pārbaudīt, vai students prot ar kursoru iezīmēt attēlu (jo citādi saskaitīt nav iespējams).

### 4.33. Funkcija *IsPicture*

Funkcija *IsPicture* pārbauda, vai norādītā slaida norādītais elements ir attēls.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs.

#### Funkcijai *IsPicture* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsPicture</i> (0, 5)	Ir bijis uzdevums ielikt attēlu Slide Master. Ieliktajam attēlam būs piektais numurs, jo parasti SlideMaster jau ir pieci teksta rāmji, un slaida elementus numurē, sākot ar 0.
2.	<i>!IsPicture</i> (4, 1)	Ir bijis uzdevums izdzēst attēlu no ceturtā slaida.

#### **4.34. Funkcija IsPictureSize**

Funkcija IsPictureSize pārbauda, vai norādītā slaida norādītais elements ir attēls ar vajadzīgo izmēru.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. attēla augstums pikseļos,
4. attēla platums pikseļos.

Tiek pārbaudīts pikseļu skaits attēlā. Attēla mērogs (t.i. redzamais izmērs) šo pārbaudi neietekmē.

#### **Funkcijai IsPictureSize lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsPictureSize(0, 5, 800, 600)	Ir bijis uzdevums ielikt Slide Master kompānijas logo. Tā kā sistēmā Tests nav iespējams pārbaudīt, vai ir ielikts pareizais attēls, tiek pārbaudīts, vai ieliktajam attēlam ir tādi pat izmēri, kā uzdevumā dotajam kompānijas logo attēlam, pieņemot, ka, ja sakrīt izmēri, tad ielikts ir pareizais attēls.

#### **4.35. Funkcija ContainsOrgCharts**

Funkcija ContainsOrgCharts pārbauda, vai norādītajā slaidā ir norādītais organizācijas shēmu skaits.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. skaits.

Ar organizācijas shēmu tiek saprasts elements, ko ieliek ar izvēlni Insert -> Diagramm -> Organization Chart.

#### **Funkcijai ContainsOrgCharts lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	ContainsOrgCharts(2, 0)	Ir bijis uzdevums izdzēst organizācijas shēmu no otrā slaida.

#### **4.36. Funkcija IsOrgChart**

Funkcija IsOrgChart pārbauda, vai norādītā slaida norādītais elements ir organizācijas shēma.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs.

#### **Funkcijai IsOrgChart lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsOrgChart(2, 1)	Vai otrā slaida pirmais elements ir organizācijas shēma?

#### **4.37. Funkcija ContainsCharts**

Funkcija ContainsCharts pārbauda, vai norādītajā slaidā ir norādītais diagrammu skaits.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. skaits.

Ar organizācijas shēmu tiek saprasts elements, ko ieliek ar izvēlni Insert -> Chart.

#### **Funkcijai ContainsCharts lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	ContainsCharts(4, 0)	Ir bijis uzdevums izdzēst diagrammu no ceturtā slaida.
2.	ContainsCharts(&0, 1)	Ir bijis ar frāzi atbildams jautājums: „Kurā no slaidiem ir diagramma?”.

#### **4.38. Funkcija IsChart**

Funkcija IsChart pārbauda, vai norādītā slaida norādītais elements ir diagramma.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs.

#### **Funkcijai IsChart lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsChart(4, 1)	Vai ceturtā slaida pirmais elements ir diagramma?

#### **4.39. Funkcija IsChartType**

Funkcija IsChartType pārbauda, vai norādītā slaida norādītais elements ir vajadzīgā tipa diagramma.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. diagrammas tips.

Diagrammas tipa tiek norādīts kā PowerPoint izvēlnē Chart -> Chart Type. Tipi drīkst būt:

1. ColumnNormal,
2. BarNormal,
3. PieNormal,
4. Line.

Diagrammas noformēšanai PowerPoint vidē ir ļoti daudz iespēju, tāpēc sistēmā Tests ir neiespējami pārbaudīt, vai students ir izveidojis tādu diagrammu, kā prasīts. Tāpēc, lai pārbaudes būtu precīzākas, vairums uzdevumos būtu jāprasa, lai students modificē esošu diagrammu, nevis izveido jaunu.

#### **Funkcijai IsChartType lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsChartType(4, 1, „Pie. Displays the contribution of each value to a total”)	Ir bijis uzdevums nomainīt diagrammas tipu no stabiņu diagrammas uz norādītā apakštipa riņķa diagrammu.

#### **4.40. Funkcija IsChartTitle**

Funkcija IsChartTitle pārbauda, vai norādītā slaida norādītais elements ir diagramma ar vajadzīgo virsrakstu.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. virsraksta teksts.

Virsraksta teksts ir teksts, kas ievadīts izvēlnē Chart -> Chart Options -> Chart Title.

#### **Funkcijai IsChartTitle lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsChartTitle(4, 1, „Kompānijas nodaļu izmaksas”)	Ir bijis uzdevums nomainīt diagrammas virsrakstu.

#### **4.41. Funkcija IsChartValueColor**

Funkcija IsChartValueColor pārbauda, vai norādītā slaida norādītais elements ir diagramma ar vajadzīgo krāsu.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. vērtības numurs,
4. krāsa.

Tiek pārbaudīta vienas vērtības attēlošanas krāsa. Ja, piemēram, apla diagrammā ir attēlotas trīs vērtības, to krāsu pārbauda ar trim IsChartValueColor funkcijām. Diagrammas tips pārbaudē nav svarīgs.

Krāsa var būt: balts, melns, pelēks, sarkans, zaļš, zils. Tā tiek norādīta un salīdzināta tāpat, kā Word funkcijā IsShadingColor (sk. 3.2.8.9).

#### **Funkcijai IsChartValueColor lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsChartValueColor (4, 1, 1, „Sarkans”) & IsChartValueColor (4, 1, 2, „Zaļš”) & IsChartValueColor (4, 1, 3, „Zils”)	Vai ceturtā slaida pirmais elements ir diagramma ar trim vērtībām kas attēlotas attiecīgi sarkanā, zaļā un zilā krāsā?



#### 4.42. Funkcija IsChartData

Funkcija IsChartData pārbauda, vai norādītā slaida norādītais elements ir diagramma, kas attēlo vajadzīgos skaitliskos datus.

Funkcijai ir sekojoši argumenti:

1. slaida numurs,
2. elementa numurs,
3. attēlojamie dati.

Dati tiek uzdoti pa rindiņām, elementus atdala ar semikolu, rindiņas atdala ar „\”. Nebūtiskie tukšumi tiek ignorēti. Pirmā rindiņa satur virsrakstu.

#### Funkcijai IsChartData lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsChartType(4, 1, „Pie. Displays the contribution of each value to a total”) & IsChartData(4, 1, „Ražošanas daļa; Realizācijas daļa\ &1; &2”)	Ir bijis uzdevums ar parametriem: „Attēlot diagrammā ar tipu „Pie. Displays the contribution of each value to a total” Ražošanas un Realizācijas daļu izmaksas, ja skaitliski šīs izmaksas ir attiecīgi &1 un &2 lati”.
2.	IsChartType(4, 1, „Pie. Displays the contribution of each value to a total”) & IsChartTitle(4, 1, „Kompānijas daļu izmaksas”) & IsChartValueColor (4, 1, 1, „Sarkans”) & IsChartValueColor (4, 1, 2, „Zaļš”) & IsChartValueColor (4, 1, 3, „Zils”) & IsChartData(4, 1, „Ražošanas daļa; Realizācijas daļa\ 38000; 12000”)	Ir bijis uzdevums pārzīmēt diagrammu, kas uzdevuma tekstā dota kā attēls, saglabājot diagrammas veidu, virsrakstu, krāsas un attēlojamās vērtības.

## 5. Pārbaudes funkcijas Access un Base vidē

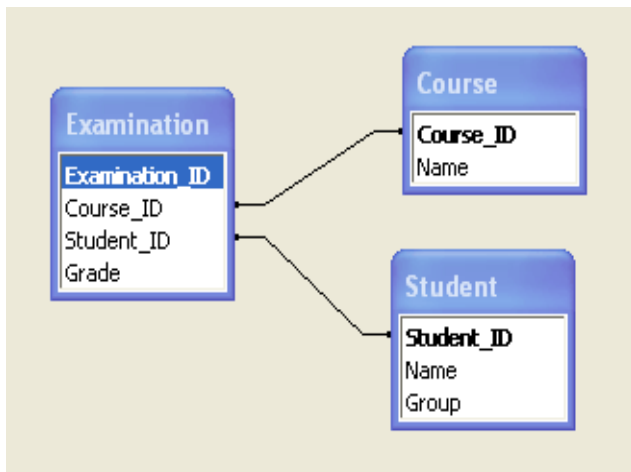
Studenta darba rezultāts ir datu bāze, fails ar paplašinājumu .mdb vai .odb. Ar zemāk aprakstītajām pārbaudes funkcijām tiek pārbaudītas tabulas, vaicājumi, formas un pārskati, kas atrodas šajā datu bāzē.

Visās funkcijās, kur notiek teksta salīdzināšana vai meklēšana, tā notiek studentam maksimāli labvēlīgi: nav case sensitive, nebūtiskie tukšumi un citi formatējuma elementi tiek izlaisti.

Visas funkcijas, kas pārbauda kādas tabulas, vaicājuma, formas vai pārskata īpašības, atbild „Nē”, ja attiecīgā objekta datu bāzē nemaz nav.

Ja dokumentiem (Word/Write), izklājlapām (Excel./Calc) un prezentācijām (PowerPoint/Impress) funkcijas Microsoft Office un Open Office vidēs darbojas faktiski vienādi, tad datu bāzu (Access/Base) gadījumā tā nav – daudzas funkcijas Base vidē nav realizētas, un, sekojot principam, ka visas šaubas jātulko par labu studentam, vienmēr atbild ar „Jā”. Visos šādos gadījumos funkcijas aprakstā tas ir norādīts.

Turpmākajās nodaļās, aprakstot pārbaudes funkcijas, piemēros parasti tiks izmantota sekojoša datu bāze.



### 5.1. Funkcija IsTable

Funkcija IsTable pārbauda, vai datu bāzē ir tabula ar norādīto vārdu.

Funkcijai ir viens arguments:

1. tabulas vārds.

#### Funkcijai IsTable lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsTable(„Student")	Vai datu bāzē ir tabula „Student”?
2.	IsTable(„&1")	Ir bijis uzdevums ar parametru: „Izveidojiet tabulu &1 ar jebkādiem laikiem”

3.	IsMatch(„Student”, &0)	Ir bijis jautājums, kura atbilde jāmeklē Access datu bāzē: „Kurā no tabulām ir visvairāk lauku?” Pārbaudei tiek lietota funkcija jautājumu pārbaudei IsMatch.
----	------------------------	---

## 5.2. Funkcija IsTableColumnCount

Funkcija IsTableColumnCount pārbauda, vai norādītajā tabulā ir vajadzīgais lauku skaits.

Funkcijai ir šādi argumenti:

1. tabulas vārds,
2. lauku skaits.

### Funkcijai IsTableColumnCount lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsTableColumnCount(„Examination”, 4)	Ir bijis uzdevums Izveidot tabulu „Examination” ar 4 laukiem.
2.	IsTableColumnCount(„Examination”, &0)	Ir bijis jautājums, kura atbilde jāmeklē Access datu bāzē: „Cik lauku ir tabulā „Examination”?” Pārbaudīt varēja arī ar funkciju IsMatch(4, &0) – šī pārbaude būtu tik pat pareiza, tikai mazāk elastīga, jo skaitlis 4 būtu ierakstīts formulā.

### 5.3. Funkcija *IsTableColumnType*

Funkcija *IsTableColumnType* pārbauda, viena tabulas lauka tipu.

Funkcijai ir šādi argumenti:

1. tabulas vārds,
2. lauka numurs,
3. lauka vārds,
4. tips,
5. izmērs,
6. formāts,
7. decimālzīmes.

Lauki (2. arguments) tiek numurēti tādā secībā, kādā tie ir tabulā. Argumenti 4-7 atbilst Access loga Table -> Design View iespēju apakškopai. Type – laukam „Data Type” (Number, Auto Number, Text un Currency), izmērs – „Field Size”, formāts – „Format”, decimālzīmes – „Decimal Places” (divi pēdējie – tikai Number tipiem). Lauka tips (argumenti 4-7) ir apakškopa no Access loga Table -> Design View iespējām.

Ja kāds no argumentiem 5-7 nav norādīts, attiecīgā pārbaude netiek veikta. Funkcija uzreiz atbild „Nē”, ja nav norādītās tabulas, tabulā nav vajadzīgā lauka, jeb arī lauka kārtas numurs nav tāds, kā norādīts.

#### Funkcijai *IsTableColumnType* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsTableColumnType(„Course”, 1, „Course_ID”, „Auto Number”, „Long Integer”) &amp; IsTableColumnType(„Course”, 2, „Name”, „Text”, 50)</code>	Ir bijis uzdevums Izveidot tabulu „Course” ar 2 norādītā tipa laukiem.

#### **5.4. Funkcija *IsTablePrimaryKey***

Funkcija *IsTablePrimaryKey* pārbauda, vai norādītais lauks ir tabulas primārā atslēga.

Funkcijai ir šādi argumenti:

1. tabulas vārds,
2. lauka vārds.

#### **Funkcijai *IsTablePrimaryKey* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<code>IsTablePrimaryKey(„Course”, „Course_ID”) &amp; IsTableColumnType(„Course”, 1, „Course_ID”, „Auto Number”, „Long Integer”)</code>	Vai tabulas „Course” primārā atslēga ir lauks „Course_ID”, un vai tam ir vajadzīgais tips?

#### **5.5. Funkcija *IsRowCount***

Funkcija *IsRowCount* pārbauda, vai tabulā ir vajadzīgais rindiņu (ierakstu) skaits.

Funkcijai ir šādi argumenti:

1. tabulas vārds,
2. lauka vārds,
3. ierakstu skaits.

#### **Funkcijai *IsRowCount* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<code>IsRowCount(„Student”, 2)</code>	Vai tabulā „Student” ir 2 ieraksti?
2.	<code>IsRowCount(„Student”, 0)</code>	Ir bijis uzdevums izdzēst visus ierakstus no tabulas Student.
3.	<code>IsRowCount(„Student”, 120)</code>	Ir bijis uzdevums izdzēst visu 3C grupas studentu ierakstus (pieņemot, ka citās grupās ir tieši 120 studenti). Šādu uzdevumu reāli var izpildīt tikai lietojot kārtošanu.

## 5.6. Funkcija IsCellValue

Funkcija IsCellValue pārbauda, vai vērtību vienā tabulas šūnā (rindas un kolonas krustpunktā).

Funkcijai ir šādi argumenti:

1. tabulas vārds,
2. lauka vārds,
3. rindas numurs
4. vērtība.

Rindas tiek numurētas tabulas primārās atslēgas noteiktajā secībā. Vērtība var būt vai nu skaitlis vai simbolu rinda.

Lai pārbaudītu visa ieraksta (rindas) saturu, jāpārbauda atsevišķi katra šūna – t.i. jālieto vairākas funkcijas IsCellValue.

### Funkcijai IsCellValue lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsCellValue(„Student”, „Name”, 1, „Jānis Kalniņš”) & IsCellValue(„Student”, „Group”, 1, „2A”) & IsCellValue(„Student”, „Name”, 2, „Pēteris Lejiņš”) & IsCellValue(„Student”, „Group”, 2, „2A”)	Vai tabulā „Student” ir 2 ieraksti ar vajadzīgo saturu? Lauks „Student_ID” netiek pārbaudīts.
2.	IsCellValue(„Student”, „Group”, 999, „2B”)	Ir bijis uzdevums: „Nomainīt studenta Kārļa Krūmiņa grupu uz 2A” (tiek pieņemts, ka Kārļa Krūmiņa ieraksti datos ir pēc kārtas 999.). Pie pietiekami liela ierakstu skaita šādu uzdevumu var izpildīt tikai lietojot meklēšanas vai kārtošanas funkcijas.
3.	IsRowCount(„Student”, 999) & IsCellValue(„Student”, „Name”, 1, „Ada Ābele”) & IsCellValue(„Student”, „Group”, 1, „1A”) & IsCellValue(„Student”, „Name”, 999, „Zoja Žubīte”) & IsCellValue(„Student”, „Group”, 999, „9Z”)	Ir bijis uzdevums ieimportēt tabulā datus no teksta faila. Tiek pārbaudītas tikai dažas vērtības, jo praktiski fails ir vai no pareizi ieimportēts, un tad visas rindas ir pareizas, vai arī nepareizi ieimportēts, un tad visas rindas ir nepareizas.

### **5.7. Funkcija IsRelationship**

Funkcija IsRelationship pārbauda, vai starp divu tabulu divām kolonām ir nodibināta noteikta tipa attiecība (izvēlne Tools -> Relationships).

Funkcijai ir šādi argumenti:

1. tabulas vārds 1,
2. lauka vārds 1,
3. tabulas vārds 2,
4. lauka vārds 2,
5. attiecības tips,
6. integritāte: Jā vai Nē.

Funkcijas argumenti atbilst izvēlnei Tools -> Relationships. Attiecības tips atbilst loga Edit Relationships apakšlogam Join Type (1, 2 vai 3). Pēdējie trīs parametri pilnībā atbilst attiecīgajiem loga Edit Relationships laukiem.

Base vidē attiecības tips un integritāte (5. un 6. arguments) netiek pārbaudīti.

#### **Funkcijai IsRelationship lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsRelationship(„Examination”, „Course_ID”, „Course”, „Course_ID”, 1, „No”)	Vai starp tabulas Examination lauku Course_ID un tabulas Course lauku Course_ID ir nodibināta attiecība ar tipu 1 (Only include rows, where the joined fields from both tables are equal) un bez integritātes kontroles?

### **5.8. Funkcija IsTableOrdered**

Funkcija IsTableOrdered pārbauda, vai tabulas noklusētais skatījums ir sakārtots pēc norādītā lauka.

Funkcijai ir šādi argumenti:

1. tabulas vārds,
2. lauka vārds,
3. augošs vai dilstošs.

Funkcija darbojas tikai Access vidē, Base vidē tā vienmēr atgriež vērtību „Jā”.

#### **Funkcijai IsTableOrdered lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsTableOrdered(„Student”, „Name”, „Ascending”)	Vai tabula Student ir sakārtota augošā secībā pēc lauka Name?

### 5.9. Funkcija *IsColumnWidth*

Funkcija *IsColumnWidth* pārbauda, vai tabulas noklusētajā skatījumā norādītajai kolonai (laukam) ir vajadzīgais platums.

Funkcijai ir šādi argumenti:

1. tabulas vārds,
2. lauka vārds,
3. platums.

Pārbaudot platumu tiek pieļauta +/-10% kļūda.

#### Funkcijai *IsTableOrdered* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsTableOrdered(„Student”, „Name”, „Ascending”)</code>	Vai tabula Student ir sakārtota augošā secībā pēc lauka Name?

### 5.10. Funkcija *IsQuery*

Funkcija *IsQuery* pārbauda, vai datu bāzē ir vaicājums (query) ar norādīto vārdu.

Funkcijai ir viens arguments:

1. vaicājuma vārds.

#### Funkcijai *IsQuery* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsQuery(„Student 1A”)</code>	Vai datu bāzē ir vaicājums „Student 1A”?

### 5.11. Funkcija *IsQueryColumnCount*

Funkcija *IsQueryColumnCount* pārbauda, vai datu bāzē norādītajam vaicājumam ir vajadzīgais lauku skaits.

Funkcijai ir sekojoši argumenti:

1. vaicājuma vārds,
2. lauku skaits.

Lauku skaitā ietilpst arī lauki, kas netiek rādīti (Show=No).

Base vidē netiek pārbaudīts, vai „kritērija” ir piesaistīts kolonai „kolonas numurs”- funkcija atgriež True, ja šāds kritērijs ir norādīts jebkurai kolonai.

#### Funkcijai *IsQueryColumnCount* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsQueryColumnCount(„Student 1A”, 2)</code>	Vai vaicājumam „Student 1A” ir tieši viena kolona?



### 5.12. Funkcija *IsQueryColumn*

Funkcija *IsQueryColumn* pārbauda vienu vaicājuma lauku (kolonu).

Funkcijai ir sekojoši argumenti:

1. vaicājuma vārds,
2. lauka numurs vaicājumā,
3. tabulas vārds,
4. kolonas vārds tabulā,
5. augoša vai dilstoši sakārtojums,
6. atlases kritērijs.

Funkcija pārbauda, vai vaicājumā (1. arguments) lauks (2. arguments) ir piesaistīts tabulas (3. arguments) kolonai (4. arguments), un vai vaicājums pēc šīs kolonas ir sakārtots, un kāds ir ierakstu atlases kritērijs.

Atlases kritēriju norāda kā tekstuālu izteiksmi – kā Expression Builder. Kritērijam jābūt pietiekami vienkāršam, lai nerastos problēmas ar formulu ekvivalenci. Ja Kritērijs satur datumu, tad datumu ievada formā, kas atšķiras no parastās:

1. kritērijs sākas ar = vai > vai <
2. pēc tam seko #
3. datums jāsniedz ievadīt ar mēnesi (un bez 0 priekšā)
4. kā atdalītāji jālieto /
  1. tālāk jānorāda diena (bez 0 priekšā), pēc tam - gads
5. beigās atkal zīme #

Piemēram, =14.04.2008 būtu nepareizi. Pareizi ir =#4/14/2008#

Ja kāds no argumentiem 5 un 6 ir izlaists, attiecīgā pārbaude netiek veikta (no tā netieši seko, ka atlases kritērijā nav iespējams pārbaudīt, vai lauks ir tukšs).

#### Funkcijai *IsQueryColumn* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsQueryColumnCount(„Student 1A”, 1, „Student”, „Group”, „=1A”) &amp; IsQueryColumnCount(„Student 1A”, 2, „Student”, „Name”, „Ascending”)</code>	Vai vaicājumā „Student 1A” tiek parādīti grupas 1A studenti alfabēta secībā?

### 5.13. Funkcija *IsForm*

Funkcija *IsForm* pārbauda, vai datu bāzē ir forma ar norādīto vārdu.

Funkcijai ir viens arguments:

1. formas vārds.

### **Funkcijai IsForm lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsForm(„Student“)	Vai datu bāzē ir forma „Student”? Tabula „Student” neder, kaut saucas tāpat.

### **5.14. Funkcija IsFormFieldCount**

Funkcija IsFormFieldCount pārbauda, vai datu bāzē norādītajai formai ir vajadzīgais lauku skaits.

Funkcijai ir sekojoši argumenti:

1. formas vārds,
2. lauku skaits.

Tiek skaitīti visi lauki, tai skaitā kā „Label” tipa lauki, ko lieto noformējumam, tā „Text Box” tipa lauki, kas ar īpašības „Controll source” palīdzību ir piesaistīti tabulas laukiem. Tiek skaitīti tikai „Detail” sekcijā esošie lauki – galveņu un kājeņu laukus neskaita.

#### **Funkcijai IsFormFieldCount lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsFormFieldCount(„Student", 4)	Vai formā „Student” ir tieši četri lauki? (var sagaidīt, ka tie ir divi Label tipa lauki un divi tabulai piesaistīti lauki)

### **5.15. Funkcija IsFormField**

Funkcija IsFormField pārbauda vienu formas lauku.

Funkcijai ir sekojoši argumenti:

1. formas vārds,
2. lauka tips
3. teksts
4. fonts
5. fonta izmērs
6. fonta krāsa.

Lauki formā apzināti netiek identificēti ar lauka vārdu, jo lauka vārdu, veidojot formu, var norādīt jebkādu. Funkcija vienkārši pārbauda, vai starp visiem formas laukiem, izņemot galvenes un kājenes, ir lauks (varbūt viens, varbūt vairāki) ar norādītajām īpašībām.

Lauka tips (3. arguments) ir Label vai Text Box.

Teksts (3. arguments) Label gadījumā atbilst īpašībai Caption, bet Text Box gadījumā – īpašībai Controll Source (izteiksme, piemēram, „Name” vai „Left(Group;1)”).

Fonts (4. arguments) atbilst īpašībai Font Name.

Fonta izmērs (5. arguments) atbilst īpašībai Font Size.

Krāsa (6. arguments) atbilst īpašībai Font Color, un tiek norādīta tāpat, kā Word funkcijā IsFontColor (sk. 2.8).

Ja kāds no argumentiem ir izlaists, attiecīgā īpašība netiek pārbaudīta.

#### **Funkcijai IsFormField lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsFormField(„Student", „Text Box”, „Name”, „MS Sans Serif”, 8, „Meln”)	Vai formā ir Text Box tipa lauks, kas rāda tabulas lauku „Name” un ir noformēts fontā MS Sans Serif, 8 pt izmērā, melnā krāsā?

### **5.16. Funkcija *IsFormHeaderFieldCount***

Funkcija darbojas tāpat, kā *IsFormFieldCount*, tikai attiecas uz laukiem formas galvenē.  
Base vidē funkcija nav realizēta un vienmēr atbild ar „Jā”.

### **5.17. Funkcija *IsFormHeaderField***

Funkcija darbojas tāpat, kā *IsFormField*, tikai attiecas uz laukiem formas galvenē.  
Base vidē funkcija nav realizēta un vienmēr atbild ar „Jā”.

### **5.18. Funkcija *IsFormFooterFieldCount***

Funkcija darbojas tāpat, kā *IsFormFieldCount*, tikai attiecas uz laukiem formas kājenē.  
Base vidē funkcija nav realizēta un vienmēr atbild ar „Jā”.

### **5.19. Funkcija *IsFormFooterField***

Funkcija darbojas tāpat, kā *IsFormField*, tikai attiecas uz laukiem formas kājenē.  
Base vidē funkcija nav realizēta un vienmēr atbild ar „Jā”.

### **5.20. Funkcija *Isreport***

Funkcija *IsForm* pārbauda, vai datu bāzē ir pārskats ar norādīto vārdu.

Funkcijai ir viens arguments:

1. pārskata vārds.

**Funkcijai *IsReport* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsReport</i> („Grades”)	Vai datu bāzē ir pārskats „Grades”?

### **5.21. Funkcija *IsReportFieldCount***

Funkcija *IsReportFieldCount* pārbauda, vai datu bāzē norādītajai formai ir vajadzīgais lauku skaits.

Funkcijai ir sekojoši argumenti:

1. pārskata vārds,
2. lauku skaits.

Funkcija darbojas līdzīgi, kā *IsFormFieldCount*.

## **5.22. Funkcija *IsReportField***

Funkcija *IsReportField* pārbauda vienu pārskata lauku.

Funkcijai ir sekojoši argumenti:

1. pārskata vārds,
2. lauka tips
3. teksts
4. fonts
5. fonta izmērs
6. fonta krāsa.

Funkcija darbojas līdzīgi, kā *IsFormField*.

### 5.23. Funkcija **IsReportSortGroup**

Funkcija **IsReportSortGroup** pārbauda, vai datu bāzē ir Atskaite ir grupēta vai kārtota pēc norādītā lauka, un ar norādītajiem atribūtiem (sk. Access pārskatos logu „Sorting And Grouping”).

Funkcijai ir sekojoši argumenti:

1. pārskata vārds,
2. lauka vārds,
3. Sort Order: Ascending vai Descending,

Lauka vārds atbilst loga „Sorting And Grouping” laukam „Field/Expression. Deīkst norādīt tikai lauka vārdu, izteiksmi – nē. Pārējie argumenti pilnībā atbilst šī loga laukiem.

Ja kāds no argumentiem nav norādīts, attiecīgā pārbaude netiek veikta. Funkcija pārbauda, vai starp pārskata laukiem eksistē kaut viens, kas apmierina visus argumentos norādītos nosacījumus. Atskaite drīkst būt kārtota vai grupēta arī pēc citiem laukiem – uzdevumu sastādītājs drīkst arī šos citus laukus pārbaudīt ar funkciju **IsReportSortGroup**.

#### **Funkcijai **IsReportSortGroup** lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<b>IsReportSortGroup</b> („Students By Group”, „Group”, „Ascending”) & <b>IsReportSortGroup</b> („Students By Group”, „Name”, „Ascending”) & <b>! IsReportSortGroup</b> („Students By Group”, „Student_ID”)	Vai atskaite ir kārtota pēc laukiem „Group” un „Name”, bet nav kārtota pēc lauka „Student_ID”? Jāņem vērā, ka funkcija <b>IsReportField</b> pārbauda arī, vai lauks „Student_ID” pārskatā vispār eksistē - tā atbild nē gan tad, ja pēc lauka „Student_ID” nenotiek kārtošana, gan arī tad, ja šāda lauka vispār nav.

### **5.24. Funkcija *IsReportPage***

Funkcija *IsReportPage* pārbauda, vai pārskata drukāšanai ir norādīta vajadzīgā lappuses orientācija (portrait/landscape).

Funkcijai ir viens arguments:

1. pārskata vārds,
2. lappuses orientācija: portrait vai landscape.

#### **Funkcijai *IsReportPage* lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	<i>IsReportPage</i> („Students By Group”, „Portrait”)	Vai pārskatam „Students By Group” lappuses orientācija ir „Landscape”?

### **5.25. Funkcija *IsReportHeaderFieldCount***

Funkcija darbojas tāpat, kā *IsFormFieldCount*, tikai attiecas uz laukiem pārskata galvenē.

Base vidē funkcija nav realizēta un vienmēr atbild ar „Jā”.

### **5.26. Funkcija *IsReportHeaderField***

Funkcija darbojas tāpat, kā *IsFormField*, tikai attiecas uz laukiem pārskata galvenē.

Base vidē funkcija nav realizēta un vienmēr atbild ar „Jā”.

### **5.27. Funkcija *IsReportFooterFieldCount***

Funkcija darbojas tāpat, kā *IsFormFieldCount*, tikai attiecas uz laukiem pārskata kājenē.

Base vidē funkcija nav realizēta un vienmēr atbild ar „Jā”.

### **5.28. Funkcija *IsReportFooterField***

Funkcija darbojas tāpat, kā *IsFormField*, tikai attiecas uz laukiem pārskata kājenē.

Base vidē funkcija nav realizēta un vienmēr atbild ar „Jā”.



## 6. Pārbaudes funkcijas Windows un Linux vidē

Uzdevumos tiek pārbaudītas studentu iemaņas veikt manipulācijas ar failiem (kopēt, dzēst...) un iemaņas noskaidrot un mainīt operētājsistēmas un datora uzstādījumus.

Visas manipulācijas ar failiem students veic mapītē \tests un tās apakš mapītēs. Mapīte \tests tiek novietota katrai operētājsistēmai citā vietā: Windows operētājsistēmām – tieši c: diska saknes direktoriā, bet Linux operētājsistēmai – lietotāja mājas direktoriā. Sastādot uzdevumus failu vārdi jāizvēlas tādi, lai tie būtu lietojami kā Windows, tā Linux operētājsistēmās. Failu vārdos ieteicams lietot tikai mazos (lowercase) burtus, jo Windows operētājsistēmā lielie un mazie burti faila vārdos tiek uzskatīti par vienādiem, bet Linux operētājsistēmā tie tiek uzskatīti par dažādiem. Par mapīšu atdalītāju lieto simbolu „\” (Windows tradīcija), un sistēma pati to nomaina, ja uzdevums tiek pildīts Linux operētājsistēmā. Visas takas zemāk minētajās funkcijās jānorāda relatīvi attiecībā pret c:\test. Uz devumu formulējumos var lietot mainīgo (&folder), kuru sistēma aizstāj ar pilnu taku uz darba mapīti atkarībā no operētājsistēmas.

Pirms katra praktiskā uzdevuma sākšanas, ja uzdevumā ir paredzēti faili, mapītes \tests saturs tiek pilnībā atjaunots no praktiskajam uzdevumam pievienotā arhīva. Uzdevuma rezultāts ir darba mapes saturs – visi faili un apakšmapītes. Pēc praktiskā uzdevuma pabeigšanas šie faili un mapītes tiek saarhivēti un saglabāti turpmākajai darbu pārbaudei.

Zināšanas par operētājsistēmas uzstādījumiem tiek pārbaudītas ar Jā/Nē jautājumiem un ar frāzi atbildamiem jautājumiem. Abos gadījumos atbildes pārbaudē lieto parametru &0 (studenta atbildi), piemēram, IsMemorySize(&0,"M"), t.i. „Vai atmiņas apjoms ir tāds, kā atbildēja students?”. Uzdevuma rezultātam šajā gadījumā tiek pievienota atskaite par tā datora stāvokli, kurā uzdevums pildīts. Šī atskaite līdzīga MS System Information, un satur visu pārbaudes funkciju realizācijai nepieciešamo informāciju (atmiņas daudzumu, disku lielumu, instalētās programmas, palaistās programmas), kā arī informāciju, kas studenta sūdzību gadījumā ļautu daudz maz droši identificēt datoru (piemēram, tīkla kartes MAC adresi, datora vārdu...).

Daudzas pārbaudes funkcijas ir dabīgi lietot kā ar frāzi atbildamos jautājumos (piemēram, IsFileSize(„report.txt”,&0,"K"), lai pārbaudītu, vai students prot sameklēt faila lielumu), tā arī praktiskajos uzdevumos (piemēram, ar IsFile(„report.txt”), lai pārbaudītu, vai students ir pārsaucis failu.

Funkciju argumentos norādītie teksti nedrīkst saturēt speciālos simbolus: apostrofu, pēdiņu, semikolu un konjunktivas zīmi.

## 6.1. Funkcija *IsFile*

Funkcija *IsFile* pārbauda, vai eksistē norādītā datne.

Funkcijai *IsFile* ir viens arguments:

1. taka, kas norāda uz datni.

Funkcija atbild „Nē” gan tad, ja nav norādītās datnes, gan tad, ja nav kādas no takā norādītajām mapītēm.

### Funkcijai *IsFile* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsFile(„atskaite.txt”) &amp; ! IsFile(„report.txt”)</code>	Ir bijis praktisks uzdevums: „Nomainīt datnes report.txt nosaukumu uz atskaite.txt”.
2.	<code>IsFile(„&amp;1”) &amp; ! IsFile(„report.txt”)</code>	Ir bijis praktisks uzdevums ar parametru: „Nomainīt datnes report.txt nosaukumu uz &1”.
3.	<code>IsFile(„dokumenti\iesniegums.doc”) &amp; ! IsFile(„iesniegums.doc”)</code>	Ir bijis praktisks uzdevums: „Izveidot mapīti dokumenti un pārnest uz tu datni iesniegums.doc”
4.	<code>! IsFile(„dokumenti\iesniegums.doc”) &amp; ! IsFile(„dokumenti\saraksts.doc”)</code>	Ir bijis praktisks uzdevums: „Izdzēst mapīti dokumenti ar visām tajā esošajām datnēm”.
5.	<code>IsFile(„&amp;0”)</code>	Ir bijis jautājums: „Kāda datne atrodas mapītē c:\tests?”.

## 6.2. Funkcija *IsDirectory*

Funkcija *IsDirectory* pārbauda, vai eksistē norādītā mapīte.

Funkcijai *IsDirectory* ir viens arguments:

1. taka, kas norāda uz mapīti.

### Funkcijai *IsDirectory* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsDirectory („dokumenti\sarakste”)</code>	Ir bijis praktisks uzdevums: „Izveidot mapīti dokumenti, bet mapītē dokumenti izveidot vēl vienu mapīti - sarakste”.

### 6.3. Funkcija *IsFileReadOnly*

Funkcija *IsFileReadOnly* pārbauda, vai datnei uzstādīts režīms „Read only”.

Funkcijai *IsFileReadOnly* ir viens arguments:

1. taka, kas norāda uz datni.

**Funkcijai *IsFileReadOnly* lietošanas piemēri**

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsFileReadOnly</i> („atskaite.txt”)	Ir bijis praktisks uzdevums: „Uzstādīt datnei <i>atskaite.txt</i> režīmu „Read only”.

### 6.4. Funkcija *IsFileModifiedAfter*

Funkcija *IsFileModifiedAfter* pārbauda, vai datnei ir norādītais pēdējās modificēšanas datums.

Funkcijai *IsFileModifiedAfter* ir sekojoši argumenti:

1. taka, kas norāda uz datni,
2. datums.

**Funkcijai *IsFileModifiedAfter* lietošanas piemēri**

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsFileModifiedAfter</i> („&1”, „&0”)	Ir bijis jautājums ar parametru: „Kāds ir datnes &1 pēdējās labošanas datums?”

### 6.5. Funkcija *IsFileSize*

Funkcija *IsFileSize* pārbauda, vai datnei ir aptuveni norādītais izmērs.

Funkcijai *IsFileSize* ir sekojoši argumenti:

1. taka, kas norāda uz datni,
2. izmērs,
3. mērvienība: K (kilobaiti), M (megabaiti) .

Ja mērvienība nav norādīta, tiek uzskatīts, ka izmērs dots baitos. Izmērā tiek pieļauta +/-10% kļūda.

**Funkcijai *IsFileSize* lietošanas piemēri**

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsFileSize</i> („&1”, „&0”, „K”)	Ir bijis jautājums ar parametru: „Kāds ir datnes &1 izmērs kilobaitos?”

## 6.6. Funkcija *IsFileSizeExactly*

Funkcija *IsFileSizeExactly* pārbauda, vai datnei ir precīzi norādītais izmērs.

Funkcijai *IsFileSizeExactly* ir sekojoši argumenti:

1. taka, kas norāda uz datni,
2. izmērs baitos .

Datnes lielums tiek pārbaudīts precīzi, nekāda kļūda netiek pieļauta. Funkcija paredzēta, lai būtu iespējams pārbaudīt vai „šī ir tā pati datne”.

### Funkcijai *IsFileSizeExactly* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<code>IsFile(„dokumenti\iesniegums.doc”) &amp; ! IsFile(„iesniegums.doc”) &amp; IsFileSizeExactly(„iesniegums.doc”, 2482)</code>	Ir bijis praktisks uzdevums: „Izveidot mapīti dokumenti un pārnest uz tu datni iesniegums.doc”. Uzdevuma sastādītājs paskatās datnes izmēru, un ieraksta to pārbaudes formulā, lai pārlicinātos, ka nokopēta pareizā datne (nevis, piemēram, izveidota jauna).
2.	<code>IsFileSizeExactly(„atskaite.txt”, 501)</code>	Ir bijis uzdevums: „Ar funkciju Search sameklēt visās mapēs c:\tests apakšmapēs datni, kas satur vārdu atskaite, un pārkopēt to uz mapīti c:\tests” un pārsaukt par atskaite.txt”. Mapītē ir pietiekami daudz apakšmapīšu un datņu, lai bez Search funkcijas vajadzīgu datni sameklēt praktiski nebūtu iespējams.

## 6.7. Funkcija *IsFileType*

Funkcija *IsFileType* pārbauda, vai datnei ir norādītais tips.

Funkcijai *IsFileType* ir sekojoši argumenti:

1. taka, kas norāda uz datni,
2. tips.

Datnes tips ir, piemēram, doc, xls, mdb. Pārbaude ir dažkārt nepieciešama, jo datnes tips, kas norādīts faila vārdā, var neatbilst reālajam saturam. Šī funkcija pārbauda tipu pēc datnes satura („Properties. General. Type of file”).

### Funkcijai *IsFileType* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsFileType</i> („izdevumi.mdb”, „mdb”)	Ir bijis uzdevums: „Izveidot jaunu MS Access datu bāzi izdevumi.mdb”. Pārbaudes funkcija nodrošinās pret gadījumu, kad datne ir izveidota, piemēram, ar Notepad, bet tikai nosaukta izdevumi.mdb

## 6.8. Funkcija *IsFileText*

Funkcija *IsFileText* pārbauda, vai datne satur norādīto tekstu.

Funkcijai *IsFileText* ir sekojoši argumenti:

1. taka, kas norāda uz datni,
2. teksts.

Funkcija pārbauda, vai norādītajā datnē ir norādītais teksts. Salīdzinot tekstu, lielle un mazie burti tiek uzskatīti par vienādiem. Bez norādītā teksta datnē var atrasties arī cits teksts.

### Funkcijai *IsFileText* lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	<i>IsFileText</i> („saule.txt”, „lietus”)	Ir bijis uzdevums: „Ar Notepad izveidot jaunu datni saule.txt un ierakstīt tajā vārdu „lietus””.
2.	<i>IsFileText</i> („kobalts.htm”, „kobalt”)	Ir bijis uzdevums: „Sameklēt internetā jebkuru lapu, kurā ir informācija par kobaltu, un saglabāt to datnē kobalts.txt”. Pārbaudot mēs apmierināties ar to, ka datnē vispār ir vārds „kobalts” (formulā vārda galotni norādīt nedrīkst).

## 6.9. Funkcija *IsMemorySize*

Funkcija *IsMemorySize* pārbauda, vai datoram ir norādītais operatīvās atmiņas daudzums.

Funkcijai IsMemorySize ir sekojoši argumenti:

1. atmiņas daudzums,
2. vienības : „K” (Kilobaiti), „M”(Megabaiti) vai G (Gigabaiti).

Ja 2. arguments nav norādīts, atmiņas apjoms jānorāda baitos. Salīdzinot tiek pieļauta +/-10% kļūda.

#### Funkcijai IsMemorySize lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsMemorySize(512, „M”)	Ir bijis jautājums: „Kāds ir datora operatīvās atmiņas daudzums megabaitos?”.

### 6.10. Funkcija IsDiskSize

Funkcija IsDiskSize pārbauda, vai datora cietajam diskam c: ir norādītais lielums.

Funkcijai IsDiskSize ir sekojoši argumenti:

1. diska lielums,
2. vienības : „K” (Kilobaiti), „M”(Megabaiti) vai G (Gigabaiti).

Tiek pārbaudīts sistēmas diska c: apjoms. Ja 2. arguments nav norādīts, diska apjoms jānorāda baitos. Salīdzinot tiek pieļauta +/-10% kļūda.

#### Funkcijai IsDiskSize lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsDiskSize(&0, „G”)	Ir bijis jautājums: „Kāds ir datora cietā diska c: apjoms gigabaitos?”.

### 6.11. Funkcija IsProgrammInstalled

Funkcija IsProgrammInstalled pārbauda, vai datorā ir uzstādīta norādītā programma.

Funkcijai IsProgrammInstalled ir sekojoši argumenti:

1. programmas vārds,
2. „Jā” vai „Nē”.

Programmas vārds jānorāda tā, kā tas parādās Controll panel -> Add or remove programms sarakstā, piemēram, „Microsoft Officer Professional Edition 2003”.

#### Funkcijai IsProgrammInstalled lietošanas piemēri

Nr.	Funkcijas lietojums	Komentārs
1.	IsProgrammInstalled(„&0”, „Jā”)	Ir bijis jautājums (labāk – ar atbilžu variantiem): „Ar Controll panel -> Add or remove programms noskaidrojiet, kāda MS Office

		programmatūras versija uzstādīta datorā”.
--	--	---

### **6.12. Funkcija IsProgrammStarted**

Funkcija IsProgrammStarted pārbauda, vai datorā ir pašreiz palaista norādītā programma.

Funkcijai IsProgrammStarted ir sekojoši argumenti:

1. programmas vārds,
2. „Jā” vai „Nē”.

Programmas vārds jānorāda tā, kā tas parādās Task manager sadaļā Processes, piemēram, „winword.exe”.

#### **Funkcijai IsProgrammStarted lietošanas piemēri**

<b>Nr.</b>	<b>Funkcijas lietojums</b>	<b>Komentārs</b>
1.	IsProgrammInstalled(„&0”, „Jā”)	Ir bijis jautājums ar atbilžu variantiem: „Ar Task Manager palīdzību noskaidrojiet, kura no sekojošajām programmām pašreiz ir palaista”.

## 7. Pārbaudes funkcijas Internet Explorer un Mozilla Firefox vidē

Visām iepriekš apskatītajām lietojumprogrammām praktiskie uzdevumi ir tādi, ka studenta darba rezultāts vienmēr ir fails (Word dokuments, Access datu bāze vai Outlook pastkaste). Uzdevums liek studentam veikt tādas darbības, kuru rezultāts ir redzams šajā failā, un tikai pēc šī rezultāta faila sistēma nosaka, vai uzdevums izpildīts pareizi. Sistēma nevērtē to, kādā veidā students sasniedzis mērķi, kādas izvēlnes lietoja, kādas taustiņu kombinācijas nospieda – tai svarīgi vienīgi pareizs rezultāts, kas atspoguļojas saglabātajā failā. Šāda pieeja padara vērtējumu objektīvāku. Strīdu gadījumā vērtēšanu vienmēr var atkārtot.

Internet Explorer un Mozilla Firefox gadījumā praktisko uzdevumu izpildes rezultāts parasti nav fails, bet gan fakti vai zināšanas, kuras students iegūst no interneta, pildot uzdevumu. Un tā kā nav rezultāta faila, tad nav arī pārbaudes funkciju. Zināšanas parasti tiek pārbaudītas vai nu ar teorētiskiem jautājumiem (bet tādiem, uz kuriem var atbildēt tikai veicot noteiktas darbības pārlūprogrammā), vai arī ar praktiskiem uzdevumiem, kas tiek pārbaudīti ar Windows/Linux funkcijām.

Tā kā informācija internetā nepārtraukti mainās, tad jautājumu ilglaicības nodrošināšanai var būt nepieciešams uzturēt speciālu mājas lapu, kurā esošo informāciju var droši izmantot jautājumos, zinot, ka tā neizmainīsies.

Zemāk doti vairāki piemēri, kā ar sistēmas Tests līdzekļiem varētu pārbaudīt ECDL 7. moduļa 1. nodaļā „Web Browsing” prasītās zināšanas. Piemēri jāuztver tikai kā ilustrācijas, to mērķis nav parādīt pilnīgi visus sistēmas Tests līdzekļus un paņēmienus, kas ir lietojami jautājumu sagatavošanā. Piemēram, vienkārši teorētiski jautājumi un jautājumi, uz kuriem atbild norādot attēlā, vispār netiek pieminēti. Jautājumu sagatavošana pārlūpprogrammu videi ir daudz radošāks darbs, nekā jautājumu sagatavošanas citām lietojumprogrammām.

Piemēros dažkārt lietotas konkrētas Web lapas, kuras reālos uzdevumos, jau pieminētās jautājumu ilglaicības dēļ, būtu jāaizstāj ar speciāli sagatavotām un nemainīgām lapām.



REF.	Task Item	Uzdevuma vai jautājuma piemērs
7.1.1.1 1	Understand what the Internet is. Understand some Internet services like: World Wide Web (WWW), email, newsgroups.	
7.1.1.1 2	Define and understand the terms: URL, hyperlink, ISP.	Praktisks uzdevums Word vidē: „Atveriet dokumentā norādīto hipersaiti <a href="http://www.lu.lv/par/vesture.html">http://www.lu.lv/par/vesture.html</a> . Papildiniet dokumentā frāzi „1919. gada ??septembrī, Rīgas Politehniskā institūta ēkā svinīgi tika atklāta Latvijas Augstskola” ar pareizo atklāšanas datumu.” Pārbaudes formula: ContainsText(„Paragraph, 3, „23”)
7.1.1.1 3	Understand the make-up and structure of a Web address.	Ar jā/nē atbildams jautājums: „Izmantojot FTP protokolu, atveriet pārlūkprogrammā serveri <a href="ftp://latnet.lv">ftp.latnet.lv</a> . Vai starp mapītēm, kas tiek uzreiz parādītas, ir mapīte „software?” Ar frāzi atbildams jautājums: „Dodieties uz tīmekļa adresi <a href="http://www.lu.lv">www.lu.lv</a> un atveriet sadaļu „Universitāte” (zila saite starp „Gribu studēt” un „Fakultātes”). Apskatiet šīs lapas URL. Kādā mapītē atrodas šīs lapas informācija?” Atbilde „par” ir pārbaudāma bez formulas.
7.1.1.1 4	Understand what a Web browser is.	
7.1.1.1 5	Know what a Search Engine is.	Jautājums ar atbildes variantiem: „Ar meklētājprogrammu Google meklējiet atslēgas vārdus „ECDL Latvija”. Kāda lapa tiek parādīta kā pirmā?”. Varianti: a) <a href="http://www.lu.lv">www.lu.lv</a> , b) <a href="http://www.ecdl.lv">www.ecdl.lv</a> , c) <a href="http://informatika.liis.lv">informatika.liis.lv</a> .
7.1.2.1	Identify a secure Web site: https, lock symbol.	Ar jā/nē atbildams jautājums: „Dodieties uz <a href="http://www.amazon.com">www.amazon.com</a> un atveriet pirkumu grozu (sadaļa Chart lapas labajā augšējā stūrī). Vai pirkumu groza parādīšanai tiek lietots drošais https protokols?”
7.1.2.2	Know what a digital certificate for a Web site is.	Jautājums ar atbildes variantiem: „Apmeklējiet „ <a href="https://www.ecdl.lv/PasesDati">https://www.ecdl.lv/PasesDati</a> ”. Vai šim serverim ir pareizs digitālais sertifikāts”. Varanti: a) ir, b) ir, bet novecojis, c) nav. Piezīme: šī lapa uz servera ir speciāli jāizveido.
7.1.2.3	Understand the term encryption.	
7.1.2.2	Identify security threats like: virus,	Ar jā/nē atbildams jautājums ar parametru &l: „Pārlūkprogrammā atveriet hipersaiti

4	malware.	http://c:/tests/&1.exe. Vai hipersaiti atverot jūs saņemat brīdinājumu par vīrusu draudiem?" Tiek sagatavoti faili url1.exe, url2.exe... Daži ar vīrusa signatūru, citi bez. Parametrs &1 ir url1, url2... Pārbaudes formula ir: If(&1, „url1”, &0, „jā”) & If(&1, „url2”, &0, „nē”) utt. Piezīme: jautājums, protams, var būt arī vienkāršs, bez parametra, un, ja ir nepieciešams, šādi parametrizēt var daudzus jautājumus
7.1.2. 5	Identify some protection measures like: anti-virus software, firewall.	
7.1.2. 6	Identify parental control options like: supervision, Web browsing restrictions, control panel access, gaming restrictions, time limits.	Ar jā/nē atbildams jautājums: „Iespējojiet pārlūkprogrammā tīmekļa satura padomnieku ar visām noklusētajām pārbaudēm un paroli „mamma”. Vai padomnieks atļauj apskatīt lapu <a href="http://www.shrek.com">www.shrek.com</a> , neuzrādot paroli? Atpējojiet tīmekļa satura padomnieku.”
7.1.3. 1	Open (and close) a Web browsing application.	Sk. iepriekš
7.1.3. 2	Display a Web page in a new window, tab.	Ar jā/nē atbildams jautājums: „Aizejiet uz www.ecdl.lv/tables. Atveriet saites Tabula1 un Tabula2 katru savā logā. Vai skaitļi abās tabulās ir vienādi?” Piezīme. Skaitļiem tabulās ir jābūt pietiekami lieliem, lai citādā veidā tos salīdzināt būtu grūti. Praksē, protams, labāk sagatavot vairākas tabulas un lietot parametru. Vēl labāk tabulu salīdzināšanas vietā prasīt „atrodī trīs atšķirības zīmējumos”.
7.1.3. 3	Stop a Web page from downloading.	
7.1.3. 4	Refresh a Web page.	Ar jā/nē atbildams jautājums: „Aizejiet uz www.ecdl.lv/number. Vai lapā parādītais skaitlis mainās, ja lapu atjaunina?”
7.1.3. 5	Use available Help functions.	Ar frāzi atbildams jautājums: „Internet Explorer palīdzības indeksā (Help - Index) sameklējiet informāciju par grāmatzīmes ielikšanu (Favorites menu – adding items to). Kādu pogu paredzēts nospiegt otrajā solī? Ierakstiet atbildē pogas nosaukumu angļu valodā”
7.1.3. 6	Display, hide built-in toolbars, ribbon.	Ar frāzi atbildams jautājums: „Aizejiet uz www.ecdl.lv/latv1. Lai varētu redzēt tekstu, pārlēdziet kodējumu uz Baltic (Windows). Kāds ir trešais vārds tekstā?” Piezīme: lapa, protams, attiecīgi jāsatagavo.
7.1.3. 7	Display, hide images in a Web page.	Šo, manuprāt, jau sen vairs nevar izdarīt.
7.1.3. 8	Preview a Web page, print part, all of a Web page.	Praktisks uzdevums: „Aizejiet uz informatika.liis.lv un ar printeri „Microsoft Office Document Image Writer” izdrukājiet šo lapu failā c:\tests\informatika.prn”

		<p>Pārbaudes formula: IsFileText(„c:\tests\informatika.prn”, „ECDL”)                  Praktisks uzdevums: „Atveriet tīmekļa pārlūkošanas lietotni un dodieties uz tīmekļa adresi www.ecdl.lv. Saglabājiet šajā lapā esošo ECDL logo attēlu mapītē c:\tests datnē logo.gif”                  Pārbaudes formula: IsFile(„logo.gif”) &amp; IsFileSizeExactly(„logo.gif”, 1376)</p>
7.1.4. 1	Set the Web browser Home Page/Startpage.	
7.1.4. 2	Adjust browsing history settings.	
7.1.4. 3	Adjust settings for suggested matches for previous entries for Web addresses, forms, user names and passwords.	
7.1.4. 4	Allow, block pop-ups.	<p>Ar frāzi atbildams jautājums: „Aizejiet uz www.ecdl.lv/popup Atļaujiet parādīties iznirstošajam logam. Kāds teksts tiek parādīts iznirstošajā lugā?”                  Praktisks uzdevums: „Aizejiet uz www.ecdl.lv/download un lejupielādējiet tur pieejamo failu darba mapītē”. Piezīme: lapa jāiekārto tā, lai parādītos brīdinājums par lejupielādi.</p>
7.1.4. 5	Adjust settings for cookies, cache.	
7.2.1. 1	Go to a URL.	<p>Ar jā/nē atbildams jautājums: „Pārbaudiet, vai lapa www.lu.lv/&amp;1 eksistē?”                  Pārbaudes formula: If(&amp;1, „par”, &amp;0, „jā”) &amp; If(&amp;1, „ecdl”, &amp;0, „nē”) utt.</p>
7.2.1. 2	Activate a hyperlink.	<p>Jautājums ar atbildes variantiem, kas izmanto sagatavotu Word failu: ”Atveriet dokumentā esošo hipersaiti ar kārtas numuru &amp;1. Vai tā satur informāciju par ECDL?” Varianti: a) satur, b) nesatur.                  Pārbaudes formula: If(&amp;1, „1”, &amp;0, „jā”) &amp; If(&amp;1, „2”, &amp;0, „nē”) utt.</p>
7.2.1. 3	Navigate backwards and forwards between previously visited Web pages.	<p>Ar frāzi atbildams jautājums: „Aizejiet uz www.ecdl.lv/saturs.htm Pārmeklējot pēc kārtas visas grāmatas nodaļas, saskaitiet, cik nodaļu nosaukumi sākas ar „ECDL””                  Piezīme. Lapā saturs.htm sākas paliels apakšlapu koks, kuru sistemātiski pārskatīt bez pogas Back ir ļoti grūti. Ja jautājumu parametrizē, parametrs var norādīt arī uz apakškoku – viss koks nav obligāti jāveido no jauna.</p>
7.2.1. 4	Display previously visited URLs using the browser address bar, history.	

7.2.1. 5	Complete a Web-based form using: drop-down menus, list boxes, check boxes, radio buttons, text boxes.	Ar frāzi atbildams jautājums: „Lapā <a href="http://www.airbaltic.lv">www.airbaltic.lv</a> noskaidrojiet, kāds ir lidojuma numurs reisam no Rīgas uz Romu”. Piezīme: atbildīgā eksāmenā labāk izmantot speciāli sagatavotu lapu.
7.2.1. 6	Submit, reset a Web-based form.	Ar frāzi atbildams jautājums: „Atveriet lappusē <a href="http://www.ecdl.lv/gramatas">www.ecdl.lv/gramatas</a> esošo saiti „Grāmatu pasūtīšana”. Aizpildiet formu saskaņā ar instrukcijām, pasūtot &1 norādītās grāmatas eksemplārus. Kāds ir noģenerētais pasūtījuma numurs?” Pasūtīšanas formu realizējam tā, lai pareizas aizpildīšanas gadījumā tā izdotu kodu, kas atbilst noteiktam šablonam. Sadomāt var daudz ko, piemēram, <code>IsMatch(&amp;0, „*2&amp;1*”)</code> , tas ir kodā ir skaitlis 2, kam seko eksemplāru skaits
7.2.2. 1	Bookmark a Web page, RSS feed.	Ar frāzi atbildams jautājums: „Aizejiet uz <a href="http://www.ecdl.lv">www.ecdl.lv</a> un pamēģiniet atzīmēt šo lapu ar grāmatzīmi. Kādu vārdu šai grāmatzīmei sistēma piedāvā pēc noklusēšanas?”
7.2.2. 2	Display a bookmarked Web page, RSS feed.	
7.2.2. 3	Create, delete a bookmark folder.	
7.2.2. 4	Add Web pages, RSS feed to a bookmark folder.	
7.2.2. 5	Delete a bookmark.	
7.3.1. 1	Select a specific search engine.	Jautājums ar atbildes variantiem: „Ar interneta meklētāju <a href="http://scholar.google.com">scholar.google.com</a> sameklējiet, kāds ir Alberta Einšteina (Albert Einstein) internetā visbiežāk minētais darbs. Šis darbs ir:”
7.3.1. 2	Use different search sources including: encyclopaedias, RSS feeds, podcasts, blogs.	Ar frāzi atbildams jautājums: „Enciklopēdijā <a href="http://en.wikipedia.org">en.wikipedia.org</a> sameklējiet informāciju par ECDL (European Computer Driving Licence). Ar Copy un Paste iekopējiet atbildē ECDL definīcijas pirmo teikumu”
7.3.1. 3	Carry out a search for specific information using a keyword, phrase.	Ar frāzi atbildams jautājums: „Ar jebkuru no Jums zināmajiem interneta meklētājiem sameklējiet, kāds ir gaismas ātrums vakuumā un ierakstiet atbildē šo ātrumu veselos metros sekundē”
7.3.1. 4	Combine selection criteria in a search	Ar jā/nē atbildams jautājums: „Vai lapa <a href="http://ru.wikipedia.org">ru.wikipedia.org</a> ir starp pirmajām 10, kas tiek atrastas, izmantojot Google izvērsto meklēšanu starp lapām krievu valodā”
7.3.2. 1	Download files from a Web page to a location on a drive.	Praktisks uzdevums: “Aizejiet uz <a href="http://informatika.liis.lv">informatika.liis.lv</a> , sadaļā „Skolotājiem” sameklējiet ECDL mācību programmu ( <a href="#">macprog4.zip</a> ) un lejupielādējiet to mapītē <code>c:\tsts</code> ”

		Pārbaudes formula: <code>IsFileSize(„logo.gif”, 69, „K”)</code>
7.3.2. 4	Save a Web page to a location on a drive.	Praktiskais uzdevums: „Ar meklētājprogrammu Google meklējiet atslēgas vārdu „calcite” Vienu no atrastajām lapām saglabājiēt darba mapītē failā calcite.txt”. Pārbaudes formula: <code>IsFile(„calcite.txt”) &amp; IsFileText(„calcite.txt”, „calcite”)</code> – pārbaudām, vai saglabātajā failā calcīts ir vismaz pieminēts.
7.3.2. 2	Subscribe to a podcast.	
7.3.2. 3	Subscribe to an RSS feed.	

## 8. Pārbaudes funkcijas Outlook un Thunderbird vidē

E-pasta glabāšanas sistēma ir vienkārša: pasaule sastāv no mapītēm, mapītēs ir vēstules, un vēstulēm ir atribūti. Visas darbības ar vēstulēm reducējas uz to ielikšanu kādā mapītē, un uz atribūtu maiņu.

Atšķirībā no visām citām Microsoft Office un Open Office pārbaudes funkcijām, e-pasta funkcijām nav sagataves un, uzsākot uzdevuma pildīšanu, pasta kastītes saturs netiek atjaunots.

Lai labi pārbaudīti studentu prasmi darboties ar e-pastu, e-pasta programma katra studenta darba vietā pirms testa būtu jākonfigurē (jānorāda pasta serveri u.tml.), kā arī jā sagatavo un jānosūta visiem studentiem vēstules, kuras viņi testa laikā varētu saņemt. Turpmākajos piemēros tiek pieņemts, ka tas nav darīts, tāpēc zināšanu pārbaude ir daļēja un samākslota.

Šajā sistēmas Tests versijā e-pasta vidē ir tikai viena pārbaudes funkcija.

### 8.1. Funkcija *IsMessage*

Funkcija *IsMessage* pārbauda, vai norādītajā mapītē ir vēstule ar norādītajiem atribūtiem.

Funkcijai *IsMessage* pirmais arguments norāda mapīti, bet pārējie - atribūtus:

1. mapīte,
2. FROM,
3. TO,
4. CC,
5. Subject,
6. Importance,
7. Body,
8. IsSent,
9. IsRead,
10. Attachment size.

Funkcija pārbauda, vai mapītē ir vismaz viena vēstule, kurai ir visi prasītie atribūti. Mapītē drīkst būt arī citas vēstules, un drīkst būt arī vairākas vēstules ar dotajiem atribūtiem. Atribūtus var izlaist (nenorādīt), tādā gadījumā attiecīgā atribūta vērtība netiek pārbaudīta.

Tekstuālo atribūtu salīdzināšana nav case sensitive, un nebūtiskie tukšumi tiek izlaisti. Teksti nedrīkst saturēt speciālos simbolus: apostrofu, pēdiņu, semikolu un konjunktīvas zīmi.

Adreses FROM, TO un CC tiek salīdzinātas, izmantojot tikai būtisko adresi (kā jkalnins@ecdl.lv) – adresāta vārds (kā adresē „Jānis Kalniņā(jkalnins@lu.lv)”) tiek ignorēts.

Kritērijs „IsSent” ir ar vērtībām „Nē” vai „Jā”, un pārbauda, vai vēstule jebkad ir bijusi nosūtīta (tai ir Sent date).

Kritērija Attachment Size prasa, lai būtu tieši viens piekabinājums un tieši noteiktajā garumā (baitos).

Zemāk doti vairāki piemēri, kā ar sistēmas Tests līdzekļiem varētu pārbaudīt ECDL 7. moduļa 2. nodaļā „Communication” prasītās zināšanas. Piemēri veidoti pieņemot, ka Outlook ir nokonfigurēts ar pareizām pasta serveru adresēm, un pasta nosūtīšana ir iespējama. Pasta

saņemšana piemēros netiek izmantota, tāpēc visas darba vietas var būt konfigurētas ar precīzi vienādiem E-mail Account.

REF.	Task Item	Uzdevuma vai jautājuma piemērs
7.4.1.1	Understand the make-up and structure of an e-mail address.	Praktisks uzdevums: “Nosūtiet jebkādu e-pasta vēstuli lietotājam ar vārdu „ecdl”, kura e-pasta serveris ir „ecdl.lv”” Pārbaudes formula: IsMail(„Sent Items”, , „ecdl@ecdl.lv”)
7.4.1.2	Understand the advantages of e-mail systems like: speed of delivery, low cost, flexibility of using a Web-based e-mail account in different locations.	
7.4.1.3	Understand the importance of network etiquette (netiquette) like: using accurate descriptions in e-mail message subject fields, brevity in e-mail responses, spell checking outgoing e-mail.	Jautājums ar atbildes variantiem: “Kura no trim vēstulēm ienākošajā pastkastē labāk atbilst interneta pieklājības normām?” Varianti (piemēram): a) Uzmanieties, vīruss, b) Nav labi!!!!!! c) Brīdinājums par vīrusu Praktisks uzdevums: “Atveriet vēstuli no mapītes „Drafts”, izlabojiet acīmredzamos etiķetes pārkāpumus un nosūtiet vēstuli adresātam” Pārbaudes formula: IsMail(„Sent Items”, , „ecdl@ecdl.lv”, , „ECDL eksāmena rezultāts”) – etiķetes pārkāpums varēja būt gramatikas kļūda Subject laukā
7.4.1.4	Be aware of possible problems when sending file attachments like: file size, file type e.g .exe file)	Ar frāzi atbildams jautājums: „Cik liels (veselos kilobaitos) ir vēstulei „Atskaites melnraksts” piekabinātais dokuments?” Ar frāzi atbildams jautājums: „Kāds ir vēstulei „Jauns iespējas” piekabinātā faila paplašinājums” Piezīme: failu var nosaukt „Kaut kas nebūt.DOC .exe” – kā daždien pie Trojas zirgiem.
7.4.2.1	Know what unsolicited e-mail is. Know about filtering measures.	Ar frāzi atbildams jautājums: „Vēstules neatverot novērtējiet, cik no ienākošā pasta kastē esošajām vēstulēm ir potenciālas mēstules?” Pārbaudes formula: IsMatch(&0, „2”) & ! IsMail(„Inbox”, , , „Need medicine? Best dr@gs at best store!” , , „Jā”) & utt. Praktisks uzdevums: „Izdzēsiet visas vēstules, kuras visdrīzāk ir mēstules” Pārbaudes formula: ! IsMail(„Inbox”, , , „Need medicine? Best dr@gs at best store!”) & utt.
7.4.2.2	Be aware of the danger of infecting the computer with a virus by opening an e-mail	Ar jā/nē atbildams jautājums: „Vai ir droši atvērt šai vēstulei &1 piekabināto failu?”



	message, opening an attachment.	Pārbaudes formula: If(&1, „Atskaite”, &0, „Jā”) & If(&1, „Dāvana”, &0, „Nē”) & utt. Bīstamos failus jāsauc Trojas zirgiem tipiskos vārdos.
7.4.2.3	Understand the term ‘phishing’. Identify signs of attempted ‘phishing’.	Ar jā/nē atbildams jautājums: „Vai Jūs atbildētu uz vēstuli %1?” Pārbaudes formula: If(&1, „Problēma ar Jūsu kredītkarti”, &0, „Jā”) & If(&1, „Problēma bankas datu bāzē”, &0, „Nē”) Pirmā vēstule varētu lūgt pagarinājumu kredītkartes izgatavošanas termiņam, bet otrā – prasīt kartes informāciju.
7.4.2.4	Understand what a digital signature is.	
7.4.3.1	Open (and close) an e-mail application.	Praktisks uzdevums Windows vidē: „Palaidiet programmu Outlook un atstājiet to palaistu” Pārbaudes formula: IsProgrammStarted(„Outlook”, „Jā”)
7.4.3.2	Create a new e-mail account.	Praktisks uzdevums: „Nomainiet e-pasta konfigurācijā savu e-pasta adresi no student@ecd1.lv uz smart.student@ecd1.lv Nosūtiet jebkādu vēstuli uz adresi ecd1@ecd1.lv” Pārbaudes formula: IsMail(„Sent Items”, „smart.student@ecd1.lv”) Ar uzdevumu nedaudz jāuzmanās, to nevar likt vienā darbā ar citu uzdevumu, kurš pārbaudei izmanto sākotnējo sūtītāja adresi student@ecd1.lv
7.4.3.3	Open (and close) a message.	Ar frāzi atbildams jautājums: „Kāds ir trešais vārds vēstulē &1?”
7.4.3.4	Use available Help functions.	Jautājums ar atbilžu variantiem: ”Izmantojot Help noskaidrojiet vienu ticamu iemeslu, kāpēc rakstot vēstuli var nebūt pieejama Send poga (Send button missing” Varianti: a) nav ievadīta saņēmēja adrese, b) nav izveidots e-pasta rēķins, 3) datoram nav pieejams internets.
7.4.3.5	Print a message.	Praktisks uzdevums: „Ar printeri „Microsoft Office Document Image Writer” izdrukājiet vēstuli „ECDL eksāmens” failā c:\tests\informatika.prn” Pārbaudes formula: IsFileText(„c:\tests\informatika.prn”, „ECDL”)
7.4.4.1	Add, remove message inbox headings like: sender, subject, date received.	Ar jā/nē atbildams jautājums: „Neatverot vēstuli, noskaidrojiet, vai vēstulei „Need medicine? Best dr@gs at best store!” ir aizpildīts CC lauks” Pārbaudes formula: IsMatch(&0, „Nē”) & ! IsMail(„Inbox”, , , „Need medicine? Best dr@gs at best store!” , „Jā”) Jautājums ar atbilžu variantiem: „Kura no pastkastē esošajām vēstulēm ir

		saņemta pati pēdējā?" Jābūt kādām 20 vēstulēm, lai bez kolonas kārtošanas uzdevums būtu ļoti grūti izpildāms.
7.4.4. 2	Display, hide built-in toolbars, ribbon.	
7.4.4. 3	Set option to reply with, without original message insertion.	
7.5.1. 1	Flag a mail message. Remove a flag mark from a mail message.	
7.5.1. 2	Identify a message as read, unread. Mark a message as unread, read.	Praktisks uzdevums: „Atzīmējiet vēstuli „ECDL eksāmens” kā neizlasītu”
7.5.1. 3	Save a file attachment to a location on a drive and open the file.	Praktisks uzdevums: „Saglabājiet vēstulei „Atskaites melnraksts” piekabināto failu mapītē c:\tests” Pārbaudes formula: IsFile(„c:\tests\Atskaite.doc”)
7.5.1. 4	Use the reply, reply to all function.	Praktisks uzdevums: „Atbildiet uz vēstuli „Izdevīgs piedāvājums” ar vienu vārdu „Nē”” Pārbaudes formula: IsMail(„Sent Items”, , , , „RE: Izdevīgs piedāvājums”)
7.5.1. 5	Forward a message.	Sk. 7.5.1.4
7.5.2. 1	Create a new message.	Sk. iepriekš.
7.5.2. 2	Insert an email address in the ‘To’ field, Copy (Cc) field, blind copy (Bcc) field.	Sk. iepriekš. BCC nav, un nevajag arī.
7.5.2. 3	Insert a title in the ‘Subject’ field.	Sk. iepriekš.
7.5.2. 4	Insert text with basic formatting into an email.	
7.5.2. 5	Copy text from another source into a message.	Praktisks uzdevums: „Nosūtiet adresātam ecld@ecdl.lv tieši tādu pat vēstuli, kāda ir nosūtīta adresātam icdl@ecdl.lv (vēstule ir saglabāta mapītē „Sent Items””
7.5.2. 6	Insert, remove a file attachment.	Praktisks uzdevums: „Nosūtiet adresātam ecld@ecdl.lv vēstuli, kurai piekabināts fails „c:\tests\atskaite.doc””

		<p>Pārbaudes formula: IsMail(„Sent Items”,,,,,, 256) – tiek pārbaudīts tikai piekabinātā faila lielums                  Praktisks uzdevums: „Nodēsiet piekabināto failu vēstulei „Atskaite” un saglabājiēt šo vēstuli bez faila”                  Pārbaudes formula: !IsMail(„Inbox”,,,,,, 256) &amp; IsMail(„Inbox”,,, „Atskaite”)</p>
7.5.2. 7	Set a message priority.	Sk. IsMessage argumentu Importance.
7.5.2. 9	Use a spell-checking tool and make changes like: correcting spelling errors, deleting repeated words.	
7.5.2. 8	Save a draft of a message.	Pārbaudes formula: IsMail(„Drafts”,,, „Atskaite”)
7.5.2. 10	Send a message.	Sk. iepriekš.
7.6.1. 1	Search for a message by sender, subject, mail content.	Jautājums ar atbilžu variantiem: „Kurā no vēstulēm ir pieminēts vārds „ECDL”?” Piezīme: jābūt vismaz 20 pagaram vēstulēm, lai bez search nevarētu iztikt
7.6.1. 2	Sort messages by name, by date, by size.	Jautājums ar atbilžu variantiem: „Kura no pastkastē esošajām vēstulēm ir saņemta pati pēdējā?” Jābūt kādām 20 vēstulēm, lai bez kolonas kārtšanas uzdevums būtu ļoti grūti izpildāms.
7.6.1. 3	Create, delete an e-mail folder.	Praktisks uzdevums: ”Izveidojiēt mapīti „Urgent” un ielieciēt tajā vēstuli „Atskaite”” Pārbaudes formula: IsMail(„Urgent”,,, „Atskaite”)
7.6.1. 4	Move messages to an e-mail folder.	Sk. 7.6.1.3.
7.6.1. 5	Delete a message.	Sk. iepriekš.
7.6.1. 6	Restore a message from the mail bin/deleted items folder.	Sk. iepriekš.
7.6.1. 7	Empty the mail bin/deleted items folder.	Sk. iepriekš.

7.6.2. 1	Create a new address list/distribution list.	
7.6.2. 2	Add a mail address to an address list.	
7.6.2. 3	Delete a mail address from an address list.	
7.6.2. 4	Update an address book from incoming mail.	