**DAUGAVPILS UNIVERSITĀTES**

**STUDIJU KURSA APRAKSTS**

|  |  |
| --- | --- |
| Studiju kursa nosaukums | ***Programmēšanas valoda C# DP [PBSP IT, PROGINZ]*** |
| Studiju kursa kods (DUIS) | DatZ2054 |
| Zinātnes nozare | #Datorzinātne  #Programmēšanas valodas un sistēmas |
| Kursa līmenis | 2 |
| Kredītpunkti | 4 |
| ECTS kredītpunkti | 6 |
| Kopējais kontaktstundu skaits | 64 |
| Lekciju stundu skaits |  |
| Semināru stundu skaits |  |
| Praktisko darbu stundu skaits | 64 |
| Laboratorijas darbu stundu skaits |  |
| Studējošā patstāvīgā darba stundu skaits | 96 |
|  | |
| Kursa autors(-i) | |
| Mg.sc.comp., lekt. Olga Perevalova  Mg.sc.comp., lekt. Andrejs Radionovs | |
| Kursa docētājs(-i) | |
| Mg.sc.comp., lekt. Olga Perevalova  Mg.sc.comp., lekt. Andrejs Radionovs | |
| Priekšzināšanas | |
| DatZ1049, Programmēšanas pamati (*C++*) I [PBSP IT, PROGINZ] | |
| Studiju kursa anotācija | |
| KURSA MĒRĶIS:  Iepazīstināt studējošos ar programmēšanas valodu *MS Visual* *C#* no *OOP* viedokļa.  KURSA UZDEVUMI:  • Sniegt priekšstatu par platformām *.NET* un *.Net Core*, izstrādes vidi *Visual Studio* un programmēšanas valodu *MS Visual C#*.  • Pievērst studentu uzmanību mūsdienīgas *UI/UE* izstrādei.  • Sniegt priekšstatu par funkciju, klašu, klašu hierarhiju, klašu bibliotēku un komponentu veidošanu.  • Parādīt, kā var risināt uzdevumus un pilnveidot kodu, izmantojot dažādas pieejas.  • Veicināt studentus patstāvīgi risināt uzdevumus no dažādām jomām, izstrādāt un pilnveidot nelielus projektus, izmantojot OOP, domāt par koda efektivitāti, lasāmību un modificēšanas iespējām. | |
| Studiju kursa kalendārais plāns | |
| |  |  |  |  | | --- | --- | --- | --- | | *Nr* | *Temats* | *Stundu*  *skaits* | *Nodarbību*  *veids* | | 1. | Platformu *.NET* un *.Net Core* vispārīgs raksturojums. Jēdziens par *IL=CIL=MSIL (Common Intermediate Language)* un asambleju=montāžu (*Assembly*).  Valodas vispārīgs *C#* raksturojums.  *Visual Studio* vide: vārdu telpa (*namespace*), jēdziens par klasēm un objektiem. Risinājums (*solution*), projekts (*project*), norāde (*reference*). Lāgošanas (*Debug*) un izlaiduma (*Release*) režīmi.  Vienkāršas programmas struktūra. Konsoles lietotne (*Console* *Application*). Metode *Main*(). Klases *Console* un *Math*. | 2 | P | | 3 | Pd | | 2. | Jēdziens par notikum-orientēto programmēšanu. Vizuālie un nevizuālie komponenti. Komponentu īpašības (*properties*), metodes (*methodes*) un notikumi (*events*).  *Windows* lietotne (*Windows Forms Application*).  Datu pamattipi. Konstantes. Mainīgie. Mainīgo inicializācija. Modernie identifikatoru rakstīšanas noteikumi.  Aritmētiskās operācijas (operators), piešķires operācija (*operator assignment*), aritmētiskās izteiksmes, operāciju prioritāte, matemātiskās funkcijas.  Aprēķini pēc formulām. Konteineru izmantošana komponentu grupēšanai. | 2 | P | | 3 | Pd | | 3. | Datu izvietošana atmiņā. Vērtību un norāžu tipi. Steks. Kaudze. Drazu savākšana.  Operatori (*statements*): sazarošanās (*if, switch*) un cikli (*for, while, do-while, foreach*).  Izņēmumu apstrāde (*try-catch*).  Aprēķini pēc nosacījumiem. | 2 | P | | 3 | Pd | | 4. | *Timer*. Nejaušo skaitļu izmantošana.  Komponenti lietotāja izvēles nodrošināšanai.  Dažādu veidu testu realizācija.  Klase *Graphics*. | 2 | P | | 3 | Pd | | 5. | Cikli. Skaitļu secības. Funkciju tabulēšana.  Līklīniju konstruēšana. | 2 | P | | 3 | Pd | | 6. | *RichTextBox* kā simbolu virkņu masīvs.  Ievadīto datu analīze. | 2 | P | | 3 | Pd | | 7. | **P/d "Pamati"**.  Mini-projekta "Grafika" nodošana. | 2 | P | | 3 | Pd | | 8. | Masīvi. Klase *Array*. Skaitļu masīvu apstrāde. | 3 | P | | 2 | Pd | | 9. | Saraksti (*List, ListArray*), vārdnīca (*Dictionary*). | 2 | P | | 3 | Pd | | 10. | Rinda (*Queue*). Steks (*Stack*). Kolekcijas (*Collection*).  Iebūvēto datu struktūru un algoritmu izmantošana. | 2 | P | | 3 | Pd | | 11. | Vizuālo komponentu masīvi, to dinamiskā izveide. | 2 | P | | 3 | Pd | | 12. | **P/d "Masīvi - min"**.  Mini-projekta "Datu struktūras" nodošana. | 2 | P | | 3 | Pd | | 13. | Simbolu virknes. Klase *String*. Simbolu virkņu masīvi. | 2 | P | | 3 | Pd | | 14. | Regulārās izteiksmes. | 2 | P | | 3 | Pd | | 15. | **P/d "Masīvi un simbolu virknes".**  Mini-projekta "Teksta apstrāde" nodošana. | 2 | P | | 3 | Pd | | 16. | Ieskaites jautājumu apskats, aplūkotā materiāla apkopojums, pašnovērtējums, refleksija. | 2 | P | | 3 | Pd | |  |  |  |  | | 17. | Metožu (funkciju) izmantošana un definēšana. Atgriežamās vērtības. Lokālie un sadalītie mainīgie. Redzamības apgabals. Metožu signatūra. Metožu parametri. Formālie un faktiskie mainīgie. Parametri-vērtības, parametri-norādes (*ref*) un parametri-rezultāti (*out*). Metodes ar mainīgu parametru skaitu (*params*). | 2 | P | | 3 | Pd | | 18. | Masīvi metodēs. Masīvi kā parametri. Masīvi kā atgriežamās vērtības. Metodes ar mainīgu parametru skaitu. Metodes *Main*() parametri (komandrindas saņemšana). Komandrindas parametri. Metožu pārslogošana. | 2 | P | | 3 | Pd | | 19. | Rekursija. Izņēmumu radīšana metodēs (*throw Exception*). Datu apmaiņa starp projekta formām. | 2 | P | | 3 | Pd | | 20. | Delegāti (*delegate*) un to izmantošana. Vispārinātās (universālās) metodes (*generic*). | 2 | P | | 3 | Pd | | 21. | **P/d "Metodes"**.  Mini-projekta "Game I" nodošana (ar metodēm). | 2 | P | | 3 | Pd | | 22. | Klases un objekti. Iekapsulēšana, datu abstrakcija, realizācijas slēpšana. Klases lauki un metodes. Pieejas modifikatori. Lauku inicializācija. Konstruktori. Metožu un operāciju pārslogošana. Rādītājs uz izsaucošo objektu (*this*). | 2 | P | | 3 | Pd | | 23. | Statiskie klases locekļi.  Īpašības. Īpašību pārslogošana.  Klašu (*class*) un struktūru (*struct*) izmantošana. | 2 | P | | 3 | Pd | | 24. | **P/d "Klases un objekti".** | 3 | P | | 2 | Pd | | 25. | Mantojamība. Klašu hierarhija. Abstraktās klases (*abstract*). Interfeisi (*interface*). Izolētas klases (*sealed*). Ieliktās klases (*nested*). Dalītās klases (*partial*). | 2 | P | | 3 | Pd | | 26. | Polimorfisms. Klase *Object*. Tipu iepakojums (*boxing*), izpakojums (*unboxing*). Vispārinātās (universālās) klases, jeb klases ar dzimtparametriem (*generic*).  Klašu hierarhijas analīze un optimizācija. | 2 | P | | 3 | Pd | | 27. | **P/d "Klašu hierarhija".** | 2 | P | | 3 | Pd | | 28. | Dinamiski pieslēdzamo bibliotēku (*dynamic-link library, dll*) izveide | 2 | P | | 3 | Pd | | 29. | Darbs ar datnēm. Klases *Directory* un *File*. Klases *StreamWriter* un *StreamReader.* | 2 | P | | 3 | Pd | | 30. | Datne kā baitu plūsma. Klase *FileStream*. Serializācija. | 2 | P | | 3 | Pd | | 31. | Mini-projekta "Game II" nodošana (ar OOP un datnēm). | 2 | P | | 3 | Pd | | 32. | Eksāmena jautājumu apskats, aplūkotā materiāla apkopojums, pašnovērtējums, refleksija. | 2 | P | | 3 | Pd | | |
| Studiju rezultāti | |
| ZINĀŠANAS:  1. Zina platformu *.NET* un *.Net Core*, izstrādes vides *MS Visual Studio* vidi un programmēšanas valodas *MS Visual C#* pamatprincipus. 2. Zina programmēšanas valodas *MS Visual C#* populārākos datu tipus un datu struktūras un vadības konstrukcijas.  3. Zina objektorientētās, vizuālās un notikumorientētas programmēšanas pamatprincipus.  PRASMES:  4. Veido un modificē nelielus projektus. 5. Risina standarta uzdevumus.  6. Veido un izmanto metodes (funkcijas).  7. Veido un izmanto klases un klašu hierarhijas.  KOMPETENCES:  8. Veido gaumīgu un ērtu lietotāja saskarni (*UI*).  9. Pielieto lāgošanas līdzekļus, pievērš uzmanību programmēšanas stilam.  10. Strādā ar speciālo literatūru, izmanto meklēšanas serverus un *help*-sistēmas. | |
| Studējošo patstāvīgo darbu organizācijas un uzdevumu raksturojums | |
| Katrā nodarbībā studējošiem tiek rekomendēti konkrēti uzdevumi no pasniedzēja izstrādātā uzdevumu krājuma, kas atbilst nodarbībā aplūkotajai tēmai. Vairākas reizes semestrī studējošiem ir jāizveido mini-projekts, kas tiks vērtēts ar i/ni vai ar atzīmi. | |
| Prasības kredītpunktu iegūšanai | |
| STUDIJU REZULTĀTU VĒRTĒŠANA  Studiju kursa apguve tā noslēgumā tiek vērtēta 10 ballu skalā saskaņā ar Latvijas Republikas normatīvajiem aktiem un atbilstoši "Nolikumam par studijām Daugavpils Universitātē" (apstiprināts DU Senāta sēdē 17.12.2018., protokols Nr. 15), vadoties pēc šādiem kritērijiem: iegūto zināšanu apjoms un kvalitāte, iegūtās prasmes un kompetence atbilstoši plānotajiem studiju rezultātiem.   |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Pārbaudījumu veidi | Studiju rezultāti | | | | | | |  |  |  | | 1. | 2. | 3. | 4. | 5. | 6. | 7. | 8. | 9. | 10. | | 1.1. pārbaudes darbs (rakstiski) |  | + |  |  | + |  |  | + | + |  | | 1.1. mini-projekta nodošana (mutiski) |  | + |  | + |  |  |  | + | + | + | | 1.2. pārbaudes darbs (rakstiski) |  | + |  |  | + |  |  | + | + |  | | 1.2. mini-projekta nodošana (mutiski) |  | + |  | + |  |  |  | + | + | + | | 1.3. pārbaudes darbs (rakstiski) |  | + |  |  | + |  |  | + | + |  | | 1.3. mini-projekta nodošana (mutiski) |  | + |  | + |  |  |  | + | + | + | | Dif. ieskaite (mutiski) | + | + |  |  |  |  |  | + | + |  | | 2.1. pārbaudes darbs (rakstiski) |  |  |  |  |  | + |  |  | + |  | | 2.1. mini-projekta nodošana (mutiski) |  |  | + | + |  | + |  | + | + | + | | 2.2. pārbaudes darbs (rakstiski) |  |  |  |  |  |  | + |  | + |  | | 2.3. pārbaudes darbs (rakstiski) |  |  | + |  |  |  | + |  | + |  | | 2.2. mini-projekta nodošana (mutiski) |  |  | + | + |  |  | + | + | + | + | | Eksāmens (mutiski) |  |  | + |  |  | + | + |  | + |  | | |
| Kursa saturs | |
| Praktiskie darbi - 64 st.,  Patstāvīgais darbs - 92 st.   Praktisko darbu tēmas:  1. Platformu *.NET* un *.Net Core* vispārīgs raksturojums. Jēdziens par *IL=CIL=MSIL (Common Intermediate Language)* un asambleju=montāžu (*Assembly*). Valodas vispārīgs *C#* raksturojums.  *Visual Studio* vide: vārdu telpa (*namespace*), jēdziens par klasēm un objektiem. Risinājums (*solution*), projekts (*project*), norāde (*reference*). Lāgošanas (*Debug*) un izlaiduma (*Release*) režīmi. Vienkāršas programmas struktūra. Konsoles lietotne (*Console* *Application*). Metode *Main*(). Klases *Console* un *Math*.  2. Jēdziens par notikum-orientēto programmēšanu. Vizuālie un nevizuālie komponenti. Komponentu īpašības (*properties*), metodes (*methodes*) un notikumi (*events*).  *Windows* lietotne (*Windows Forms Application*).  Datu pamattipi. Konstantes. Mainīgie. Mainīgo inicializācija. Modernie identifikatoru rakstīšanas noteikumi. Aritmētiskās operācijas (operators), piešķires operācija (*operator assignment*), aritmētiskās izteiksmes, operāciju prioritāte, matemātiskās funkcijas.  Aprēķini pēc formulām. Konteineru izmantošana komponentu grupēšanai.  3. Datu izvietošana atmiņā. Vērtību un norāžu tipi. Steks. Kaudze. Drazu savākšana.  Operatori (*statements*): sazarošanās (*if, switch*) un cikli (*for, while, do-while, foreach*).  Izņēmumu apstrāde (*try-catch*). Aprēķini pēc nosacījumiem.  4. *Timer*. Nejaušo skaitļu izmantošana. Komponenti lietotāja izvēles nodrošināšanai.  Dažādu veidu testu realizācija. Klase *Graphics*.  5. Cikli. Skaitļu secības. Funkciju tabulēšana. Līklīniju konstruēšana.  6. *RichTextBox* kā simbolu virkņu masīvs. Ievadīto datu analīze.  7. **P/d "Pamati"**. Mini-projekta "Grafika" nodošana.  8. Masīvi. Klase *Array*. Skaitļu masīvu apstrāde.  9. Saraksti (*List, ListArray*), vārdnīca (*Dictionary*).  10. Rinda (*Queue*). Steks (*Stack*). Kolekcijas (*Collection*). Iebūvēto datu struktūru un algoritmu izmantošana.  11. Vizuālo komponentu masīvi, to dinamiskā izveide.  12. **P/d "Masīvi - min"**. Mini-projekta "Datu struktūras" nodošana.  13. Simbolu virknes. Klase *String*. Simbolu virkņu masīvi.  14. Regulārās izteiksmes.  15. **P/d "Masīvi un simbolu virknes"**.Mini-projekta "Teksta apstrāde" nodošana.  16. Ieskaites jautājumu apskats, aplūkotā materiāla apkopojums, pašnovērtējums, reflekisja.  17. Metožu (funkciju) izmantošana un definēšana. Atgriežamās vērtības. Lokālie un sadalītie mainīgie. Redzamības apgabals. Metožu signatūra. Metožu parametri. Formālie un faktiskie mainīgie. Parametri-vērtības, parametri-norādes (*ref*) un parametri-rezultāti (*out*). Metodes ar mainīgu parametru skaitu (*params*).  18. Masīvi metodēs. Masīvi kā parametri. Masīvi kā atgriežamās vērtības. Metodes ar mainīgu parametru skaitu. Metodes *Main*() parametri (komandrindas saņemšana). Komandrindas parametri. Metožu pārslogošana.  19. Rekursija. Izņēmumu radīšana metodēs (*throw Exception*). Datu apmaiņa starp projekta formām.  20. Delegāti (*delegate*) un to izmantošana. Vispārinātās (universālās) metodes (*generic*).  21. **P/d "Metodes"**. Mini-projekta "Game I" nodošana (ar metodēm).  22. Klases un objekti. Iekapsulēšana, datu abstrakcija, realizācijas slēpšana. Klases lauki un metodes. Pieejas modifikatori. Lauku inicializācija. Konstruktori. Metožu un operāciju pārslogošana. Rādītājs uz izsaucošo objektu (*this*).  23. Statiskie klases locekļi. Īpašības. Īpašību pārslogošana. Klašu (*class*) un struktūru (*struct*) izmantošana.  24. **P/d "Klases un objekti".**  25. Mantojamība. Klašu hierarhija. Abstraktās klases (*abstract*). Interfeisi (*interface*). Izolētas klases (*sealed*). Ieliktās klases (*nested*). Dalītās klases (*partial*).  26. Polimorfisms. Klase *Object*. Tipu iepakojums (*boxing*), izpakojums (*unboxing*). Vispārinātās (universālās) klases, jeb klases ar dzimtparametriem (*generic*). Klašu hierarhijas analīze un optimizācija.  27. **P/d "Klašu hierarhija".**  28. Dinamiski pieslēdzamo bibliotēku (*dynamic-link library, dll*) izveide  29. Darbs ar datnēm. Klases *Directory* un *File*. Klases *StreamWriter* un *StreamReader.*  30. Datne kā baitu plūsma. Klase *FileStream*. Serializācija.  31. Mini-projekta "Game II" nodošana (ar OOP un datnēm).  32. Eksāmena jautājumu apskats, aplūkotā materiāla apkopojums, pašnovērtējums, refleksija.    Patstāvīgais darbs:  Katrā nodarbībā studējošiem tiek rekomendēti konkrēti uzdevumi no pasniedzēja izstrādātā uzdevumu krājuma, kas atbilst nodarbībā aplūkotajai tēmai. Vairākas reizes semestrī studējošiem ir jāizveido mini-projekts, kas tiks vērtēts ar i/ni vai ar atzīmi. | |
| Obligāti izmantojamie informācijas avoti | |
| 1. O.Perevalova. C#: konspektu un uzdevumu komplekts, sagataves un prasības projektiem. URL:  *https://estudijas.du.lv/course/view.php?id=1269* 2. Andrew Troelsen, Philip Japikse. Pro C# 7 with .NET and .NET Core.: Apress, 2018. URL: *https://dl.ebooksworld.ir/motoman/Apress.Pro.Csharp.7.With.NET.and.NET.Core.www.EBooksWorld.ir.pdf*  RU: Эндрю Троелсен, Филипп Джепикс. Язык программирования C# 7 и платформы .NET и .NET Core.: Диалектика, 2018. URL: *https://sd.blackball.lv/books/17526?mode=read*  3. Mark J. Price. C# 12 and .NET 8 – Modern Cross-Platform Development Fundamentals. Packt Publishing, 2023. - URL: *https://dl.ebooksworld.ir/books/CSharp.12.and.NET.8.9781837635870.EBooksWorld.ir.pdf*  4. Stack Overflow Documentation. Learning C# Language (free PDF), 2019. Url: *https://www.computer-pdf.com/programming/csharp/879-tutorial-learning-c-language.html*  ~~1. O.Perevalova. C#: konspektu un uzdevumu komplekts, sagataves un prasības projektiem. URL:~~  *~~https://estudijas.du.lv/course/view.php?id=1269~~* ~~2. Andrew Troelsen, Philip Japikse. Pro C# 7 with .NET and .NET Core.: Apress, 2018. URL:~~ *~~https://dl.ebooksworld.ir/motoman/Apress.Pro.Csharp.7.With.NET.and.NET.Core.www.EBooksWorld.ir.pdf~~*  ~~RU: Эндрю Троелсен, Филипп Джепикс. Язык программирования C# 7 и платформы .NET и .NET Core.: Диалектика, 2018. URL:~~ *~~https://sd.blackball.lv/books/17526?mode=read~~*  ~~3. Herbert Schildt. C# 4.0: The Complete Reference.: Mc Graw Hill, 2010. URL:~~ *~~https://memberfiles.~~*  *~~freewebs.com/02/83/78118302/documents/McGraw.Hill.CSharp.4.0.The.Complete.Reference.Apr.2010.pdf~~*  ~~RU: Г.Шилдт. C# 4.0: полное руководство.: Вильямс, 2011. URL:~~ *~~http://ijevanlib.ysu.am/wp-content/uploads/2017/12/ C-4.0-полное-руководство-Герберт-Шилдт\_2011.pdf~~*  ~~4. Stack Overflow Documentation. Learning C# Language (free PDF), 2019. URL:~~ *~~https://www.computer-pdf.com/programming/csharp/879-tutorial-learning-c-language.html~~* | |
| Papildus informācijas avoti | |
| 1. C# Tutorial. - © TutorialPoint 2024., URL: *https://www.tutorialspoint.com/csharp/index.htm*  2. C# documentation. © MS 2024., URL: *https://docs.microsoft.com/en-us/dotnet/csharp/*  RU: Документация по *C*#. URL: *https://docs.microsoft.com/ru-ru/dotnet/csharp/*  3. Mark J. Price. Apps and Services with .NET 8: Build practical projects with Blazor, .NET MAUI, gRPC, GraphQL, and other enterprise technologies. Packt Publishing, 2023.  4. Jon Skeet. C# in Depth. - 2019. URL: https://www.manning.com/books/c-sharp-in-depth-fourth-edition  RU: Джон Скит. C# для профессионалов. Тонкости программирования. - Вильямс, 2019.  5. RU: Р.Мартин. Чистый код: создание, анализ и рефакторинг. - Питер, 2019. URL: *https://ru.pdfdrive.com/Чистый-код-создание-анализ-и-рефакторинг-e188599881.html*  ~~1. C# Tutorial. URL:~~ *~~https://www.tutorialspoint.com/csharp/index.htm~~*  ~~2. C# documentation. URL:~~ *~~https://docs.microsoft.com/en-us/dotnet/csharp/~~*  ~~RU: Документация по~~ *~~C~~*~~#. URL:~~ *~~https://docs.microsoft.com/ru-ru/dotnet/csharp/~~*  ~~3. Steve McConnell. Code Complete. - MS Press, 2004. URL:~~ *~~http://aroma.vn/web/wp-content/uploads/2016/11/code-complete-2nd-edition-v413hav.pdf~~* ~~RU: С.Макконнелл. Совершенный код. - Русская редакция, 2010. URL:~~ *~~https://fktpm.ru/file/84-soversennyi-kod.pdf~~*  ~~4. Robert C. Martin. Clean Code: A Handbook of Agile Software Craftsmanship. - Pearson Education, 2009. URL:~~ *~~https://www.pdfdrive.com/a-handbook-of-agile-software-craftsmanship-by-robert-c-martinpdf-e32317666.html~~* ~~RU: Р.Мартин. Чистый код: создание, анализ и рефакторинг. - Питер, 2019. URL:~~ *~~https://ru.pdfdrive.com/Чистый-код-создание-анализ-и-рефакторинг-e188599881.html~~* | |
| Periodika un citi informācijas avoti | |
| 1. Welcome to the Unity Scripting Reference! URL: *https://docs.unity3d.com/ScriptReference/*  2. Ферроне Харрисон. Изучаем C# через разработку игр на Unity. URL: *https://flibusta.site/b/657687*  3. MS Developer Network forums. URL: *https://social.msdn.microsoft.com/Forums*  4. RIP Tutorial (Tags, Topics, Examples). URL: *https://riptutorial.com/* | |
| Piezīmes | |
| Profesionālās augstākās izglītības bakalaura studiju programmas “Informācijas tehnoloģijas” studiju kurss. Kurss tiek docēts latviešu valodā. | |