

IZGLĪTĪBAS IESTĀDE
PROGRAMMAS VEIDS
PROGRAMMU KOPA
IEGŪSTAMĀ KVALIFIKĀCIJA
IEPRIEKŠĒJĀ IZGLĪTĪBA
ĪSTENOŠANAS ILGUMS
IEGUVES FORMA

Vidzemes Tehnoloģiju un dizaina tehnikums
Profesionālās vidējās izglītības programma
Programmēšana
Programmēšanas tehniks, 4. LKI līmenis
Pamatizglītība
Četri gadi (5768 stundas)
Klātie

APSTIPRINU

Vidzemes Tehnoloģiju un
dizaina tehnikuma direktors

_____/ A. Sņegovičs
2022. gada 31. augustā

Moduļa programma PROGRAMMĒŠANAS TEHNIKĀ PRAKSE UZNĒMUMĀ

Apjoms stundās: 720 stundas

Mērķis: nostiprināt un pilnveidot izglītojamā spējas izstrādāt programmatūru, pārbaudīt tās darbību un veikt atbilstošas izmaiņas programmatūras uzlabošanai, izveidojot funkcionālu un izvirzītajām prasībām atbilstošu programmatūras risinājumu vai tā komponentes darba vidē.

Uzdevumi:

1. pirms darba uzdevumu izpildes uzsākšanas izvērtēt un novērst darba vides riska faktorus, iekārtojot darba vietu atbilstoši ergonomikas prasībām;
2. izveidot programmatūras vienību, lietojot vienu vai vairākas uzdevumam atbilstošas programmēšanas valodas, ievērojot programmatūras izstrādes metodoloģiju un labo praksi;
3. lasīt un analizēt esošu programmatūras vienību pirmkodu, uzrakstīt programmatūras vienību, pārbaudīt tās atbilstību lietotāja vajadzībām, veicot testēšanu;
4. izvērtēt programmatūras vienības pirmkoda struktūras atbilstību tā risināmajam uzdevumam un veikt koda uzlabošanu, neietekmējot programmatūras funkcionalitāti (refaktorēšana);
5. analizēt, veidot un uzlabot datu plūsmu un glabāšanu, izmantojot atbilstošus datu pārvaldības risinājumus un datu bāzes;
6. testēt programmatūras atbilstību tehniskajai dokumentācijai, novērst savstarpējās atšķirības, kā arī novērst programmatūras lietotāja konstatētas kļūdas un to cēloņus, saskaņojot iznākumu ar tehnisko dokumentāciju;
7. atbilstoši dotajiem kritērijiem aizpildīt prakses dokumentāciju, izvērtējot darba rezultātus un izvirzot ieteikumus tālākai darbībai, kā arī iepazīstināt ar saviem sasniegumiem auditoriju.

Nosacījumi pielaidei pie aizstāvēšanas:

izglītojamais atbilstoši prakses programmai ir iesniedzis:

- 1.1. elektroniski prakses ikdienas darbu aprakstu jeb prakses dienasgrāmatu (jāpilda katru darba dienu),
- 1.2. pašnovērtējumu par praksi (3 (trīs) darba dienu laikā pēc pēdējās prakses dienas),
- 1.3. prakses vadītāja darba vietā vērtējumu par praksi (3 (trīs) darba dienu laikā pēc pēdējās prakses dienas);
- 1.4. prakses prezentāciju (1 (vienu) darba dienu pirms prakses aizstāvēšanas);

Ja netiek izpildīts kāds no augstāk minētajiem punktiem, prakse netiek vērtēta.

Sasniedzamais rezultāts	Temats	Ieteicamais saturs	Mācību sasniegumu apguves līmeņu apraksti		Stundu skaits		
			Vidējs apguves līmenis	Optimāls apguves līmenis	Teorija	Prakse	Kopā
1. Spēj pirms darba uzdevumu izpildes uzsākšanas izvērtēt un novērst darba vides riska faktorus, iekārtojot darba vietu atbilstoši ergonomikas prasībām.	1.1. Ergonomiskas darba vietas iekārtošana.	1.1.1. Darba vietas iekārtošana, darba vides riski.	Iekārto darba vietu atbilstoši ergonomikas prasībām.	Iekārto darba vietu atbilstoši ergonomikas prasībām, izsaka pamatotos priekšlikumus darba vietas un vides uzlabošanai.	-	30	30
2. Spēj izveidot programmatūras vienību, lietojot vienu vai vairākas uzdevumam atbilstošas programmēšanas valodas, ievērojot programmatūras izstrādes metodoloģiju un labo praksi.	2.1. Programmas izstrāde.	2.1.1. Iepazīšanās ar labās prakses piemēriem programmēšanā. Programmas izstrādes gaitas un laika plānošana.	Izvērtē un saskata atšķirības savas un labās prakses stilu programmas izstrādē. Izplānota programmas izstrādes gaita un laiki.	Izvērtē un lieto labās prakses stilu programmas izstrādē. Precīzi izplānota programmas izstrādes gaita un laiki, spēj argumentēt un pamatot to.	-	210	210
		2.1.2. Programmas izstrāde atbilstoši uzdevuma formulējumiem.	Izstrādā programmu, izpildot formulētās prasības.	Patstāvīgi izstrādā un noformē programmu, izpildot formulētās prasības.			
		2.1.3. Programmas izstrādes metodoloģija.	Izstrādā programmu pēc norādītās metodoloģijas.	Patstāvīgi izvēlas efektīvāko programmas izstrādes metodoloģiju.			
		2.1.4. Programmas strukturēšana.	Strukturē programmas atbilstoši strukturēšanas nosacījumiem.	Strukturē programmas atbilstoši strukturēšanas un komentāru pievienošanas nosacījumiem.			
3. Spēj lasīt un analizēt esošu programmatūras vienību pirmkodu, uzrakstīt programmatūras vienību, pārbaudīt tās atbilstību lietotāja vajadzībām, veicot testēšanu.	3.1. Programmas analīze.	3.1.1. Pirmkoda izstrāde un testēšana.	Izstrādā un testē programmas pirmkodu.	Patstāvīgi izstrādā un testē programmas pirmkodu.	-	108	108
		3.1.2. Programmas atbilstības izvērtēšana.	Izvērtē programmas atbilstību un fiksē neatbilstības prasībām.	Izvērtē un dokumentē programmas atbilstību prasībām un novērš neatbilstības.			
		3.1.3. Programmu lasīšana un analizēšana.	Lasa un analizē programmas darbības principu.	Patstāvīgi lasa un analizē programmas darbības principu.			

4. Spēj izvērtēt programmatūras vienības pirmkoda struktūras atbilstību tā risināmajam uzdevumam un veikt koda uzlabošanu, neietekmējot programmatūras funkcionalitāti (refaktorēšana).	4.1. Koda atbilstības izvērtēšana, uzlabošana (refaktorēšana).	4.1.1. Koda atbilstības labajai praksei izvērtēšana un analizēšana.	Analizē koda atbilstību labajai praksei.	Patstāvīgi analizē un izvērtē koda atbilstību.	-	108	108
		4.1.2. Koda uzlabošana.	Pēc programmu diagnosticēšanas uzlabo programmas kodu, nemainot tās funkcionalitāti.	Patstāvīgi uzlabo programmu un efektīvi novērš nepilnības.			
5. Spēj analizēt, veidot un uzlabot datu plūsmu un glabāšanu, izmantojot atbilstošus datu pārvaldības risinājumus un datu bāzes.	5.1. Datu pārvaldība un datu bāzes izstrāde.	5.1.1. Datu bāzes izstrāde.	Izstrādā datu bāzi.	Izstrādā un analizē datu bāzi. Izskaidro datu bāzes darbības principu.	-	108	108
		5.1.2. Datu bāzes datu piesaiste sistēmām.	Apraksta datu bāzu piesaistes principus citām sistēmām.	Piesaista datu bāzi ārējām sistēmām.			
		5.1.3. Datu bāzu pārvaldība.	Izstrādā drošas datu bāzu pārvaldības sistēmas.	Izstrādā drošas un efektīvas datu bāzu pārvaldības sistēmas.			
6. Spēj testēt programmatūras atbilstību tehniskajai dokumentācijai, novērst savstarpējās atšķirības, kā arī novērst programmatūras lietotāja konstatētas kļūdas un to cēloņus, saskaņojot iznākumu ar tehnisko dokumentāciju.	6.1. Programmas testēšana un tehniskās dokumentācijas uzturēšana	6.1.1. Programmas testēšana atbilstoši tehniskajai dokumentācijai.	Pielieto programmas testēšanas pamatprincipus, pilnveido programmu, ja diagnosticēta neatbilstība dokumentācijai.	Patstāvīgi testē programmu, seko dokumentācijai, nepieciešamības gadījumā pilnveido pirmkodu.	-	132	132
		6.1.2. Pasūtītāja prasību testēšana un salīdzināšana ar galaproduktu.	Identificē galaprodukta neatbilstību pasūtītāja prasībām.	Identificē un novērš galaprodukta neatbilstību pasūtītāja prasībām.			
		6.1.3. Kļūmju un kļūdu novēršana un dokumentēšana.	Identificē, dokumentē un daļēji novērš programmā pieļautās kļūdas un kļūmes.	Patstāvīgi identificē, dokumentē un novērš programmā pieļautās kļūdas un kļūmes.			
7. Spēj atbilstoši dotajiem kritērijiem aizpildīt prakses dokumentāciju, izvērtējot darba rezultātus un izvirzot ieteikumus tālākai darbībai, kā arī iepazīstināt ar saviem sasniegumiem auditoriju.	7.1. Nepieciešamo prakses dokumentu aizpildīšana un prezentēšana. ¹	7.1.1. Dokumentācijas aizpildīšana.	Aizpilda prakses dokumentāciju ar pēc prakses vadītāja norādījumiem, veic tajā korekcijas.	Patstāvīgi aizpilda prakses dokumentāciju pēc dotajiem kritērijiem.	-	24	24
		7.1.2. Rezultātu prezentēšana.	Iepazīstina auditoriju ar prakses mērķiem un tajā paveikto, spējot daļēji atbildēt uz auditorijas jautājumiem.	Iepazīstina auditoriju ar prakses mērķiem un tajā paveikto, sniedzot padziļinātu ieskatu tajā un spējot pilnība atbildēt uz auditorijas jautājumiem.			
Kopā:					0	720	720

¹ 16 stundas paredzētas prakses programmas starpaizstāvēšanām.