

Kursuseprogramm

Ainekood IFI6074	PROGRAMMEERIMISE ALUSED		
Maht 4 EAP	Kontakttundide maht: 52, iseseisva töö maht 52	Õppesemester: S	Eksam
Eesmärk:	Luu eeldused programmeerimise olemuse, baasmõistete ja põhimeetodite mõistmiseks. Luu eeldused protseduurseks programmeerimiseks sobiva keele kasutamiseks ning tüüpiliste keeletarindite mõistmiseks. Aidata kaasa algoritmimisoskuse ning lihtsamate tüüpalgoritmide kasutamisoskuse kujunemisele. Toetada baasi kujunemist, mis on vajalik teiste programmeerimisega seotud ainete läbimisel.		
Aine lühikirjeldus: (sh iseseisva töö sisu kirjeldus vastavuses iseseisva töö mahule)	Arv- ja tekstandmete esitamine arvutis. Arvusüsteemid. Ülevaade programmeerimiskeeltest. Kõrgtaseme keelte süntaks ja semantika. Lihtsamad algoritmid: summeerimine, loendamine, suurim ja vähim väärtus. Tegevusskeem algoritmide esitamiseks. Muutujad. Lihtandmetüübid. Struktuursed andmetüübid. Aritmeetika- ja loogikaavaldised. Omistamine. Sisend. Väljund. Juhtstruktuurid: jada, valik ja kordus ning vastavad keeletarindid. Alamprogrammid ja parameetrite edastamine. Failid. Programmi koostamise tüüpilised sammud, testimine ja silumine. Aine toimub praktikatundidena arvutiklassis. Kontakttundide ja iseseisva töö tundide maht on võrdne. Iseseisva tööna on vaja lahendada kaks ülesannet, esitada programmid ja vastavad algoritmid tegevusskeemidena. Täpsed ülesanded saavad üliõpilased õppetöö jooksul. Kuna need tööd lähevad arvesse hinde kujunemisel, tuleb tööd ka kaitsta. Lisaks tuleb igal nädalal iseseisvalt lõpetada praktikatunnis pooleli jäänud ülesanded ning mõistete kinnistamiseks lugeda kursuse materjali.		
Õpiväljundid:	Kursuse edukalt läbinud üliõpilane Tunneb protseduurse programmeerimisega seonduvaid põhimõtteid, mõisteid ja keelt ning programmeerimise olemust. Analüüsib lihtsamaid probleeme ning koostab nende lahendamiseks algoritme, kasutades sealjuures tüüpalgoritme. Esitab algoritme tegevusskeemi abil ning „tõlgib“ algoritmi programmeerimiskeelde, jälgides kodeerimise reegleid ja häid tavasid. Oskab programmikoodi töötlemiseks kasutada sobivat keskkonda, programmikoodi siluda ja testida.		

Hindamismeetodid:	Eksam Hinne kujuneb kontrolltöö (kirjalik töö mõistete ja keelekonstruktsioonide tundmisest), koduste tööde kaitsmise (üliõpilane selgitab koostatud programmide, algoritmide ja testide olemust, näitab oma arusaamist kirja pandust) ja kirjaliku eksami (probleemi lahendamine ja programmi koostamine) põhjal. Eksamile pääsemise eeldusena peavad kaks kodutööd olema esitatud ja kaitstud ning kontrolltöö sooritatud.
Õppejõud:	Inga Petuhhov, õpetaja
Ingliskeelne nimetus:	Programming Fundamentals
Eeldusaine:	Puudub
Kohustuslik kirjandus:	Materjalid kursuse veebilehel: http://www.cs.tlu.ee/~inga/progbaas/ A. Downey. How to Think Like a Computer Scientist http://www.greenteapress.com/thinkpython/thinkpython.html (ptk 1., 2., 3., 5., 7., 8., 10., 14.)
Asenduskirjandus: (üliõpilase poolt läbi töötatava kirjanduse loetelu, mis katab ainekursuse loengulist osa)	Kuna kursusel puudub selgelt eristuv loenguline osa, ei ole võimalik eraldi välja tuua asenduskirjandust loengute jaoks. Lugeda tuleks samu materjale, mis on esitatud kohustusliku kirjandusena.
Õppetöös osalemise ja eksamile/arvestusele pääsemise nõuded	Eksamile pääsemise eeldusena peavad kaks kodutööd olema esitatud ja kaitstud. Võlgnevuse likvideerimiseks tuleb kodutööd ära teha ja kaitsta. Praktikatumides osalemine on kohustuslik. Puudumise korral tuleb tunnis tehtud ülesanded iseseisvalt ära teha enne järgmist tundi, sest see tagab uue osa mõistmise.
Iseseisva töö nõuded	Arvestuslike iseseisvate töödena lahendab üliõpilane kaks õppejõu poolt antud ülesannet: koostab ülesande lahendamiseks algoritmi ning esitab selle tegevusskeemina, koostab algoritmile vastava programmi ning esitab programmi lähtekoodi ja koostab ühele ülesandele testid olulisemate vigade testimiseks ning esitab nende kirjeldused. Praktikatundides pooleljäävad ülesanded tuleb kodus lõpuni lahendada. Kohustusliku kirjanduse läbitöötamist kontrollitakse nii

	kontrolltöö kui ka kodutööde kaitsmisel toimuva vestluse käigus.
Eksami hindamiskriteeriumid või arvestuse sooritamiseks vajalik miinimumtase	Hindamiskriteeriumid, millest hindamisel lähtutakse: 1. Protseduurse programmeerimisega seotud mõistete tundmine A – Orienteerub mõistetes vabalt, seostab neid omavahel ja konkreetse keelega ning algoritmimisel ja kodeerimisel ettetulevate vajadustega. B, C – Seostab mõisteid omavahel ning oskab neid praktikas kasutada. D, E – Oskab selgitada olulisemaid mõisteid ja/või tuua nende kasutamise kohta näiteid 2. Protseduurse keele põhikonstruktsioonide ning andmetüüpide tundmine A – Tunneb protseduurse keele lauseid ja andmetüüpe, oskab neid õigel viisil kombineerituna kasutada, oskab selgitada nende toimimist. Tunneb keeletarindeid laiemalt ja oskab üldistada erinevatele keeltele. Saab aru võõrast lihtsamast programmist. B – Tunneb protseduurse keele lauseid. Suudab materjali abil tutvuda uute keelekonstruktsioonidega, funktsioonidega ja täiendavate andmetüüpidega ning neid sihipäraselt kasutada C - Oskab lugeda võõrast programmi ning selgitada selles kasutatavaid keelevahendeid. Valib muutujatele sobivad andmetüübid ja algorimi jaoks sobilikud keelekonstruktsioonid D – Oskab muutujatele jaoks valida otstarbele vastavad andmetüübid. Kasutab sihipäraselt peamisi keelekonstruktsioone programmis. E – Ühe keele näitel tunneb protseduursetele keeltele omaseid tüüpilisi keelekonstruktsioone ning oskab selgitada nende toimimist. Tunneb peamisi andmetüüpe ja nende omadusi. Oskab neid kasutada jada, valiku ja tsüklit sisaldava algoritmi realiseerimiseks 3. Programmikoodi kirjutamine. Lihtsamate tüüpalgoritmide (summa, loendus, suurim ja vähim) kasutamine programmikoodis, nende sihipärane kasutamine probleemide lahendamisel. A – Kombineerib programmikoodi kirjutamisel edukalt

	erinevaid algoritme, kasutamata sealjuures standardfunktsioone. Programm arvestab erinevate eri- ja veaolukordadega ning välistatud on täitmisaegsete vigade tekkimine kõigis programmi töö faasdes. B - – Kombineerib programmikoodi kirjutamisel edukalt erinevaid algoritme, kasutamata sealjuures standardfunktsioone. Programm arvestab eri- ja veaolukordadega ning välistatud on olulisemate täitmisaegsete vigade tekkimine C, D - Kasutab programmikoodi kirjutamisel mitut tüüpalgoritmi. Oskab muuta koodinäiteid nii, et nad lahendaksid teisi sarnaseid probleeme. E – Oskab kasutada programmi kirjutamisel lihtsamaid tüüpalgoritme (summa, loendus, vähim/suurim) ühe kaupa, kasutamata sealjuures standardfunktsioone. 4. Probleemi analüüsimine ja lahendusalgoritmi koostamine A – Oskab koostada algoritmid keskmise keerukusega probleemide lahendamiseks, kus ei saa läheneda standartseid võtteid kasutades. Algoritmid on koostatud optimaalselt ja ei raiska asjatult mälu ning aega. Algoritmi esitamiseks kasutab tegevusskeemi. B - Oskab koostada algoritmid keskmise keerukusega probleemide lahendamiseks, kasutades sealjuures nii tüüpalgoritme, kui nende kombinatsioone. Algoritmid on koostatud optimaalselt ja ei raiska asjatult mälu ning aega. Algoritmi esitamiseks kasutab tegevusskeemi. C - Oskab koostada algoritmi lihtsama probleemi lahendamiseks, kasutades sealjuures mitut tüüpalgoritmi ja viimaseid modifitseerides. Algoritmi esitamiseks kasutab tegevusskeemi. D – Oskab koostada algoritmi lihtsama probleemi lahendamiseks, kasutades sealjuures mitut tüüpalgoritmi. Algoritmi esitamiseks kasutab tegevusskeemi. E – Oskab koostada algoritmi lihtsama probleemi lahendamiseks, mis eeldab ühe-kahe tüüpalgoritmi kasutamist. Algoritmi esitamiseks kasutab tegevusskeemi. 5. Programmikoodi korrektsus ja reeglitele vastavus A – Programmi kood on korrektselt kommenteeritud, muutujate nimed vastavad nende otstarbele ja headele tavadele ning on valitud süsteemselt. Kood on trepitud tavapärasel viisil ning on
--	---

	väga hästi jälgitav ja loetav B – Programmi kood järgib üldiselt soovitusi ja tavasid ning on loetav nii struktuurilt kui sisult C – Enamus programmi kood vastab soovitustele ja on üldiselt loetav tänu kommentaaridele D – Programmi kood vastab soovitustele, kuid on üle- või alakommenteeritud. Muutajete nimed on küll mnemoonilised, kuid valitud üsna juhuslikult E – Programmi kood vastab üldiselt soovitustele, kuid esineb puudujääke muutujate nimedes, kommentaarides jms. 6. Programmikoodi silumine A – Oskab programmi siluda nii väljatrükkide abil kui ka programmeerimissüsteemi silumisvahendeid kasutades. Oskab enamuse süntaksivigu läbinäha ja likvideerida juba enne kompileerimist. B, C – Kasutab programmi koodi silumiseks muutujate väljatrükke, oskab seda teha sihipäraselt ja leiab sel teel vigu. Süntaksi vead leiab keskkonna tagasiside abil edukalt. D, E – Saab hakkama oma programmi silumisega, kuid see ei toimu alati sihipäraselt ning sisaldab tugevat „katse-eksituse“ komponenti. Juhuslikkust esineb ka süntaksiveateadete tõlgendamisel ning nende kasutamisel. 7. Programmi testimine A – Koostab testid enamuse olukordade kontrollimiseks, samuti programmi käitumise hindamiseks veaolukordades. Oskab selgitada iga testi eesmärgi. B – Koostab testid loogikavigade leidmiseks, mis võivad avalduda harva ja vaid teatud sisendite puhul. Kasutab testide leidmiseks ekvivalentsiklasside meetodit. Oskab selgitada testide eesmärgi. Teab, et ammendav testimine pole võimalik. C – Koostab testid olulisemate vigade leidmiseks. Testid on koostatud ennekõike programmikoodi ülesehitusest lähtudes. Oskab selgitada iga testi eesmärgi. D – Koostab testid ja testib oma programmi olulisemate veaolukordade leidmiseks.
--	---

	E – Testib oma programmi juhuslike sisenditega, kuid siiski sellistega, mis tagavad programmi korrektse töö juhul, kui veaolukordi ei teki ning läbitakse tüüpiline töövoog. Ei oska täpselt selgitada, miks ta just sellised testid koostas
Informatsioon kursuse sisu kohta, kursuse jaotumine teemade kaupa sh kontakttundide ajad	Läbitavad teemad nädalate või loengute kaupa. Toimumisajad (nt vahearvestused, kontrolltööd, iseseisvate tööde esitamise ja hindamise tähtajad).
9.09.10 kell 16.15	Sissejuhatus. Üldine tutvustus. Programmeerimine. Algoritm. Algoritmi üleskirjutamine – tegevusskeem. Jada, kui algoritmi osa. Jada ja paralleelsus tegevusskeemil.
10.09.10 kell 8.15	Algoritmid probleemide lahendamiseks.
16.09.10 kell 16.15	Sissejuhatus Pythonisse. IDLE kasutamine. Süntaksivead.
17.09.10 kell 8.15	Lihtsad programmid jadaga.
23.09.10 kell 16.15	Andmete esitamine arvutis: bit, bait, sõna. Numbriliste andmete esitamine. Positsioonilised arvusüsteemid. Tekstiliste andmete digitaalne esitamine. Peamised andmetüübid. Erinevad kooditabelid..
24.09.10 kell 8.15	Mõistete kordamine.
30.09.10 kell 16.15	Valik kui algoritmi ehituskivi. Ülesanded valikutega. Valik tegevusskeemil. Võrdlustehed. Valikulause Pythonis.
1.10.10 kell 8.15	Testimisest põhjalikumalt. Testide koostamine juba lahendatud probleemidele.
7.10.10 kell 16.15	Loogikaavaldised. Plokid programmis ja programmikoodi treppimine.
8.10.10 kell 8.15	Tsükel, kui algoritmi ehituskivi. Lihtsad algoritmid summeerimiseks ja loendamiseks.
14.10.10 kell 16.15	Probleemid, mis vajavad jada ja valikut.
15.10.10 kell 8.15	Esimese kodutöö esitamise tähtaeg
21.10.10 kell 16.15	Struktuursed andmetüübid: massiivid / listid. Nende kasutamine programmides.

22.10.10 kell 8.15	Ülesanded suuremate andmehulkade töötlemisega. Tüüp algoritmide (summeerimine, loendamine, suurim, vähim) kasutamine massiivis
4.11.10 kell 16.15	Ülesanded suuremate andmehulkade töötlemisega.
5.11.10 kell 8.15	Kontrolltöö
11.11.10 kell 16.15	Alamprogrammid Pythonis.
12.11.10 kell 8.15	Alamprogramm: veidi teooriat. AP kui protseduurse programmeerimise alus. Protseuur. Funktsioon. Andmevahetus alamprogrammidega. Globaalsed ja lokaalsed muutujad ja skoop. Moodulid.
18.11.10 kell 16.15	Töö tekstifailidega: andmete lugemine ja kirjutamine.
19.11.10 kell 8.15	Mitmemõõtmeline massiiv. List listis.
25.11.10 kell 16.15	Ülesannete lahendamine.
26.11.10 kell 8.15	Sõnastik. Sõnastiku kasutamine programmis
2.12.10 kell 16.15	Teise kodutöö tähtaeg.
3.12.10 kell 8.15	Ülesannete lahendamine: kahemõõtmeline andmestik + tekstifail +funktsioonide kasutamine.
9.12.10 kell 16.15	Ülesannete lahendamine: kahemõõtmeline andmestik + tekstifail +funktsioonide kasutamine.
10.12.10 kell 8.15	Ülesannete lahendamine. Kodutööde kaitsmine.
16.12.10 kell 16.15	Ülesannete lahendamine. Kodutööde kaitsmine.
17.12.10 kell 8.15	Eksam

Õppeainet kureeriv üksus:	Informaatika instituut
Kursuseprogrammi koostaja	Inga Petuhhov

Allkiri:	
Kuupäev:	

Kursuseprogramm registreeritud akadeemilises üksuses

Kuupäev	31.08.2010
Õppeassistendi nimi	Hanna-Liisa Pender
Allkiri	