

# Rakenduste loomise ja programmeerimise alused

## Gümnaasiumi valikkursus

### 1. Õppe- ja kasvatuseesmärgid

Valikkursusega taotletakse, et õpilane:

- 1) arendab loovust, loogilist, analüütilist ja algoritmilist mõtlemist ning süsteemset käsitusviisi probleeme ja ülesandeid lahendades;
- 2) teadvustab ja tunnetab programmjuhtimisega a süsteemide tööpõhimõtet ning info esitamise ja töötlemise põhiprotsesside olemust;
- 3) tunneb rakenduste ja programmide loomise vahendeid ning põhimeetodeid;
- 4) omandab programmide ja algoritmide koostamise ning probleemide lahendamise baasoskused programmjuhtimisega süsteemide abil;
- 5) tutvub objektorienteeritud modelleerimise, analüüsi ja disaini põhimõtetega;
- 6) saab aru objektide ja andmete olemusest, nende omadustest ning nendega täidetavatest tegevustest algoritmides ja programmides;
- 7) omandab algoritmimise ja programmeerimise põhikontseptsioonid ja mõisted ning oskused kavandada, koostada, siluda ja testida programme, mis koosnevad mitmest koostoimivast üksusest (protseduurist); kasutab objekte, skalaarandmeid ja massiive ning kirjeldab eri liiki protsesse.

### 2. Kursuse lühikirjeldus

Kursuse põhiosad:

- 1) rakenduste loomise põhimõtted;
- 2) mudelid ja modelleerimise alused;
- 3) algoritmimise ja programmeerimise alused.

Osi ega nende teemasid ei läbita järjestikku, vaid sobiva käsitluse valib õpetaja, arvestades kasutatavat programmeerimisvahendit, kursuse suunitlust, oma metoodilisi kogemusi ja eelistusi jmt. Rakenduste loomise põhimõtteid ja põhifaase tutvustatakse praktilise töö kaudu, koostades mõned projektid, mis sisaldavad modelleerimise, analüüsi ning disaini elemente. Modelleerimises on peamine koostada ja esitada algoritme unifikseeritud modelleerimiskeele UML abil realiseeritavate programmide jaoks. Klassimudelite olemust võib tutvustada

võrdlemisi lühidalt ja üldiselt, arvestades, et õppija oskaks lugeda nende abil esitatud süsteemide ja andmete struktuure ning saada neist aru. Põhiosa ajast kulub programmeerimise ning algoritmimise aluste omandamisele praktilise töö kaudu.

Kursuse eesmärkide saavutamiseks on tähtis teadvustada ning tunnetada programmide ja programmjuhtimise olemust, koostades praktilises töös programme ning realiseerides neid arvutil.

### **3. Õppetegevus**

1. Õppetegevus toimub arvutiklassis praktikumide ja seminaride vormis.
2. Õppetööd peab toetama elektroonsete õppematerjalide komplekt: teooria põhiosad, harjutused, näited, lingid, elektroonsed töövihikud jm.
3. Õpilased teevad iseseisvalt (kas klassis või kodus) 3–4 tööd, mille alusel määratakse hinne.

### **4. Füüsiline õpikeskkond**

Programmeerimise õpikeskkond. Valiku teeb kool ise.

### **5. Hindamine**

Valikaine „Rakenduste loomise ja programmeerimise alused“ õpitulemuste hindamine lähtub gümnaasiumi riikliku õppekava üldosas ja teistes hindamist reguleerivates dokumentides toodud hindamisalustest.

Õpitulemuste kontrolli ja hindamise eesmärk on saada ülevaade õpitulemuste saavutatusest ja õpilase individuaalsest arengust ning kasutada saadud teavet õppe tulemuslikumaks kavandamiseks. Hinnatakse nii teadmisi ja nende rakendamise oskust kui ka üldpädevuste saavutatust, sh õpioskusi kirjalike ja praktiliste tööde ning praktiliste tegevuste alusel, arvestades õpilase teadmiste ja oskuste vastavust ainekava taotletavatele õpitulemustele.

Õpitulemusi hinnatakse sõnaliste hinnangute ja arvestatud/mitte arvestatud hinnandutega. Õpilane peab teadma, mida ja millal hinnatakse, milliseid hindamisvahendeid kasutatakse ja millised on hindamise kriteeriumid. Kursus lõppeb hindelise arvestusega (praktiline töö).

Kursuse hinne kujuneb iseseisvate- ja praktiliste tööde hinnetest.

Iseseisvateks töödeks on erinevad praktilised harjutused, mille käigus peab õpilane rakendama teoreetilisi teadmisi. Kontrollitakse ja hinnatakse õpilase teoreetilisi teadmisi, informatsiooni leidmist ja selle rakendamiskust, loovust ja vormistuse korrektsust ning töö iseseisvat sooritust.

Hinnatakse:

- 1) õppekavas ettenähtud õpitulemuste saavutamist ning seonduvate pädevuste olemasolu veenvat rakendamist;
- 2) arvutige loodud rakenduste tehnilist teostust, esteetilisust ja originaalsust;
- 3) õpilasepoolset praktilise tegevuse mõtestamist;
- 4) õpilase arengut.

## 6. Õppesisu ja õpitulemused

Õpitulemused	Tund	Teoreetiline ja praktiline tegevus tunnis	Hindamine
<b>Sissejuhatus.</b> Õpilane pääseb ligi e-õppekeskkonnale ja on tutvunud kursuse ülesehituse ja nõuetega.	1	Tunnis tutvustatakse õpilasele kursuse sisu ja ülesehitust (esitlus „Rakenduste .....“). Millised on nõuded kursuse läbimiseks. Kursuse e-õppekeskkonnaga tutvumine ja kasutajaks registreerumine.	
<b>Rakenduste loomise põhimõtted.</b> Õpilane teab ning oskab võrrelda ja hinnata rakenduste loomise erinevaid vahendeid ning meetodeid. Õpilane tunneb ja teab erinevaid programmeerimiskeeli ja programmeerimissüsteem e. Õpilane eristab ja oskab kirjeldada rakenduste loomise põhifaase: ülesande püstitus, analüüs, disain, realisatsioon.	2	Teoreetiline: õpetaja tutvustab esitluse abil erinevaid programmeerimiskeeli ja –süsteeme samuti tutvustab ta rakenduste loomise põhifaase (püstitus, analüüs, disain ja realisatsioon). Näidete abil tutvustatakse rakenduste loomise faase ja objekte.	
<b>Sissejuhatus Pythonisse.</b> Õpilane teab kuidas kasutada Pythoni kasutajaliidest ja selle võimalusi. Õpilane oskab koostada Pythonis arv-, tekst- ja loogikaavaldisi ja omistamislauseid. Õpilane	3	Tunni alguses õpetaja tutvustab Pythoni kasutajaliidest (IDLE) ja selle võimalusi ning tutvustab Pythoni ajalugu. Koos proovitakse lihtsamate avaldiste kirjutamist ja testimist. Tunni lõpus proovitakse skripte kirjutada ja käivitada.	Praktilised harjutused IDLE kasutajaliidises.

oskab koostada lihtsa Pythoni skripti, seda käivitada ja salvestada faili			
<b>Pythoni laused. Moodulid.</b> Õpilane teab Pythoni lihtlauseid ja oskab avaldistes kasutada funktsioone. Õpilane tunneb Pythoni liitlausete süntaksit ja teab mõningaid valiku- ja korduslauseid .	3	Õpetaja tutvustab töölehte. Edasi vaadatakse omistamis- ja print-lauseid töölehe näidetes. Õpetaja tutvustab Pythoni mooduleid. Kordus- ja valik-lauseid tutvustamine ja skriptides kasutamine (taanded).	Praktilised harjutused. Erinevate näidete läbitöötamine.
<b>Python. Lihtsa skripti koostamine.</b> Õpilane teab Pythoni lihtlauseid ja oskab avaldistes kasutada funktsioone. Õpilane oskab koostada Pythonis lihtsa skripti. Õpilane oskab kasutada valikulaused. Õpilane oskab avastada ja parandada vigu skriptis.	2	Õpilane täidab töölehel olevad korraldused ja proovib koostada töötava skripti. Õpilane peab kontrollima, et skript töötaks. Pärast valmis tööde esitamist annab õpetaja tagasisidet, et millised olid lahenduste põhilised vead ja kuidas neid vältida. Õpetaja juhendab õpilasi oma programme siluma ja korralikult tööle panema. Õpetaja selgitab uuesti tähtsamaid kitsaskohti ja kuidas neid avastada (print korraldus).	Hindeline arvestus
<b>Python. Funktsioonid.</b> Õpilane teab Pythoni lihtlauseid ja oskab avaldistes kasutada funktsioone. Õpilane oskab koostada Pythonis lihtsa skripti ja seda kasutada. Õpilane oskab avastada ja parandada vigu oma skriptis.	2	Õpetaja tutvustab töölehte Funktsioonid. Õpetaja seletab lahti kuidas koostada ja kasutada funktsioone Pythonis. Koos proovitakse mõningaid funktsioone. Edasi proovivad õpilased iseseisvalt koostada mõningaid funktsioone.	Praktilised harjutused liitlausetega.
<b>Python. Kordused.</b> Õpilane teab ja oskab kasutada Pythoni korduslauseid	2	Õpetaja tutvustab for-kordust ja while-kordust. Koos vaadatakse ja modifitseeritakse näiteid.	Praktilised harjutused kordustega.
<b>Python. Rehmaatika.</b> Õpilane oskab Pythonis kasutada valikulaused ja korduslauseid. Õpilane	2	Õpetaja tutvustab töölehte. Koos arutatakse, et millised tegevused peaksid selles Skriptis olema. Õpetaja juhendamisel koostatakse skript etteantud algoritmi alusel.	Praktilised harjutused valik- ja korduslausetega.

oskab koostada skripti etteantud algoritmi alusel. Õpilane oskab avastada ja parandada vigu skriptis.			
<b>Python. Arvestustöö.</b> Õpilane teab Pythoni lihtlauseid ja oskab avaldistes kasutada funktsioone. Õpilane oskab Pythonis kasutada valikulaused ja korduslauseid. Õpilane oskab koostada skripti etteantud algoritmi alusel. Õpilane oskab avastada ja parandada vigu skriptis.	3		Hindeline arvestus.
<b>Python. Tekstiavaldised.</b> Õpilane teab võimalusi tekstist sümboli(te) eraldamiseks, oskab koostada tekstiavaldisi. Õpilane oskab koostada skripti etteantud algoritmi alusel. Õpilane oskab avastada ja parandada vigu skriptis.	2	Õpetaja tutvustab töölehte Tekstiavaldised. Õpetaja tutvustab erinevaid tekstiavaldisi (len jms). Koos koostatakse erinevaid funktsioone etteantud algoritmise alusel (sõnade tagurpidi kirjutamine jms).	Praktilised harjutused tekstiavaldistega.
<b>Python. Loendid.</b> Õpilane oskab luua loendi Pythonis. Õpilane oskab lugeda väärtusi loendist. Õpilane oskab lisada ja kustutada loendi elemente. Õpilane oskab avastada ja parandada vigu skriptis.	2	Õpetaja tutvustab õpilastele, mis on loend ja kuidas seda Pythonis luua ja kasutada. Koos proovitakse loendeid luua, sinna väärtusi lisada ja kustutada. Erinevate skriptide uurimine ja testimine ning vigade avastamine ja parandamine.	Praktilised näited loenditest.
<b>Python. Sõnastik ja Edetabel.</b> Õpilane oskab lugeda väärtusi loendist, lisada ja kustutada loendi elemente. Õpilane teab, kuidas lugeda loendisse admeid tekstifailist ja kirjutada tulemusi tekstifaili. Õpilane oskab avastada ja parandada vigu skriptis.	2	Õpetaja näitab, et loendid võivad paikneda ka kuskil failis. Mõningate näidete abil proovib õpetaja selgitada kuidas on võimalik kasutada loendeid mis asuvad failis ja kuidas neid tekstifailidesse kirjutada. Koos koostatakse erinevaid funktsioone ja proovitakse ning parandatakse neid.	Praktilised harjutused andmete lugemiseks ja kirjutamiseks failidest.

<b>Python. Moodul turtle.</b> Õpilane teab ja oskab kasutada moodulit turtle. Õpilane tunneb turtle lihtsamaid korraldusi. Osab joonistada lihtsamaid kujundeid.	2	Õpetaja tutvutab õpilastele Pythoni graafilist moodulit turtle. Õpetaja näitab kuidas seda kasutada ja millised on põhilised korraldused mida turtle tunneb. Näite harjutusena üritavad õpilased joonistada lahtsit kirjaümbrikku ja Lumememme. Kilpkonna juhtimiseks on kuus lihtsat käsku: forward(n), backward(n) – edasi või tagasi n sammu, left(d), right(d) – vasakule või paremale d kraadi, up(), down() - pliiatsi üles tõstmine ja langetamine.	Praktilised harjutused turtle mooduliga.
<b>Python. Riskülik ja ring.</b> Õpilane teab ja oskab kasutada mooduli turtle.py mõningaid funktsioone. Õpilane oskab avastada ja parandada vigu skriptis.	2	Õpetaja tutvustab töölehte ja koos tutvutakse turtle funktsioonidega töölehel. Õpilased iseseisvalt koostavad skripte.	Praktilised harjutused.
<b>Python. Lõputöö.</b> Õpilane teab ja tunneb Pythoni mooduleid ja korraldusi ning oskab neid kasutada rakenduse loomiseks.	3	Lill kolmnurkadest või ruutudest. Õpilased peavad vastavalt juhisele lahendama ülesande, kasutades kõiki eelpool õpitud Pythoni võimalusi.	Lõputöö.
<b>Varu</b>	2		

## 7. Õppeprotsessi kirjeldus

### 7.1. Sissejuhatus ainesse

*Maht:* 1 tund

*Teoreetiline osa:* Tunnis tutvustatakse õpilasele kursuse sisu ja ülesehitust (esitlus „Rakenduste .....“). Millised on nõuded kursuse läbimiseks. Kursuse e-õppekeskkonnaga tutvumine ja kasutajaks registreerumine.

*Praktiline osa:* arutelu, esitlus.

*Hindamisvõimalus:* ei hinnata.

*Õpisisu:* Õpilasel on ligipääs e-õppekeskkonnale ja ta on tutvunud kursuse ülesehituse ja nõuetega.

### 7.2. Rakenduste loomise põhimõtted

*Maht:* 2 tund

Teoreetiline osa: õpetaja tutvustab esitluse abil erinevaid programmeerimiskeeli ja –süsteeme samuti tutvustab ta rakenduste loomise põhifaase (püstitus, analüüs, disain ja realisatsioon). Näide Pong ja Kits\_hunt\_kapsas tutvustamine. Näidete abil tutvustatakse rakenduste loomise faase ja objekte. Programmi Scratch lühike tutvustus, ülevaade.

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Rakenduste loomise erinevaid vahendeid ning meetodid. Erinevaid programmeerimiskeeled ja programmeerimissüsteemid. Rakenduste loomise põhifaasid: ülesande püstitus, analüüs, disain, realisatsioon.

### **7.3. Sissejuhatus Pythonisse**

Maht: 3 tund

Teoreetiline osa: Õpetaja tutvustab Pythoni kasutajaliidest (IDLE) ja selle võimalusi ning tutvustab Pythoni ajalugu. Koos proovitakse lihtsamate avaldiste kirjutamist ja testimist. Tunni lõpus proovitakse skripte kirjutada ja käivitada failist.

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Pythoni kasutajaliides ja selle võimalused ning kasutamine. Pythonis arv-, tekst- ja loogikaavaldiste ja omistamislausete koostamise põhimõtted. Lihtsate Pythoni skriptide koostamine, käivitamine ja salvestamine faili.

### **7.4. Pythoni laused. Moodulid**

Maht: 3 tund

Teoreetiline osa: Vaadatakse omistamis- ja print-lauseid töölehe näidetes. Õpetaja tutvustab Pythoni mooduleid. Kordus- ja valik-lausete tutvustamine ja skriptides kasutamine (taanded).

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Pythoni lihtlauseid ning avaldistes funktsioonide kasutamine. Pythoni liitlausete süntaks ja valiku- ja korduslausete näited.

### **7.5. Lihtsa skripti koostamine**

Maht: 2 tund

Teoreetiline osa: Hindeline arvestus. Õpilane täidab töölehel olevad korraldused ja proovib koostada töötava skripti. Õpilane peab kontrollima, et skript töötaks. Pärast valmis tööde

esitamist annab õpetaja tagasisidet, et millised olid lahenduste põhilised vead ja kuidas neid vältida. Õpetaja juhendab õpilasi oma programme siluma ja korralikult tööle panema. Õpetaja selgitab uuesti tähtsamaid kitsaskohti ja kuidas neid avastada (print korraldus).

Praktiline osa: Hindeline arvestus

Õpisisu: Lihtlauseid ja funktsioone kasutamine avaldistes. Valikulausete kasutamine.

## **7.6. Funktsioonid**

Maht: 2 tund

Teoreetiline osa: Õpetaja tutvustab töölehte Funktsioonid. Õpetaja seletab lahti kuidas koostada ja kasutada funktsioone Pythonis. Koos proovitakse mõningaid funktsioone. Edasi proovivad õpilased iseseisvalt koostada mõningaid funktsioone.

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Liitlause kasutamine koos funktsioonidega. Vigade parandamine skriptis.

## **7.7. Kordused**

Maht: 2 tund

Teoreetiline osa: Õpetaja tutvustab for-kordust ja while-kordust. Koos vaadatakse ja modifitseeritakse näiteid.

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Korduslause kasutamine skriptides programme modifitseerimiseks.

## **7.8. Rehmaatika**

Maht: 2 tund

Teoreetiline osa: Õpetaja tutvustab töölehte. Koos arutatakse, et millised tegevused peaksid selles Skriptis olema. Õpetaja juhendamisel koostatakse skript etteantud algoritmi alusel

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Pythonis valikulausete ja korduslause kasutamine. Skripti koostamine etteantud algoritmi alusel. Vigade avastamine ja parandamine skriptis.

## **7.9. Tekstiavaldised**

Maht: 3 tund



Teoreetiline osa: Õpetaja tutvustab töölehte Tekstiavaldised. Õpetaja tutvustab erinevaid tekstiavaldisi (len jms). Koos koostatakse erinevaid funktsioone etteantud algoritmise alusel (sõnade tagurpidi kirjutamine jms).

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Tekstist sümboli(te) eraldamine ja erinevate tekstiavaldiste koostamine. Etteantud algoritmi alusel skripti koostamine. Vigade avastamine ja parandamine skriptis.

## **7.10. Arvestustöö**

Maht: 2 tund

Praktiline osa: Hindeline arvestus

Õpisisu: Pythoni lihtlaused ja funktsioonide kasutamine. Valikulaused ja korduslaused. Skripti koostamine etteantud algoritmi alusel.

## **7.11. Loendid**

Maht: 2 tund

Teoreetiline osa: Õpetaja tutvustab õpilastele, mis on loend ja kuidas seda Pythonis luua ja kasutada. Koos proovitakse loendeid luua, sinna väärtusi lisada ja kustutada. Erinevate skriptide uurimine ja testimine ning vigade avastamine ja parandamine.

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Loendite loomine Pythonis. Väärtuste lugemine loendist. Elementide lisamine ja kustutamine loendist.

## **7.12. Sõnastik ja edetabel**

Maht: 2 tund

Teoreetiline osa: Õpetaja näitab, et loendid võivad paikneda ka kuskil failis. Mõningate näidete abil proovib õpetaja selgitada kuidas on võimalik kasutada loendeid mis asuvad failis ja kuidas neid tekstifailidesse kirjutada. Koos koostatakse erinevaid funktsioone ja proovitakse ning parandatakse neid.

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Väärtuste lugemine loendist ning elementide lisamine ja kustutamine loendist. Elementide lugemine loendisse tekstifailist ja kirjutamine tekstifaili.

## **7.13. Turtle moodul**

Maht: 2 tund

Teoreetiline osa: Õpetaja tutvutab õpilastele Pythoni graafilist moodulit turtle. Õpetaja näitab kuidas seda kasutada ja millised on põhilised korraldused mida turtle tunneb. Näite harjutusena üritavad õpilased joonistada lahtsit kirjaümbrikku ja Lumememme. Kilpkonna juhtimiseks on kuus lihtsat käsku:forward(n), backward(n) – edasi või tagasi n sammu, left(d), right(d) – vasakule või paremale d kraadi, up(), down() - pliiatsi üles tõstmine ja langetamine.

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Turtle mooduli kasutamine. Turtle lihtsamate korralduste kasutamine. Lihtsamate kujundite joonistamine.

#### **7.14. Ristkülik ja ring**

Maht: 2 tund

Teoreetiline osa: Õpetaja tutvustab töölehte ja koos tutvutakse turtle funktsioonidega töölehel. Õpilased iseseisvalt koostavad skripte.

Praktiline osa: arutelu, esitlus.

Hindamisvõimalus: ei hinnata.

Õpisisu: Mooduli turtle mõningate funktsioonide kasutamine.

#### **7.15. Lõputöö**

Maht: 3 tund

Teoreetiline osa: Õpilased peavad vastavalt juhisele lahendama ülesande, kasutades kõiki eelpool õpitud Pythoni võimalusi.

Praktiline osa: Arvestustöö

Õpisisu: Pythoni moodulite ja korralduste kasutamine rakenduse loomiseks.