

Loengud

| Loengu nr. | Loengu teemad |
|------------------------|---|
| Loeng 1 (Slaidid 1) | <i>Kursuse läbimisest praktiliselt</i> , Python, sissejuhatus programmeerimisse Pythonis, Pythoni installeerimine, konsool, Jupyter, minu esimene programm, lihtne aritmeetika, sõne (string), funktsioon <code>print</code> . |
| Loeng 2 | Omistus operaator, muutuja, sõne indekseerimine, sõne meetodid, sõne vorming, f-string, r-string, list, korteez, hulk, kujutus, lausend <code>del</code> . |
| Loeng 3 | Ülevaade: andmetüübid ja operatorid, numbrisüsteemid, tehete ja avaldiste prioriteetsus, lausend <code>def</code> , <code>if</code> laused, voodiagramm, algoritm. |
| Loeng 4 | <code>if</code> valikuavaldis, rekursiivne funktsioon, kõrgemat järku funktsioon, anonüümne funktsioon ja lausend <code>lambda</code> , <code>and</code> ja <code>or</code> operaatorite rakendamine mitte-booli andmetüüpidele. |
| Loeng 5 | Määramata ja määratud itereerimine, <code>while</code> ja <code>for</code> tsükliid/blokid, lausendid: <code>continue</code> , <code>break</code> , funktsioonid <code>input</code> ja <code>enumerate</code> . |
| Loeng 6 | Lahtipakkimine ja operaatorid <code>=</code> , <code>*</code> , <code>**</code> , muutujate skoop, LEGB reegel, lausendid <code>global</code> ja <code>nonlocal</code> , <code>for-in</code> avaldis, <code>for-in-if</code> avaldis. |
| Loeng 7 | Generaatorid, iteraatorid ja laisk itereerimine, lausendid <code>yield</code> ja <code>yield from</code> , generaatori avaldis. |
| Loeng 8 | Sisseehitatud funktsioonide ülevaade, funktsiooni sulund, dekoraator ja <code>@</code> operaator, lausend <code>assert</code> , programmi käivitamine käsurealt/konsoolist, IDE tutvustus ja kasutamine, <i>kodutöö juhend ja näidis</i> . |
| Loeng 9 | Modulaarsus, lausend <code>import</code> , moodul, pakett, peaprotseduur, <code>__name__=='__main__'</code> , Pythoni standardteegid, kolmandate osapoolte teegid ja moodulid, Python Package Index (PyPI), Pythoni distributsiooni haldamine. |
| Loeng 10 | Erisused, erisuste haldamine, erisuste hierarhia, konstruktsioon: <code>try/except/else/finally</code> , lausend <code>raise</code> , silumine, süntaksiviga, täitmisaegne viga, loogikaviga, versioonikontroll. |
| Loeng 11 | Fail, binaarne fail, tekstifail, failide haldus Pythonis, faili objekti meetodid, funktsioon <code>open</code> , kontekstihaldur <code>with/as</code> , serialiseerimine, moodulid <code>pickle</code> ja <code>shelve</code> , moodul <code>csv</code> . |
| Loeng 12 | Objektorienteeritud programmeerimine 1 (OOP): klass, konstruktor ja spetsiaalne meetod <code>__init__</code> , meetodi defineerimine, atribuut, spetsiaalne meetod, plümorfism, pärilikkus, lausendid: <code>class</code> , <code>self</code> , funktsioonid: <code>isinstance</code> , <code>issubclass</code> . |
| Loeng 13 | Objektorienteeritud programmeerimine 2: meetodi dekoraator, dünaamiline atribuut, klassimeetod, staatiline meetod, pärilikkus, pesastatud klass, lausend: <code>cls</code> , funktsioon: <code>super</code> . |
| Loeng 14 | Teadusarvutused Pythonis, algoritm (meeldetuletus), teegid: NumPy, SciPy, matplotlib, jm., andmemassiiv <code>numpy.ndarray</code> , jooniste ja graafikute loomine, joonistega töötamine Spyder IDEs ja Jupyteris, näiteid. |
| Loeng 15 | Matplotlib ja mitme 2D teljepaari loomine joonise kohta, jooniste vormistamine, NumPy ja SciPy teekide kasutamise näiteid. |
| Slaidid 16 | <i>Külalisloeng</i> , Kokkuvõte, mida ja kuidas edasi õppida?, <i>ettevalmistus eksamiks</i> . |

Kursuse ülesehitus

- Sissejuhatus
 - Arvuti
 - Programmeerimine ja programmeerimiskeeled
 - Pythoni programmeerimiskeel
- Pythoni alused
 - Tööriistad, interpretaator, IDE: IPython, Jupyter, Spyder jt.
 - Sõne, sõne vorming
 - Pythoni andmetüübid, tehted ja avaldised
 - Funktsioon, anonüümne funktsioon, rekursioon, sisseehitatud funktsioonid, generaator, klassi meetodid, protseduur, peaprotseduur
 - Määratud ja määramata tsüklid
 - Silumine ja erisuste haldus
 - Failihaldus ja serialiseerimine, modulaarsus ja teekide import, distributsiooni haldamine
 - Objektorienteeritud programmeerimine (OOP)
- Teadusarvutused ja muud rakendused
 - Algoritm
 - NumPy — Andmete esitus massiividena
 - SciPy — Teadusarvutuste algoritmid
 - Matplotlib — Andmete analüüs ja visualiseerimine
 - SymPy — Sümbolarvutus, computer algebra system (CAS)
 - jm.

Kursuse ainekava/tutvustus

Programmeerimise olemus. Programmeerimiskeele Python ja selle töövahendite ülevaade. Pythoni süntaks ja programmi struktuur. Standardsed andmetüübid (`int`, `float`, `complex`, `str`, `bool`, `list`, `tuple`, `dict`, `set`, `object`, `type`), objektide defineerimine ja kasutamine. Muutujate defineerimine ja kasutamine. Avaldised (tehted erinevat andmetüüpi objektidega: `+`, `-`, `*`, `/`, `//`, `**`, `%`, `not`, `and`, `or`, `!`, `&`, `^`, `[]`, `()`, `in`, `is`, `<`, `>`, `==`, `!=`, tehete prioriteetsus) ja lausendid (`del`, `for`, `while`, `if/elif/else`, `try/except/with`, `break/continue/pass`). Standardsed sisseehitatud funktsioonid (`print`, `range`, tüüpi teisendused, jne), funktsioonide defineerimine (`def` ja `return` lausendid, `lambda` avaldis) ja kasutamine, generaatorid (`yield` lausend), dekoraatorid (`@` operaator). Objektorienteeritud programmeerimise alused (`class` lausend). Standardsed moodulid, moodulite loomine ja kasutamine (`import` ja `from` lausendid), moodulite installeerimine (Anaconda, `conda`, `pip`). Andmefailide loomine ja kasutamine. Teadusarvutuse paketid (`numpy`, `scipy`, `matplotlib`) ja töökeskkonnad (PyCharm, Spyder, Jupyter, IPython).