

IAX0586 Programmeerimine III

Kursuse eesmärgid

Kursuse eesmärkideks on:

- Anda kuulajatele baasteadmised programmeerimiskeelest C++.
- Anda ülevaade programmeerimises kasutatavatest andmetötluse struktuuridest ja algoritmidest, sealhulgas ka C++ tugitarkvarasse kuuluvatest konteineritest ja andmetötluse meetoditest.
- Treenida üliõpilasi et nad omandaksid teatud praktilise C++-s programmeerimise oskuse.

Eeldused

Osalejatelt eeldatakse, et nad on läbinud kursused IAX0583 (Programmeerimine I) ja IAX0584 (Programmeerimine II) või siis on õppinud programmeerimist C keeles mingil muul moel. C keelt absoluutselt mitteoskavate üliõpilaste jaoks on see kursus liiga raske.

Varustus

Kursusel kasutatav operatsioonisüsteem on Windows ning programmeerimiskeel C++. Kuna osa tööd tuleb nagunii teha väljaspool auditooriumi, peab igal üliõpilasel olema arvuti (soovitavalt 64-bitise protsessoriga), millel jookseb:

- Windows (versioonid 10 või 11). Kursusetöös kasutatakse tarkvara, mis Linux ja Mac ei pruugi töötada.
- Microsoft Visual Studio arenduskeskkond C/C++ toega (2022 aasta versioon, suure tõenäosusega sobib ka 2019 aasta versioon)). Vabavarast *Visual Studio Community* piisab täielikult (<https://visualstudio.microsoft.com/downloads/>).

Kursusetööd

Kursusetöid on kaks. Neist täieliku ülevaate saamiseks tuleb alla laadida vastavad spetsifikatsioonid. Allpool esitatakse lühike kokkuvõte.

Esimese kursusetöö jaoks on õppejõu poolt on ette valmistatud DLL, mis genereerib lähteandmeid ja paigutab need teatud keerulisse, viitadega ühendatud andmestruktuuri. Ülesandeks on koostada C++ klass, mille tähtsamaiks atribuudiks on viit genereeritud struktuurile. Klassis tuleb implementeerida meetodid, millede abil saab andmestruktuuri lisada uusi liikmeid, eemaldada olemasolevaid liikmeid, kirjutada terve struktuur kettale ja lugeda sealt tagasi mälli, väljastada kõik andmestruktuuri liikmed ekraanile. Samuti peab klass sisaldama konstruktoreid (k.a. kopeeriv konstruktor), destruktorit ja operaatorite ülelaadimise (operator overloading) meetodeid. Koostatud tarkvara õigsust kontrollitakse õppejõu poolt esitatud testidega.

Teine kursusetöö erineb esimesest selle poolest, et andmestruktuur koosneb mitmest erinevatest STL konteineritest (*vector*, *list*, *map*). Meetodite implementeerimisel tuleb kasutada STL algoritme.

Hindamine

Eksamit klassikalises mõttes ei ole. Hinne antakse kursusetööde põhjal.

Kummagi kursusetöö eest antakse punkte järgmiste reeglite alusel:

- Loodud tarkvara vastab täielikult spetsifikatsioonile ja läbib kõik testid – 30 punkti.
- Tarkvara läbib kõik testid kuid esineb väiksemaid kõrvalekaldumisi spetsifikatsioonist – 25 punkti.
- Mõned testid ei tööta – 20 punkti.
- Enamik testidest ei tööta - 15 punkti.
- Kood on lõpetamata, kompileerimine ja linkimine ei ole võimalik – 10 punkti.
- Tegemist on osalise või täieliku plagiaadiga – 0 punkti.
- Mitte midagi ei ole esitatatud – 0 punkti.

Kursusetööde esitamise tähtajaks on vastavalt 13 ja 16 õppenädal. Hilinemise eest võetakse 5 punkti maha.

Lõpphinne arvutatakse saadud punktide kogusummast järgmise tabeli alusel:

- "0" – alla 30 punkti.
- "1" – 35 punkti.
- "2" – 40 punkti.
- "3" – 45 punkti.
- "4" – 50 punkti.
- "5" – enam kui 50 punkti.

Aeg ja koht

Kursust peetakse ainult sügissemestril (erandkorras ka 2025 kevadsemestril). Tunnid (kord nädalas, 4 * 45 minutit) hõlmavad nii loenguid kui ka harjutusi.

Keel

Õppetöö toimub eesti keeles. Õppematerjalid on inglise keeles.

Loengute teemad

Kuivõrd eestikeelsed terminid tihtipeale puuduvad või ei ole üldtunnustatud, siis arusaamatuste vältimiseks on kursuses käsitletavate teemade loetelu kirjutatud inglise keeles.

1. Introduction to C++:
 - a. Memory allocation operators.
 - b. Unicode and type *wchar_t*. Operations with Unicode strings.
 - c. Scope resolution operator.
 - d. Default values for function arguments.
 - e. Overloading.
 - f. Inline functions.
 - g. References. *lvalue* and *rvalue*.
 - h. Exceptions.
2. Object-oriented programming fundamental paradigms in C++:
 - a. Objects and classes.
 - b. Declaration of classes and definition of member functions.
 - c. Creating of objects and working with them.
 - d. Access modifiers and accessor functions.
 - e. Constructors.
 - f. Initialization of member variables.
 - g. Destructors.
 - h. Inline member functions.

- i. Aggregation.
 - j. Inheritance.
 - k. Protected members and deriving modes.
 - l. Polymorphism, virtual functions and late binding.
- 3. Deeper into C++:
 - a. Structs in C++.
 - b. Copy constructor.
 - c. Pointer *this*.
 - d. Friends.
 - e. Operator overloading.
 - f. Assignment overloading.
 - g. Static members.
 - h. Constant members.
 - i. Constant objects.
 - j. C++ casting operations.
 - k. New keywords: *bool* and *nullptr*.
- 4. Most widely used C++ standard classes
 - a. Namespaces and the *using* directive.
 - b. Overview of C++ standard library.
 - c. Input / output streams. Global objects *cin* and *cout*.
 - d. File operations in C.
 - e. File streams in C++.
 - f. Standard exceptions.
 - g. Strings and operations with them.
 - h. Stringstreams.
- 5. Linear data structures:
 - a. Arrays and linked lists.
 - b. Operations with single linked lists.
 - c. Doubly linked lists, circular lists, skip lists.
 - d. Stack, queue and deque.
 - e. Searching algorithms: sequential search, binary search, self-organizing search.
 - f. Merge sort.
- 6. Nonlinear data structures:
 - a. Trees and binary search trees.
 - b. Binary tree operations: inserting and removing nodes, searching.
 - c. Tree traversal algorithms.
 - d. Balancing and AVL trees.
 - e. Splay trees.
 - f. Bitwise operations and bitfields in C / C++.
 - g. Digital search trees and binary tries..
 - h. Multiway trees: B-trees, 2-3-4 trees.
 - i. Red-black trees.
 - j. Priority queues and leftist trees.
 - k. Hashing: hash functions, collision handling.
- 7. C++ containers:
 - a. Class templates and function templates.
 - b. New keywords: *auto*.

- c. Constructor initializers.
 - d. Vectors.
 - e. Iterators.
 - f. Lists.
 - g. Forward lists.
 - h. Range-based *for*-loop.
 - i. Convenience container classes: array, stack, queue, etc.
 - j. Pairs.
 - k. Maps.
8. C++ standard algorithms
- a. Pointers to functions.
 - b. Lambda expressions.
 - c. Function wrappers.
 - d. Standard searching algorithms (*find*, *find_if*, *min*, *max*, etc.).
 - e. Standard algorithm *for_each*.
 - f. Standard filling algorithms (*generate*, *fill*, etc.)
 - g. Standard sorting algorithms.

Õppematerjalid

Powerpointi salidi peaksid sisaldama kogu kursuses käsitletavat materjali. Neilt leiab ka linke netis olevatele materjalidele. Kodu slaidide uurimisest siiski ei piisa, sest paljud detailid ja selgitused esitatakse loengutel üksnes suuliselt. See ei ole e-kursus.

C++ keeles programmeerimise kohta on kirjandust väga palju ning põhimõtteliselt võib igaüks leida endale sobiva raamatu. Probleem on selles, et raamatukogudes leidub neid parimal juhul vast üks või kaks eksemplari, e-poodides (näiteks Amazon.de või Amazon.se) on hinnad aga kõrged.

Mõningad autorid, kelle teoseid võib soovitada:

- Herbert Schildt – väga viljakas autor, tema C++ õpikud olid väga populaarsed. Viimane raamat temalt ilmus aastal 2014, seetõttu C++ standardisse kuuluvaid klasse, konteinereid ja algoritme käsitletakse neis vähe või üldse mitte. Enne aastat 2000 ilmunud õpikud on vananenud ning kasutud.
- Ivor Horton – samuti paljude väga heade õpikute autor, neid leidub ka meie raamatukogus. Tema viimane raamat on *Beginning C++ 23: From Beginner to Pro*, ISBN 978-1484293423, aastast 2023. Kuid siinse kursuse jaoks sobivad ka vanemad väljaanded.
- Marc Gregoire – temalt on 6 raamatut pealkirjaga *Professional C++*, neist viimane on ilmunud aastal 2024, ISBN 978-1394193172. Väga põhjalik, kuid pigem käsiraamat kui õpik. Elukutselisele programmeerijale äärmiselt vajalik. Mõned eksemplarid on ka meie raamatukogus.
- Marc Allen Weiss – temalt on mitu raamatut pealkirjaga *Data Structures & Algorithms in C++*. Neist viimane on aastast 2013 ISBN 978-0132847377. C++ õpikuks nad ei sobi, kuid andmestruktuute ja algoritme on käsitletud väga põhjalikult. Siinse kursuse jaoks sobivad ka vanemad väljaanded.
- Adam Drozdek – temalt on mitu raamatut pealkirjaga *Data Structures and Algorithms in C++*. Neist viimane on aastast 2012 ISBN 978-8131521267. C++ õpikuks nad ei sobi, kuid andmestruktuute ja algoritme on käsitletud väga põhjalikult. Siinse kursuse jaoks sobivad ka vanemad väljaanded.

Interneti teatmikud:

- <http://msdn.microsoft.com/en-US/>
- <http://www.cplusplus.com/reference/>

Head nõu saab tihti peale lehelt <https://www.geeksforgeeks.org/c-plus-plus/>.

Konsultatsioonid

Probleemide puhul kirjutage viktor.leppikson@liewenthal.ee