Prüfungsaufgabe C/C++ im Sommersem. 2020

Prof. Dr. Carsten Link, June 17, 2020

Aufgabe

Erstellen Sie in Einzelarbeit ein C++-Programm, in dem die wesentlichen in den Materialpaketen vorgestellten C++-Themen verwendet werden.

Bespiele für ihr Szenario: kleines Programm, z.B.

- Rechentrainer für Grundschüler
- generierte Kunst (z. B. Fraktale auf AnsiConsole oder PngWriter)
- Berechnungsprogramm (Fixkosten, Zinsrechner, …)
- Ratespiel
- Spiel: Mondlandung (siehe beispielsweise https://en.wikipedia.org/wiki/Lunar_Lander_(video_game_genre))
- Spiel: Gorilla.bas (siehe beispielsweise https://en.wikipedia.org/wiki/Gorillas_(video_game))
- Spiel: Snake/Nibbles (siehe beispielsweise https://en.wikipedia.org/wiki/Nibbles_(video_game))

Bei der Wahl ihres Szenarios sind Ihnen also große Freiheiten gegeben. Nutzen Sie diese und seien Sie kreativ!

Topics

Die in den Materialpaketen vorgestellten C++-Themen sind diese:

(1) Toolchain: Komponenten
(2) Toolchain: getrennte Übersetzung *
(3) Toolchain: Fehler und Warnings *
(4) Daten: Darstellung *
(5) Daten: Datentypen *
(6) Daten: Umwandlung
(7) Daten: Bitmuster
(8) Daten: Bitoperationen
(9) Kontrollfluß: grundlegende Kontrollstrukturen *
(10) Kontrollfluß: fortgeschrittene Kontrollstrukturen
(11) benutzerdefinierte Datentypen: Abstraktionen *
(12) benutzerdefinierte Datentypen: operator overloading
(13) Object Orientation: Inheritance *
(14) Object Orientation: Interface vs. Implementation *
(15) Object Orientation: subtyping polymorhism / virtual methods *
(16) Object Orientation: Entity Types vs. Value Types

(17) Object Orientation: Object Lifecycle *
(18) frühe Bindung von Methoden und Funktionen
(19) späte Bindung von Methoden und Funktionen
(20) einfache Templates
(21) wichtige Container der C++-Standardbibliothek *
(22) wichtige Algorithmen der C++-Standardbibliothek
(23) Design Patterns *
(24) Idiome *
(25) Best Practices: 0 bis 33
(26) häufige Fehlerquellen
(27) Objektkopien
(28) smart pointers

Es sollten insgesamt von Ihnen mindestens 14 Topics in ihrem Code umgesetzt werden. Mit * markierten Topics gehören zu den wichtigen – diese sollten bevorzugt umgesetzt werden! Die Topics sollten im Quellcode in Kommentaren aufgezeigt werden:

- z.B. //Topic: virtual method
- z.B. //Topic: Best Practice 21 "…"
- z.B. /*Topic: operator overloading

Randbedingungen

- ich werde das .tar entpacken, build.sh aufrufen – dann sollte es auf der bereitgestellten Linux-VM compilieren
- keine externen Bibliotheken (ggf. header-only, bzw alles mitgeliefert)
- kein komplexer build
- Sie können externe Quellen verwenden, müssen diese aber angeben (bzw. beilegen damit es zur Benotung compiliert)

Abgabe

Die Abgabe ist am 18.07.2020 fällig. Laden Sie in den Moodle-Kurs eine tar- oder zip-Datei mit

- README.txt (UTF-8)
- `*.cpp` `*.hpp` `*.sh` und weitere nötige Dateien (z. B. wenn ihr Programm eine .csv-Datei liest)
- keine `*.o` keine `*.out`!
- maximal 1 MB groß

Bewertungskriterien

An diesen Bewertungskriterien können Sie sich orientieren:

- Basics:
  - Compiliert ohne Fehler und Warnungen auf der bereitgestellten Linux-VM
  - läßt sich starten, bedienen und beenden
  - Umfang (darf nicht zu gering sein)
- Einhaltung der best practices
  - Strukturierung des Codes
  - Benennung der Variablen, Methoden, Klassen
  - Verständlichkeit/Lesbarkeit des Codes
- Abdeckung der Topics: Quantität
- Abdeckung der Topics: Qualität
- Pluspunkte: Eleganz im Code
- Pluspunkte: Robustheit (falsche Eingabedaten behandeln)
- …