

Лабораторијска вежба број 2 из Објектно оријентисаног програмирања I

У сваком задатку где има смисла обезбедити конструкторе, деструктор и оператор за доделу вредности који су потребни за безбедно коришћење класа. Користити операторске функције где је то назначено у тексту задатка.

1) Написати на језику C++ следеће класе:

- **Време** се задаје бројем часова h , минута m и секунди s (подразумевано 0,0,0; не треба проверавати исправност). Могу да се дохвате делови времена (три методе), да се испита да ли су два времена једнака ($t1==t2$), да се време учита из улазног тока ($ut>>t$) и да се време упише у излазни ток ($it<<t$) у облику $h:m:s$.
- **Листа** времена се ствара празна после чега се времена додају на крај листе ($lst+=t$). Може да се одреди број појављивања датог времена у листи и да се листа упише у излазни ток ($it<<lst$) у облику $[t1,t2,...]$.

Написати на језику C++ програм који прочита с главног улаза број времена, читајући времена с главног улаза направи листу, испише направљену листу на главном излазу, прочита једно време с главног улаза, одреди и испише на главном излазу број појављивања тог времена у листи.

2) Написати на језику C++ следеће класе:

- **Мерење** температуре се задаје измереном реалном температуром (подразумевано 20°C) и реалним координатама x и y (подразумевано (0,0)) места на којем је извршено мерење. Могу да се дохвате мерно место (свака координата посебно) и температура, да се израчуна растојање између два мерна места, да се мерење прочита из улазног тока ($ut>>mer$) и да се упише у излазни ток ($it<<mer$) у облику $[температура,x,y]$.
- **Збирка** мерења садржи низ мерења. Ствара се празна задатог капацитета (подразумевано 10) после чега се мерења додају једно по једно ($zbr+=mer$; ако се низ препуни, повећава му се капацитет за 5 места). Може да се дохвати број мерења у збирци, да се приступи мерењу са задатим редним бројем ($zbr[i]$; индекс изван опсега прекида програм), да се одреди средња температура мерења која су извршена унутар круга задатог полупречника (подразумевано 1) и центра у задатој тачки (подразумевано (0,0)) и да се садржај збирке упише у излазни ток ($it<<zbr$) у облику $\{мерење,...,мерење\}$.

Написати на језику C++ програм који читајући податке с главног улаза направи збирку мерења и одреди средњу температуру унутар круга задатог полупречника око мерења задатог редног броја у збирци.

3) Написати на језику C++ следеће класе:

- **Одбојкаш** има број, висину, улогу у тиму (вредност из скупа LIBERO, PRIMAC, TEHNICAR, KOREKTOR, BLOKER) и место на терену (1-6). Место на терену се може променити. Сви наведени подаци могу да се дохвате. Може се испитати да ли је одбојкаш мањи од задате висине (`odb<vis`). Одбојкаш може да се упише у излазни ток (`it<<odb`) у облику `број(висина, улога) [место]`.
- **Тим** са највише 6 играча се ствара празан, након чега се могу додавати показивачи на одбојкаше, један по један (`tim+=odb`). Из тима се могу избацити одбојкаши са улогом коректора и мањом висином од задате (`tim-=vis`). Тим може да се упише у излазни ток (`it<<tim`). Тим не може да се копира ни на који начин.

Написати на језику C++ интерактиван програм (с менијем) са следећим операцијама:

- прочитај одбојкаша с главног улаза и укључи га у тим,
- прочитај висину и избаци све коректоре са мањом висином из тима,
- испиши тим на главном излазу,
- заврши с радом.

4) Написати на језику C++ следеће класе:

- **Вагон** има једнозначан аутоматски додељен целобројан идентификатор и број седишта. Вагон је већи од другог вагона (`wagon1>wagon2`), ако је број седишта већи. Вагон може да се упише у излазни ток (`it<<wagon`) у облику `ид_вагона[бр_седишта]`.
- **Композиција** вагона неограниченог капацитета се ствара празна, након чега се вагони могу додавати један по један (`kompozicija+=wagon`) на крај композиције. Може да се уклони вагон с краја композиције (`kompozicija--`) или с почетка композиције (`--kompozicija`), могу да се уклоне сви вагони већи од задатог, може да се дохвати број вагона у композицији и да се композиција упише у излазни ток (`it<<kompozicija`). Композиција не може да се копира ни на који начин.

Написати на језику C++ интерактиван програм (с менијем) са следећим операцијама:

- прочитај вагон с главног улаза и стави у композицију,
- избаци последњи вагон из композиције,
- избаци први вагон из композиције,
- прочитај вагон са главног улаза и избаци из композиције све вагоне веће од њега,
- испиши садржај композиције на главном излазу,
- заврши с радом.

НАПОМЕНЕ:

- а) Потребно је решавати искључиво задатак чији се број добије на почетку вежбе.
- б) За израду лабораторијске вежбе, на располагању је **120** минута.
- в) Дозвољено је коришћење оригиналних књига и збирки задатака (не фотокопија).
- г) Није дозвољено коришћење унапред припремљених решења у било којем облику. Студент који користи унапред припремљена решења, биће удаљен уз анулирање поена на свим лабораторијским вежбама.
- д) У току израде лабораторијске вежбе, дежурни може студентима да постаља питања у вези њихових решења, што може утицати на број освојених поена на лабораторијској вежби.
- ђ) Студент може бити позван на накнадну одбрану рада, која може да утиче на број поена. Непојављивање студента на одбрани или показивање вишег степена неразумевања сопственог решења повлачи анулирање поена на свим лабораторијским вежбама.
- е) Сваку класу стављати у засебне датотеке (обавезно `.h`, по потреби и `.cpp`) и засебно програм (`.cpp`) – све на мрежном уређају `Rad(L:)`.
- ж) Оцене радова биће објављене на *Web*-у на адреси: `home.etf.rs/~kraus/` (одреднице: *настава* | <име предмета> | *оцене* | *колоквијуми*).