

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, као и полиморфно копирање и писање објеката полиморфних класа. Користити операторске функције где је то назначено у тексту задатка.

1) Написати на језику C++ следеће класе:

- **Купа** се задаје реалним полупречником r и висином h (подразумевано 1 и 2). Може да се израчуна запремина купе ($V=r^2\pi h/3$), да се испита да ли је запремина једне купе мања од друге ($k_1 < k_2$) и да се купа упише у излазни ток ($it << k$) у облику r, h .
- **Зарубљена купа** је купа чији је врх одсечен на растојању d од врха (подразумевано 1; $V=r^2\pi(h^3-d^3)/3h^2$). У излазни ток се пише у облику r, h, d .
- **Складиште** може да садржи произвољан број купа произвољне врсте. Ствара се празно после чега се купе додају једна по једна ($skl += k$). Складиште не сме да се копира ни на који начин. Може да се одреди број купа у складишту чије се запремине налазе између две задате вредности ($skl(v_1, v_2)$) и да се садржај складишта упише у излазни ток ($it << skl$) у облику $[k|k|\dots|k]$, где је k – резултат писања једне купе.

Написати на језику C++ програм који читајући податке са главног улаза направи складиште са неколико купа различитих врста, испише садржај складишта на главном излазу, испише колико купа у складишту имају запремине између две прочитане вредности и понавља претходне кораке све док не прочита неки сигнал за завршетак.

2) Написати на језику C++ следеће класе:

- Апстрактном **отпорнику** може да се одреди отпорност и да се упише у излазни ток ($it << r$).
- **Прост** отпорник има задату отпорност (подразумевано 1Ω). У излазни ток се уписује вредност отпорности.
- Апстрактан **сложен** отпорник је отпорник који се састоји од отпорника произвољне врсте. Ствара се празан задатог капацитета (подразумевано 10), после чега се отпорници додају појединачно ($R += r$; повратна вредност је индикатор успеха). У излазни ток се уписује у облику (r, r, \dots, r) , где је r – резултат писања једног отпорника.
- **Редна** веза отпорника је сложен отпорник. Отпорност је једнака збиру отпорности садржаних отпорника. У излазни ток се уписује у облику **red**(r, r, \dots, r).

Написати на језику C++ програм који направи једну редну везу отпорника која садржи неколико отпорника различите врсте и испише је на главном излазу рачунара. Користити константне параметре (не треба ништа учитавати с главног улаза).

3) Написати на језику C++ следеће класе:

- Апстрактном **предмету** може да се одреди величина (реалан број) и да се упише у излазни ток (`it<<predm`).
- **Збирка** може да садржи одређен број предмета. ствара се празна задатог капацитета, после чега се предмети додају појединачно (`zbirka+=predm`; повратна вредност је индикатор успеха). Може да се израчуна збир величина садржаних предмета и да се упише у излазни ток (`it<<zbirka`, сваки предмет у новом реду).
- **Ставка рачуна** је предмет који садржи назив артикла (ниска), јединичну цену и количину (реални бројеви). Величину предмета представља вредност робе у ставци.
- **Рачун** је збирка ставки рачуна допуњена именом купца и датумом куповине (дугачак цео број по формату `gggmmdd`). У излазни ток се уписује име купца, датум куповине, списак ставки и укупну вредност робе.

Написати на језику C++ програм који направи рачун са неколико ставки рачуна и испише га на главном излазу. Користити константне параметре (не треба ништа учитавати с главног улаза).

4) Написати на језику C++ следеће класе:

- **Тачка** у тродимензионалном простору се задаје координатама x , y и z (подразумеване вредности су $(0,0,0)$). Може да се израчуна растојање од друге тачке $(t1-t2)$ и да се тачка упише у излазни ток (`it<<t`) у облику (x,y,z) .
- **Мерење** се описује паром (температура, време), при чему је температура реалан, а време цео број такав да се дневно време изражава у `ms` мерено од 00:00.
- **Мерна тачка** је тачка која садржи низ мерења задате дужине у задатој тачки простора. Може да се приступа мерењу са задатим редним бројем (`mt[ind]`) и да се одреди средња температура у задатом интервалу времена (`mt(t1,t2)`).
- **Збирка мерних тачака** може да садржи задат број мерних тачака. Ствара се празна, после чега се тачке додају појединачно (`zmt+=mt`; препуњавање низа прекида програм). Може да се дохвати капацитет и број попуњених места у збирци и да се одреди средња вредност средњих температура у свим тачкама збирке у задатом интервалу времена (`zmt(t1,t2)`).

Написати на језику C++ програм који направи пример збирке мерних тачака и испише средњу вредност средњих температура у датом интервалу времена на главном излазу. Користити константне параметре (не треба ништа учитавати с главног улаза).

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*).