

Лабораторијска вежба број 2 из Објектно оријентисаног програмирања I

У сваком задатку где има смисла обезбедити конструкторе, деструктор и операторе за доделу вредности који су потребни за безбедно и ефикасно 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 ($k1 < k2$) 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 ($skl(V1, V2)$) 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:

- Апстрактно **тело** има специфичну тежину и једнословну ознаку врсте. Може да му се одреди запремина и тежина и да му се дохвати ознака врсте. Може да се упише у излазни ток ($it << t$).
- Дрвена **лопта** је тело задато полупречником и специфичном тежином одговарајућег дрвета. Ознака врсте је 'L'.
- Дрвени **ваљак** је тело задато полупречником основе, висином и специфичном тежином одговарајућег дрвета. Ознака врсте је 'V'.
- Дрвена **глиста** може да садржи произвољан број тела (чланака), али тако да не могу бити два чланка исте врсте један до другог. Глиста се ствара празна после чега јој се чланци додају један по један на крај ($gl += t$). Додавање погрешног чланка изазива прекид програма. Глиста може да се израчуна тежина и да се упише у излазни ток ($it << gl$) у облику $[t | t | \dots | t]$, где је t – резултат писања једног чланка.

Написати на језику C++ програм који читајући податке са главног улаза направи једну глисту, испише је на главном излазу, израчуна и испише њену тежину и понавља претходне кораке све док не прочита неки сигнал за завршетак.

3) Написати на језику C++ следеће класе:

- **Документ** садржи јединствен, аутоматски генерисан целобројан идентификатор и низ пасуса текста (један пасус је једна ниска) задатог капацитета (подразумевано 5). Ствара се празан, после чега се пасуси додају један по један (`dok+=pasus`). Може да се дохвати број пасуса, да се одреди дужина целог текста и да се документ упише у излазни ток (`it<<d`) тако што се у првом реду испише идентификатор, а затим испишу сви пасуси, са једним редом прореда између њих.
- **Ауторизован документ** је документ са потписом аутора (ниска). Потпис аутора се сматра посебним пасусом и улази у дужину текста.
- **Збирка** докумената садржи документе различитих врста. Ствара се празна, задатог капацитета, после чега се документи додају један по један (`zbr+=dok`). Може да се израчуна укупна дужина текстова свих докумената и да се упише у излазни ток (`it<<zbr`), тако што се редом испишу документи са једним редом звездица између. Збирка не може да се копира ни на који начин.

Написати на језику C++ програм који читајући податке са главног улаза направи једну збирку ауторизованих и неауторизованих докумената, затим испише на главном излазу збирну дужину текстова у збирци и понавља претходне кораке све док не прочита неки сигнал за завршетак.

4) 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 `ggggmmdd`). 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).

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: <http://rti.etf.bg.ac.rs/rti/ir2ool/index.html>

