

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:

- **Збирка** целих бројева може да садржи задати број елемената. Ствара се празна са задатим почетним капацитетом (подразумевано 10) и кораком повећавања капацитета (подразумевано 4). Бројеви се додају један по један на крај збирке ($z+=b$) уз повећање капацитета по потреби. Може да се дохвати број попуњених места у збирци, да се дохвати вредност задатог елемента збирке ($z[ind]$ – спречити могућност промене вредности елемента) и да се садржај збирке упише у излазни ток ($it\ll z$) у облику (b, b, \dots, b) .
- **Уређена** збирка целих бројева је збирка чији су елементи уређени по неоппадајућем редоследу вредности. Нови бројеви се умећу на одговарајућа места тако да збирка остаје уређена.
- **Скуп** целих бројева је збирка чији су сви елементи међусобно различити. При додавању броја, ако исти већ постоји у скупу, не ради се ништа.

Написати на језику C++ програм који читајући податке са главног улаза направи збирку одабране врсте, испише садржај збирке на главном излазу и понавља претходне кораке све док као ознаку врсте збирке не прочита неку специјалну вредност.

2) Написати на језику C++ следеће класе:

- **Вектор** у простору се задаје реалним координатама x , y и z (подразумевано $(0,0,0)$). Може да се израчуна збир $v1+v2$ и разлика $v1-v2$ вектора као и апсолутна вредност $\sim v$ вектора. Вектор може да се упише у излазни ток ($it\ll v$) у облику (x, y, z) .
- **Изломљена линија** садржи низ вектора који одређују њена темена. Ствара се празна задатог максималног броја темена (подразумевано 3) после чега темена могу да се додају једно по једно ($l+=v$, повратна вредност је индикатор успеха, неуспех је ако се низ препуни). Изломљена линија не сме да се копира ни на који начин. Може да јој се дохвати број темена, да јој се израчуна дужина и да се

упише у излазни ток ($it \ll l$) у облику $[t|t|...|t]$, где је t – резултат писања једног темена.

- **Многоугао** је изломљена линија код које је последње теме спојено с првим. У излазни ток се пише у облику Ml , где је l – резултат писања одговарајуће изломљене линије.

Написати на језику C++ програм који читајући податке с главног улаза направи многоугао, испише га на главном излазу, одреди његов обим и понавља претходне кораке све док за број темена не прочита недозвољену вредност.

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 ($it \ll d$) 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 ($it \ll zbr$), 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 ($it \ll predm$).
- **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 ($it \ll zbirka$, 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).

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/ir2001/index.html>