ЕЛЕКТРОТЕХНИЧКИ ФАКУЛТЕТ
Одсек за СИ Београд 07-08.12.2006.

Лабораторијска вежба број 2 из Објектно оријентисаног програмирања I

1) Написати на језику *C++* следеће класе (класе опремити оним конструкторима, деструктором и оператором за доделу вредности који су потребни за безбедно коришћење класа):

- ***Тачка*** у простору задаје се реалним координатама x, y и z (подразумевано (0,0,0)). Може да се израчуна удаљеност тачке d од координатног почетка и да се тачка упише у излазни ток (`t.pisi(it)`) у облику **(**x**,**y**,**z**)**.
- ***Пондерисана тачка*** у простору је тачка у простору са додатним тежинским фактором q (подразумевано 1). Удаљеност од координатног почетка се рачуна као $d \times q$, где је d – природна уадљеност тачке од координатног почетка. У излазни ток се пише у облику **(**x**,**y**,**z**)*** q.
- ***Многоугао*** садржи низ од задатог броја тачака произвољне врсте које чине темена многоугла. Ствара се празан задатог капацитета (подразумевано 3) после чега тачке могу да се додају једна по једна (`m+=t`, ако се низ препуни, програм се прекида). Многоугао не сме да се копира ни на који начин. Може да се дохвати број темена, да се пронађе теме које је најближе координатном почетку и да се моногоугао упише у излазни ток (`it<<m`) у облику **[**t**|**t**|…|**t**]**, где је t – резултат писања једног темена.

Написати на језику *C++* програм који читајући податке са главног улаза направи многоугао, испише га на главном излазу, пронађе и испише на главном излазу теме које је најближе координатном почетку и понавља претходне кораке све док за број темена не прочита недозвољену вредност.

2) Написати на језику *C++* следеће класе (класе опремити оним конструкторима, деструктором и оператором за доделу вредности који су потребни за безбедно коришћење класа):

- ***Квадар*** се задаје реалним ивицама a, b и c (подразумевано 1, 1 и 1). Може да се израчуна запремина квадра, да се испита да ли два квадра имају једнаке ивице (`kv1==kv2`) и да се квадар упише у излазни ток (`it<<kv`) у облику **K(**a**,**b**,**c**)**.
- ***Листа*** квадара може да садржи произвољан број квадара. Ствара се празна после чега се квадри додају један по један на крај листе (`lst+=kv`). Листа може да се упише у излазни ток (`it<<lst`) у облику **[**k**|**k**|…|**k**]**, где је k – резултат писања једног квадра.
- ***Скуп*** квадара је листа квадара у којој су сви квадри међусобно различити. Покушај додавања постојећег квдара нема никаквог ефекта.

Написати на језику *C++* програм који читајући податке са главног улаза направи листу и скуп квадара, испише добијене резултате на главном излазу и понавља претходне кораке све док не прочита неки сигнал за завршетак.

3) Написати на језику *C++* следеће класе (класе опремити оним конструкторима, деструктором и оператором за доделу вредности који су потребни за безбедно коришћење класа):

- ***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 (`k.pisi(it)`) 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**`|`**…**`|`**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.

4) Написати на језику *C++* следеће класе (класе опремити оним конструкторима, деструктором и оператором за доделу вредности који су потребни за безбедно коришћење класа):

- ***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<<z`) у облику **`(`**b**`,`**b**`,`**…**`,`**b**`)`**.
- ***Уређена*** збирка целих бројева је збирка чији су елементи уређени по неопадајућем 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 сдржај збирке на главном излазу и понавља претходне кораке све док као ознаку врсте збирке не прочита неку специјалну вредност.

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 фотокопије).

г) Није дозвољено коришћење унапред припремљених решења у било којем облику. Студент који користи недозвољене материјале, биће удаљен и **губи право на полагање колоквијума**.

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) Текст решења распоредити у `*.h` и `*.cpp` датотеке (не стављати цело решење у једну датотеку).

е) Оцене радова биће објављене на *Web*-у на адреси: `galeb.etf.bg.ac.yu/~kraus/` (одреднице: *настава* | <име предмета> | *оцене* | *колоквијуми*).