

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):

- **Вектор** у простору се задаје реалним координатама x , y и z (подразумевано $(0,0,0)$). Може да се израчуна збир v_1+v_2 и разлика v_1-v_2 вектора као и апсолутна вредност $\sim v$ вектора. Вектор може да се упише у излазни ток (`it<<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 ($l+=v$, ако се низ препуни, програм се прекида). 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 (`l.pisi(it)`) 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.

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 квадрата, да се испита да ли два квадрата имају једнаке ивице ($k v_1 == k v_2$) и да се квадар упише у излазни ток (`it<<kv`) у облику $K(a,b,c)$.
- **Листа** квадрата може да садржи произвољан број квадрата. Ствара се празна после чега се квадрати 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 квадрата.
- **Скуп** квадрата је листа квадрата у којој су сви квадрати међусобно различити. Покушај додавања постојећег квадрата нема никаквог ефекта.

Написати на језику C++ програм који читајући податке са главног улаза направи листу и скуп квадрата, 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 ($k_1 < k_2$) 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 $[k|k|\dots|k]$, 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 има 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):

- **Тело** има специфичну тежину и једнословну ознаку врсте. Може да му се одреди запремина и тежина (подразумевано су једнаке нули) и да му се дохвати врста. Може да се упише у излазни ток (`t.pisi(it)`).
- Дрвена **лонта** 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. Глисти 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.

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 унапред припремљена решења, биће удаљен уз анулирање поена на свим лабораторијским вежбама и колоквијумима.
- 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 и колоквијумима.
- е) Текст решења распоредити у *.h и *.src датотеке (не стављати цело решење у једну датотеку).
- ж) Оцене радова биће објављене на Web-у на адреси: kondor.etf.bg.ac.yu/~kraus/ (одреднице: *настава* | <име предмета> | *оцене* | *колоквијуми*).