

Задатак:

Написати двије класе: *Artikal* и *Racun*, те низ *spisak_artikala* дужине 10 који представља списак објеката типа *Artikal*.

Сљедеће ставке се односе на класу *Artikal*.

a) Класа *Artikal* као приватне атрибуте има:

1. *sifra* артикла типа *const int*, овај атрибут треба бити јединствен за сваки објекте из низа *spisak_artikala*.
2. *naziv* артикла типа *char **
3. *kolicina* артикла типа *float*. Вриједност овог атрибута не може бити негативна
4. *cijena* артикла типа *float*. Вриједност овог атрибута мора бити позитивна

b) Написати конструктор класе *Artikal* који за аргументе узима један податак типа *int*, једну ниску(податак типа *char **) и два податка типа *float*. Наведени аргументи представљају вриједности за сљедеће атрибуте: *sifra*, *naziv*, *kolicina* и *cijena*.

c) За сваки од приватних атрибута написати јавне методе које враћају његову вриједност

d) Написати јавни метод *promijrni_kolicinu* који као улазни параметар има податак типа *float* и вриједност атрибута *kolicina* повећава за вриједност улазног параметра (односно самањује ако је вриједност улазног параметра негативна). Уколико је вриједност атрибута *kolicina* негативна, до наведене промјене ипак неће доћи. Овај метод као повратну вриједност враћа *True* уколико је промјена успјешна, у противном враћа *False*.

e) Креирани објекат типа *Artikal* убацити у низ *spisak_artikala*.

f) Написати деструктор класе *Artikal* који посматрани објекат тип *Artikal* брише из низа *spisak_artikala*.

Начин организације низа *spisak_artikala* је остављен да студент сам изабере.

Сљедеће ставке се односе на класу *Racun*.

a) Класа *Racun* има сљедеће приватне атрибуте:

1. *kolicina*, овај атрибут је низ података типа *float*, дужине 10, те вриједност од *kolicina[i]* показује колико је урачунато артикла који се налази на *i*-тој позицији низа *spisak_artikala*.
2. *vrijeme*, овај атрибут је типа *const time_t* и он представља вријеме издавања рачуна
3. *iznos*, атрибут типа *float*.

b) Написати конструктор класе по свом избору.

c) Написати деструктор класе по свом избору

d) Написати јавни метод *dodaj_artikal* који захтијева унос шифре артикла, те количине за обрачун посматраног артикла. У случају да је расположива количина артикла са унесеном шифром мања од унесене количине или не постоји артикал са унесеном шифром, испишује се порука: "*GRESKA: zahtjev je odbijen!*". У супротном унесена количина се чува у низу *kolicina* на описани начин, те се ажурира вриједност атрибута *iznos* и расположива количина артикла. Водити рачуна ако је унесен артикал који је раније додат на рачун, не додавати га опет већ промијенити количину која се односи на посматрани артикал.

e) Написати јавни метод *obrisi_artikal* који захтијева унос шифре артикла. Уколико се на рачуну налази артикал са унесеном шифром, он бива обрисан са рачуна, те се ажурира расположива количина артикла и износ рачуна. У овом случају се испише порука: "*Artikal <naziv artikla> je uspjesno obrisan sa racuna*". У супротном се испише порука: "*GRESKA: zahtjev je odbijen*"

f) Написати јавни метод *vрати_iznos* који враћа вриједност атрибута *iznos*.

g) Написати јавни метод *vрати_vrijeme_izdavanja_racuna* који враћа вриједност атрибута *vrijeme*.

h) Преклопити оператор *+* тако да сума два рачуна буде нови рачун који има све артикле који се налазе на бар неком од "сабирака", док количина артикла треба бити једнака:

- I. суми количина посматраног артикла са оба рачуна, ако се артикал налази на оба рачуна
- II. количини посматраног артикла са оног рачуна гдје се артикал налази, ако се артикал налази на само једном рачуну

Вриједност атрибута *вријеме* треба да буде тренутно системско вријеме. Те артикле са рачуна чију суму рачунамо треба избрисати и њихову вриједност поставити на 0.00

i) Преклопити оператор << тако да испис рачуна буде у сљедећем облику:

Vrijeme izdavanja racuna: <вриједност атрибута vrijeme>

<назив артикла 1> <количина артикла 1> <износ артикла 1>KM

.

.

.

<назив артикла к> <количина артикла к> <износ артикла к>KM

Iznos racuna: <вриједност атрибута iznos>

Обавезно написати тестни програм у коме ће се тестирати све наведене методе.