

## Динамичка алокација на меморија и композиција

1. Да се дефинира класа **Student**, за која се чуваат информации за:

- името и презимето на студентот (низа од 50 знаци),
- насоката на која е запишан студентот (низа од 4 знаци),
- динамичко поле од објекти од класата **Ispit** во кое се чува листа на положените испити.

Во класата **Ispit** се чува информација за името на испитот (низа од 20 знаци), бројот на кредити на испитот (цел број) и добиената оценка (цел број). За класата **Student** да се дефинира конструктор со 2 аргументи (полето во кое се чуваат положените испити на почетокот е празно). За класата **Ispit** да се дефинира конструктор со три аргументи. Во рамките на класата **Student** да се преотвори операторот `+` за додавање на нов испит во листата на положени испити. Да се преоптовари и операторот `==` за споредба на два студенти според насоката на која се запишани, ако се запишани на иста насока функцијата враќа **true**, во спротивно - **false**. Надвор од класите да се дефинира функција **details** која како влезен аргумент прима објект од класата **Student** и печати информација во следниот облик:

```
ime i prezime: <името на студентот>
broj na polozeni ispiti: <бројот на положени испити>
vkupen broj na krediti: <бројот на освоени кредити>
prosečna ocenka: <пресечната оценка на студентот>
```

2. Да се напише класа за опишување на компјутери. За компјутерите се чуваат:

- информации за моделот (низа од 20 знаци),
- производител (низа од 20 знаци),
- RAM меморија (цел број),
- капацитет на хард диск (цел број),
- брзина на процесорот (децимален, број)
- цена (цел број).

За класата да се напишат:

- default конструктор,
- copy конструктор,
- деструктор,
- set и get методи,
- метод за печатење на информациите за компјутерот.

Дополнително да се напише класа за опис на компанија за продавање на компјутери за која се чуваат информации за името (низа од 20 знаци), адреса (низа од 30 знаци), низа од компјутери (максимум 50) и број на компјутери. За класата да се напишат:

- конструктор со 4 аргументи,
- деструктор,
- метод за печатење на карактеристиките на компјутерите сортирани според цената,
- метод којшто како резултат го враќа компјутерот со најголема брзина и најмала цена.

Да се напише main функција за тестирање на имплементираниите методи.

3. Да се напише класа за опишување на датум. За датумот се чуваат информации за денот, месецот и годината (цели броеви). За класата да се напишат соодветните конструктори и да се преоптоварат операторите = и << (датумот се печати во облик во облик dd.mm.yyyy).

Да се напише класа за опишување на SMS пораки. За пораките се чуваат информации за содржината (динамички алоцирана низа), за цената на пораката (цел број), и за датумот на испраќање (објект од класата датум). Цената на пораката зависи од должината на содржината при што за секои 160 знаци, се плаќа 5 денари. За класата да се напишат конструктори, деструктор да се преоптоварат операторите = и <<.

На крај да се напише класа корисник за кој се чуваат информации за името презимето на корисникот (низа од 30 знаци), низа од пораки (динамички алоцирана) и бројот на пораките. Да се напишат конструктори, деструктор, метод за пресметка на вкупно потрошена вредност за пораки и да се преоптоварат операторите + = (за праќање на нова порака) и <<.

Да се напише main функција за тестирање на имплементираниите методи во класите.