

توابع مجازی

در این سوال قصد داریم تا شما را با توابع مجازی آشنا کنیم . سه کلاس *Student* و *Person* , *Professor* ایجاد کنید.

در کلاس *Person* دو عضو داده ای سن (*age_*) و نام (*name*) وجود دارد. دو کلاس *Student* و *Professor* از کلاس *Person* ارث میبرند.

در کلاس *Professor* باید دو عضو داده ای *publications* و *curid* وجود داشته باشد همچنین دو تابع عضو زیر وجود دارند :

- تابع *getdata* : وظیفه ی گرفتن اطلاعات استاد شامل نام و سن و انتشارات استاد میباشد.
- تابع *putdata* : وظیفه ی نمایش اسم ، سن ، انتشارات و *curid* را دارد.

در کلاس *Student* دو عضو داده ای نمره (*marks*) که یک آرایه 6 تایی و *curid* داریم. در این کلاس دو تابع عضو زیر وجود دارند :

- تابع *getdata* : وظیفه ی گرفتن اطلاعات دانشجو شامل نام و سن و نمرات دانشجو در شش درس میباشد.
- تابع *putdata* : وظیفه ی نمایش اسم ، سن ، مجموع نمرات ، *curid* را دارد.

برای هر شئی که از کلاس استاد یا دانشجو ساخته میشود *curid* یک عدد ترتیب افزایشی که از یک شروع میشود به آن ها داده شوند.

این مساله را با توابع مجازی ، سازنده و مقادیر *static* حل کنید. شما میتوانید عضو های داده ای بیشتری در کلاس ها بنابر نیاز ایجاد کنید.

ورودی :

خط اول ورودی تعداد شی هایی است که میسازیم سپس در صورتی که کاربر در خط بعدی عدد یک را وارد کرد به معنای ساخت شئی استاد و در صورتی که دو را وارد کرد به معنای ساخت شئی دانشجو میباشد.

بعد از وارد کردن نوع شئی در خط بعدی موارد مورد نیاز هر شئی از کاربر گرفته میشود . برای مثال برای استاد به ترتیب اسم ، سن ، انتشارات گرفته میشود یا در صورتی که دانشجو بود به ترتیب اسم ، سن ، نمرات 6 درس گرفته میشود.

خروجی :

دو نوع خروجی بر اساس نوع شئی ها داریم :

۱. استاد : مشخصات آن استاد را با یک اسپیس در خروجی چاپ کند.

۲. دانشجو : مشخصات آن دانشجو را با یک اسپیس در خروجی چاپ کند.

مثال :

ورودی :

```
4
1
Walter 56 99
2
Jesse 18 50 48 97 76 34 98
2
Pinkman 22 10 12 0 18 45 50
1
White 58 87
```

خروجی :

```
Walter 56 99 1
Jesse 18 403 1
Pinkman 22 135 2
White 58 87 2
```

کد main :

```
int main(){

    int n, val;
    cin>>n; //The number of objects that is going to be created.
    Person *per[n];

    for(int i = 0;i < n;i++){

        cin>>val;
        if(val == 1){
            // If val is 1 current object is of type Professor
            per[i] = new Professor;

        }
        else per[i] = new Student; // Else the current object is of type Student

        per[i]->getdata(); // Get the data from the user.

    }

    for(int i=0;i<n;i++)
        per[i]->putdata(); // Print the required output for each object

    return 0;

}
```