

# • گزارش پروژه

iostream

برای استفاده از cin و cout

string

برای استفاده از نوع داده ای string

algorithm

برای استفاده از تابع sort

---

struct Sub\_ISE, struct Sub\_IT

~ دو ساختمان برای دروس دو رشته صنایع و کامپیوتر که شامل نام درس (name\_of\_sub)، نمره (score) و تعداد واحد (courseID) هستند.

~ تعریف دو آرایه با نام های subISE و subIT که هر کدام به ترتیب داری ۶ شیء از نوع SUB\_ISE و SUB\_IT هستند.

دو آرایه subISE[6] و subIT[6] برای نگهداری ۶ درس هر رشته استفاده شده است.

~ایجاد دو سازنده پیش فرض برای هر دو کلاس.  
Sub\_IT ، Sub\_ISE : (دو سازنده پیش فرض برای  
مقداردهی اولیه تعریف شده‌اند.)

```
Sub_IT() : name_of_sub(""), score(0),  
courseID(0) {}
```

```
Sub_ISE() : name_of_sub(""), score(0),  
courseID(0) {}
```

و دو سازنده با پارامتر

```
Sub_IT(string e, double f, int g) :  
name_of_sub(e), score(f), courseID(g) {}
```

```
Sub_ISE(string e, double f, int g) :  
name_of_sub(e), score(f), courseID(g) {}
```

---

ایجاد تابع `calculate_GPA()` برای محاسبه GPA با  
توجه به نمره و تعداد واحد هر درس، برای هر دو رشته.  
اگر رشته ISE باشد، معدل از `subISE` محاسبه

می‌شود و اگر IT باشد، از subIT.

تابع `displaying_info()` برای نمایش اطلاعات دانشجویان.

تابع `edit_info()` برای ویرایش اطلاعات دانشجو مانند نام، نام خانوادگی، رشته تحصیلی، شناسه دانشجویی، و همچنین اطلاعات دروس دانشجو است و بعد از ویرایش، معدل به روز رسانی می‌شود.

تابع `delete_student()` برای حذف دانشجو با شناسه ای مشخص از آرایه است.

---

بخش main:

تعداد دانشجویان در ابتدا گرفته می‌شود

~ به دلیل متغیر بودن تعداد دانشجویان یک آرایه از اشیا `Student` به اندازه `numStudents` ایجاد می‌شود و به دلیل حافظه داینامیک، حافظه را بعد از استفاده از این حافظه آن را با `delete[]` آزاد می‌کنیم.

~وارد کردن اطلاعات دانشجویی و اطلاعات دروس  
برای هر دانشجو.

ابتدا دروس مربوط به رشته او وارد می شود. سپس با  
استفاده از این اطلاعات، یک شی **Student** ساخته  
می شود

~پس از وارد کردن اطلاعات، دانشجویان بر اساس  
رشته و سپس معدل (به ترتیب نزولی) مرتب می شوند.  
که از تابع **sort** استفاده شده.

~ویرایش یا حذف دانشجو:

اگر انتخاب کاربر 1 باشد، اطلاعات دانشجویی با  
شناسه‌ی مشخص برای ویرایش دریافت می شود.  
اگر انتخاب 2 باشد، دانشجویی با شناسه‌ی مشخص  
حذف می شود.

اگر انتخاب 0 باشد، برنامه هیچ عملیاتی انجام  
نمی دهد.

کدهای رنگ:

\033[31m

\033[32m

\033[33m

\033[34m

\033[36m

پس از ویرایش یا حذف، لیست جدید دانشجویان دوباره نمایش داده می شود.

و در نهایت آزادسازی حافظه.