



برنامه نویسی پیشرفته – ترم 4001

فاز دوم پروژه شماره 4 – مکانیک اتومبیل

استاد : مهندس بطحائیان

محمدحسین الطافی - 9912358004

تغییرات نسبت به فاز قبل

- اضافه شدن سربارگذاری عملگرها برای کلاس ها
 - اضافه شدن کلاس tools برای ارث بری
 - اضافه شدن تابع virtual برای پلی مورفیسم
-
- حذف شدن پرایوت ممبر price و توابع set و get از دو کلاس قبلی و ارث بردن آن از کلاس tools

ساختار برنامه

کلاس tools (همه وسایلی که مکانیک نیاز دارد) و توابع مربوط به آن:

```
//operator haye marboot be gheyamat mesle +,-,+=,-= dar cpp neveshte shode
class tools
{
public:
    tools(int givenquality , float prc); //constructor
    void set_price(float prc); //tabe set baraye gheyamat
    float get_price() const; //tabe get baraye gheyamat
    void set_quality(int givenquality); //tabe set baraye keyfiat
    int get_quality() const; //tabe get baraye keyfiat
    virtual void printinfo() const = 0; //tabe print etelaat ke = 0 baraye abstract shodane classe tools estefade shode
    void operator+=(const int addedprice); //operatore += baraye gheyamat
    void operator-=(const int decreasedprice); //operatore -= baraye gheyamat

private:
    float price; //gheyamat
    int quality; //keyfiat
};
```

کلاس های screwdriver و carlight از این کلاس ارث می برند.

در این کلاس 2 پرایوت ممبر price و quality وجود دارند و هر کدام توابع set و get مربوط به خود را دارند.

تابع printinfo به صورت virtual تعریف شده و برای این که کلاس abstract شود، مساوی 0 است. این تابع پیاده سازی دارد و در تابع پرینت 2 کلاس دیگر، این تابع پرینت فراخوانی می شود.

اپراتور های += و -= که مربوط به قیمت هستند، پیاده سازی شده اند و در همه کلاس های تعریف شده قابل استفاده اند.

دو اپراتور + و - در کلاس های دیگر به صورت تابع دوست در نظر گرفته شده اند.

پیاده سازی اپراتورها:

```
void tools::operator+=(const int addedprice) //operatore += baraye gheymat
{
    this->set_price(this->get_price()+addedprice);
}

void tools::operator-=(const int decreasedprice) //operatore -= baraye gheymat
{
    this->set_price(this->get_price()- decreasedprice);
}

void operator+(tools &tools1 , const int addedprice) //operatore + baraye ezafe shodane gheymat
{
    tools1.set_price(tools1.get_price()+addedprice);
}

void operator+(const int addedprice, tools & tools1) //operatore + baraye ezafe shodane gheymat ba estefade az tabe ghabl
{
    tools1 + addedprice;
}

void operator-(tools &tools1 , const int decreasedprice) //operatore - baraye ezafe shodane gheymat
{
    tools1.set_price(tools1.get_price()-decreasedprice);
}
```

پیاده سازی تابع پرینت مربوط به این کلاس:

```
void tools::printinfo() const
{
    cout << "Price : " << get_price() << " $" << endl; //chape gheymat
    cout << "Quality : ";
    switch (quality) //switch case baraye quality
    {
        case 1:
            cout << "Low" << endl;
            break;
        case 2:
            cout << "Average" << endl;
            break;
        case 3:
            cout << "High" << endl;
            break;
        default:
            cout << "No such quality exists" << endl;
            break;
    }
}
```

تغییرات کلاس carlight :

```
✓ class carlight : public tools //classe carlight marboot be anvae cheragh haye mashin ke public inheritance az tools darad
{
    friend void operator+(tools &carlight1 , const int addedprice); //operatore + baraye ezafe shodane gheyamat
    friend void operator+(const int addedprice , tools & carlight1); //operatore + baraye ezafe shodane gheyamat
    friend void operator-(tools &carlight1 , const int decreasedprice); //operatore - baraye kam shodane gheyamat
    //operator haye += va -= dar classe tools tarif shode and

    public:
        carlight(color col , lighttype type , brightness brght , float prc, int quality); //tabe constructor
        void set_color(color col); //tabe set baraye color
        color get_color() const; //tabe get baraye color (az noe enum tarif shode)
        void set_light_type(lighttype type); //tabe set baraye light type
        lighttype get_light_type() const; //tabe get baraye light type (az noe enum tarif shode)
        void set_brightness(brightness brght); //tabe set baraye brightness
        brightness get_brighrness() const; //tabe get baraye brightness (az noe enum tarif shode)
        virtual void printinfo() const override; //tabe chape etelaat
        bool operator==(const carlight & carlight1); //operatore == baraye tasavie lighttype va color
        bool operator>(const carlight & carlight1); //operatore > baraye moghayese brightness
        bool operator<(const carlight & carlight1); //operatore < baraye moghayese brightness
        bool operator>=(const carlight & carlight1); //operatore >= baraye moghayese brightness
        bool operator<=(const carlight & carlight1); //operatore <= baraye moghayese brightness

    private:
        color chosencolor; //estefade az enum baraye tarife noe color
        lighttype light_type; //estefade az enum baraye tarife noe light type
        brightness chosenbrightness; //estefade az enum baraye tarife noe brightness
};
```

اپراتورهای + و - چون به صورت عضو تعریف نشده اند، به صورت دوست در نظر گرفته شده اند.

تابع printinfo چون باید در کلاس پدر وجود داشته باشد و چون بازنویسی است، در تعریف آن از override استفاده شده و در کلاس پدر به صورت virtual است.

اپراتور == برای بررسی تساوی light_type و color پیاده سازی شده و اپراتورهای <,<=,>,>= برای مقایسه brightness پیاده سازی شده اند.

تغییرات کلاس screwdriver :

```
class screwdriver : public tools//classe screwdriver marboot be anvae pich gooshti ke public inheritance az tools darad
{
    friend void operator+(tools &screw1 , const int addedprice); //operatore + baraye ezafe shodane gheyamat
    friend void operator+(const int addedprice , tools &screw1); //operatore + baraye ezafe shodane gheyamat
    friend void operator-(tools &screw1 , const int decreasedprice); //operatore - baraye kam shodane gheyamat
    //operator haye += va -= dar classe tools tarif shode and

public:
    screwdriver(screwdriver_head_size headsize , screwdriver_head_type headtype , screwdriver_length len , float prc , int quality);
    void set_head_size(screwdriver_head_size headsize); //tabe set baraye head size
    screwdriver_head_size get_head_size() const; //tabe get baraye head size (az noe enum tarif shode)
    void set_head_type(screwdriver_head_type headtype); //tabe set baraye head type
    screwdriver_head_type get_head_type() const; //tabe get baraye head type (az noe enum tarif shode)
    void set_length(screwdriver_length len); //tabe set baraye length
    screwdriver_length get_length() const; //tabe get baraye length (az noe enum tarif shode)
    virtual void printinfo() const override; //tabe chape etelaat
    bool operator==(const screwdriver &screw1); //operatore == baraye check kardane tasavie headtype
    bool operator>(const screwdriver &screw1); //operatore > baraye moghayese headsize
    bool operator<(const screwdriver &screw1); //operatore < baraye moghayese headsize
    bool operator>=(const screwdriver &screw1); //operatore >= baraye moghayese headsize
    bool operator<=(const screwdriver &screw1); //operatore <= baraye moghayese headsize

private:
    screwdriver_head_size head_size; //estefade az enum baraye tarife noe head size
    screwdriver_head_type head_type; //estefade az enum baraye tarife noe head type
    screwdriver_length length; //estefade az enum baraye tarife andaze length
};
```

اپراتورهای + و - چون به صورت عضو تعریف نشده اند، به صورت دوست در نظر گرفته شده اند.

تابع printinfo چون باید در کلاس پدر وجود داشته باشد و چون بازنویسی است، در تعریف آن از override استفاده شده و در کلاس پدر به صورت virtual است.

اپراتور == برای بررسی تساوی head_type پیاده سازی شده و اپراتور های <,<=,>,>= برای مقایسه head_size پیاده سازی شده اند.

تابع main:

در تابع main , نمونه هایی برای تست اپراتورهای پیاده سازی شده و تست اشاره گر مربوط به پلی مورفیسم تابع printf وجود دارند.

```
//teste esharegare marboot be polymorphism tabe print
// tools t;// errore chon abstracte va tabe printesh = 0 gharar dade shode va virtual ham hast
tools * c = nullptr;
c = &a;
cout << "using polymorphism and pointer for the function printf of screwdriver a: " << endl;
c->printf();
c = &d;
cout << "using polymorphism and pointer for the function printf of carlight d: " << endl;
c->printf();
```

محمدحسین الطافی

9912358004