



برنامه نویسی پیشرفته – ترم 4001

فاز سوم پروژه شماره 4 – مکانیک اتومبیل

استاد : مهندس بطحائیان

محمدحسین الطافی - 9912358004

تغییرات نسبت به فاز قبل

- اضافه شدن کلاس mechanic
- اضافه شدن وکتور برای نگهداری ابزارهای خریداری شده
- اضافه شدن منو به برنامه برای خرید ابزار, چاپ اطلاعات ابزار های خریداری شده و مقدار موجودی پول مکانیک و افزایش آن
- اضافه شدن تابع choseandset_price که بر اساس ویژگی هایی که کاربر میخواهد, قیمت آن ابزار را مشخص می کند.

ساختار برنامه

کلاس mechanic:

```
class mechanic
{
public:
    mechanic(float givenmoney, string givenname); //constructor
    void set_money(float givenmoney); //tabe set baraye money
    float get_money(); //tabe get baraye money
    void set_name(string givenname); //tabe set baraye esm
    string get_name(); //tabe get baraye esm
    vector <tools *> alltools; //baraye rahatie kar public dar nazar gerefte shode

private:
    string name;
    float money;
};
```

این کلاس پرایوت ممبرهای پول و اسم دارد که مربوط به مکانیک هستند.

وکتوری برای نگهداری ابزارهای خریداری شده نیز در نظر گرفته شده که برای راحتی کار و دسترسی آسان تر به صورت پابلیک است.

این وکتور، شامل اشاره گرهایی از کلاس tools است که با ویژگی های پلیمورفیزم میتوان به همه ابزارهای خریداری شده از هر دو کلاس carlight و screwdriver دسترسی داشت.

آپشن های منو برنامه

```
1- Buy a Screwdrier
2- Buy a Carlight
3- Print bought Screwdrivers
4- Print bought Carlights
5- Print all bought tools
6- Your balance
7- Increment your balance
8- Exit
```

- **Buy a screwdriver:** با فراخوانی تابع `makescrewdriver` ابتدا اشاره گری از نوع پیچگوشتی ساخته شده و حافظه ای برای آن `new` می شود. سپس ویژگی های پیچگوشتی درخواستی از کاربر پرسیده می شود و قیمتی برای آن تعیین می شود. اگر کاربر پول کافی داشته باشد، به وکتور ابزارها اضافه می شود، اگر کاربر پول کافی نداشته باشد، حافظه در نظر گرفته شده، `delete` می شود و به کاربر پیامی در رابطه با ناکافی بودن موجودی نمایش داده می شود.
- **Buy a carlight:** با فراخوانی تابع `makecarlight` کاری مشابه با گزینه قبل اما برای چراغ ماشین انجام می دهد.
- **Print bought Screwdrivers:** برای چاپ اطلاعات پیچگوشتی های خریداری شده استفاده می شود. (با ویژگی های پلیمورفیسم و استفاده از `dynamic_cast`، پیچگوشتی ها از وکتور ابزار پیدا می شوند و تابع `print` آن ها فراخوانی می شود).
- **Print bought Carlights:** مثل گزینه قبل اما برای چراغ ماشین های خریداری شده استفاده می شود.
- **Print all bought tools:** برای چاپ اطلاعات تمام ابزار های خریداری شده استفاده می شود.
- **Your balance:** مقدار پول کاربر را نمایش می دهد.

- Increment your balance: افزایش پول کاربر برای خرید بیشتر.
- Exit: تمامی حافظه های new شده delete می شوند و برنامه به پایان می رسد.

تابع choseandset_price

این تابع که برای هر دو کلاس پیچگوشتی و چراغ ماشین تعریف شده، با استفاده از اپراتور اورلود شده + و یک قیمت پایه، قیمت هر ابزار را بر اساس ویژگی های که کاربر می خواهد، تعیین می کند.

```
void carlight::choseandset_price() //t
{
    set_price(11);

    if(light_type == headlight)
    {
        *this + 4;
    }
    else if(light_type == fog_light)
    {
        *this + 3;
    }
    else if(light_type == blinker)
    {
        *this + 2;
    }
    else if(light_type == brake_light)
    {
        *this +3;
    }

    if(get_quality() == 1)
    {
        *this + 1;
    }
    else if(get_quality() == 2)
    {
        *this + 4;
    }
    else if(get_quality() == 3)
    {
```

تابع main:

در این تابع، اسم و پول از کاربر دریافت می شود تا یک مکانیک ساخته شود و تابع maketools (تابعی که در آن منو چاپ می شود و فراخوانی توابع دیگر انجام می شود) فراخوانی می شود.

```
int main()
{
    cout << "Welcome to the mechanic app!" << endl;
    cout << "Please enter your name: ";
    string name;
    cin >> name;
    cout << "How much money do you have?" << endl;
    float money;
    cin >> money;
    mechanic karbar(money,name);
    maketools(karbar);
}
```

محمدحسین الطافی

9912358004