

برنامه سازی پیشرفته

گزارشکار فاز سوم

هانیه اکبری

۹۹۱۲۳۵۸۰۰۳

```
#include <exception>
```

اضافه کردن exception :

```
void Wood :: set_length (float l)
{
    if (l > 0)                //Check the length of the positive
    {
        length = l;
        return;
    }
    else
        throw invalid_argument ("Length should be positive");
}
```

```
void Wood :: set_height (float h)
{
    if (h > 0)                //Check the height of the positive
    {
        height = h;
        return;
    }
    else
        throw invalid_argument ("Height should be positive");
}
```

یک سری از توابع که خروجی bool داشتند و true و false برمی گردانند، خروجی آن‌ها void شده و یک exception ، throw می کنند. به این صورت که اگر عدد وارد شده مثبت بود برنامه به درستی کار می کند در غیر این صورت ارور داده می شود.

```
vector <Wood> addWood (vector <Wood> & w)
```

در برنامه یک تابع برای اضافه کردن چوب تعریف شده است.

این تابع طول و عرض و ارتفاع و نوع چوب را از کاربر می گیرد و اگر اطلاعات وارد شده توسط کاربر درست بودند اطلاعات چوب به انتهای وکتور اضافه می شود.

```
void addTool (vector <Tool *> & t)
```

یک تابع دیگر برای اضافه کردن ابزار تعریف شده است.

در این برنامه دو ابزار دریل و اره وجود دارد برای اضافه کردن ابزار ابتدا برنامه از کاربر می پرسد که اره می خواهد یا دریل سپس اطلاعات دیگر نظیر سایز و نوع و .. پرسیده می شود و به انتهای وکتور اضافه می شود.

```
if (typeTool == 0 || typeTool == 1)
{
    s->set_type(typeTool);
}
```

از اونجایی که برای نوع ابزار تنها می‌توان دو عدد ۰ و ۱ را وارد کرد در صورت مشاهده هر عدد دیگر

```
else
{
    throw invalid_argument ("You should enter 0 or 1");
}
```

یک `exception` `throw` می‌شود.

برای سباز ابزار هم همچنین چیزی نوشته شده است.

```
void show (vector <Wood> w)
{
    for (size_t i = 0; i < w.size(); i++)
    {
        cout << w[i].type << "\t" << w[i].length << "\t" << w[i].width
<< endl;
    }
}
```

همچنین دو تابع `show` در برنامه نوشته شده است که اطلاعات تمام چوب‌های داخل وکتور را نمایش می‌دهد. (برای ابزار هم تابع `show` نوشته شده است).

```
void Menu ()
{
    puts(" ");
    cout << "To add wood      [wood]" << endl;
    cout << "To add tool      [tool]" << endl;
    cout << "To exit          [exit]" << endl;
    cout << "Enter a command : ";
}
```

کاربر با وارد کردن گزینه‌های موجود در منوی برنامه به بخش مورد نظر می‌رود و در صورت وارد کردن `exit` با تابع `exit(0)` مواجه شده و برنامه خاتمه می‌یابد.

