

برنامه سازی پیشرفته

گزارشکار فاز دوم پروژه

هانیه اکبری

۹۹۱۲۳۵۸۰۰۳

```
class Saw : public Tool
{
    public :
        int set_number_of_times_used();
        int get_number_of_times_used();
    private :
        int number_of_times_used;
};
```

در فاز دوم پروژه کلاس Saw اضافه شده است که از کلاس ابزار ارث می‌برد.

یک متغیر متفاوت از متغیرهای موجود در کلاس ابزار برای آن تعریف شده است که برای تعداد دفعات استفاده از این ابزار می‌باشد.

```
class Drill : public Tool
{
    public :
        bool set_condition(int c);
        int get_condition() const;
        int set_number_of_times_used();
        int get_number_of_times_used();
    private :
        int condition;           //on and off key (0 for off, 1 for on)
        int number_of_times_used;
};
```

کلاس Drill هم از کلاس ابزار ارث برده است و متغیرهای condition و number_of_times_used برای آن تعریف شده است. اولی برای تعیین خاموش یا روشن بودن دریل و دومی برای تعداد دفعات استفاده شده از دریل.

```
virtual void print() const;
```

همچنین یک تابع print در کلاس ابزار اضافه شده است.

```
void Tool :: print() const
{
    cout << "Name tool : " << get_name() << endl;
    cout << "Type tool : " << get_type() << endl;
    cout << "Size tool : " << get_size() << endl;
}
```

این تابع برای تمامی کلاس‌هایی که از کلاس ابزار ارث برده‌اند قابل دسترس است.

```
bool Drill :: set_condition(int c)
{
    if (c == 0)
    {
        return false;
    }

    if (c == 1)
    {
        return true;
    }
    return false;
}
```

تابع `set_condition` مربوط به کلاس `Drill` است. اگر `c` صفر باشد به معنی خاموش بودن دریل است و `false` برگردانده می‌شود و اگر `c` یک باشد به معنی روشن بودن دریل است و `true` برگردانده می‌شود.

```
int Drill :: get_condition() const
{
    return condition;
}
```

تابع `get_condition` روشن یا خاموش بودن دریل را برمی‌گرداند.

```
int Drill :: set_number_of_times_used()
{
    number_of_times_used = 0;
    return number_of_times_used ;
}
```

در ابتدا تعداد دفعات استفاده از ابزار صفر می‌باشد این تابع برای دریل هم تعریف شده است.

```
void operator+ (const int & i, Saw & s)
{
    if (s.get_size() == 3)
    {
        cout << "It is not possible to change the size" << endl;
        return;
    }
    s.set_size(s.get_size() + i);

    if (s.get_size() > 3)
    {
        cout << "It is not possible to change the size" << endl;
        return;
    }
}
```

عملگر + اورلود شده است. در این برنامه عملگر + به این منظور استفاده می‌شود که سائز ابزار را بزرگ یا کوچک کند. همچنین اگر سائز ابزار ۳ بود که بزرگترین سائز ممکن است خطا داده میشود و سائز تغییری نمی‌کند.

