



## مقدمه

برنامه‌نویسی عمومی (Generic Programming) رویکردی است که در آن برنامه فارغ از گونه نوشته می‌شود و به هنگام استفاده گونه آن مشخص می‌شود. در این تمرین نوشتن کلاس‌ها و توابع عمومی که با انواع متفاوتی از داده‌ها کار می‌کنند، مورد تمرین قرار می‌گیرد. علاوه بر این نیم نگاهی به مبحث اشاره‌گرها نیز خواهیم داشت.

## شرح تمرین

### پشته جادویی (Magical stack)

با ساختار داده پشته در درس مبانی و مباحث قبلی این درس آشنا شده‌اید.

در این تمرین به پیاده‌سازی پشته با استفاده از الگوها می‌پردازیم. پشته به صورت عمومی و فارغ از گونه خاصی پیاده‌سازی می‌شود. به عبارت دیگر، می‌توانیم پشته از هرگونه‌ای داشته باشیم که گونه موردنظر در هنگام استفاده مشخص می‌شود و همه توابع به طور مستقل از گونه پیاده‌سازی می‌شوند.

پشته جادویی باید کارکردهای زیر را پشتیبانی کند.

+ تابع سازنده (Constructor)

+ تابع مخرب (Destructor)

+ تابع ورود (Push)

+ تابع خروج (Pop)

+ گرفتن سر پشته (Top)

+ گرفتن تعداد المان‌های پشته (elem\_count)

+ گرفتن پیمایشگر

پشته جادویی می‌تواند در لحظه‌ای که می‌خواهیم اطلاعات خود را به ما بدهد. به این منظور در آن لحظه یک پیمایشگر برای پشته موردنظر ایجاد می‌کنیم و از اطلاعات آن پشته استفاده می‌کنیم. تابع آخر به این منظور مورد استفاده قرار می‌گیرد.

راهنمایی: کلاس پیمایشگر در داخل کلاس پشته و به صورت **public** تعریف می‌شود و سازنده آن **private** است اما کلاس پشته را به عنوان دوست (**friend**) خود در نظر می‌گیرد تا کلاس پشته بتواند سازنده‌ی آن را فراخوانی کند.

یک نمونه از **main** برای تست پشت مورد نظر در زیر آمده است.

```
int main() {
    stack<int> s(10);
    stack<string> a(20);
    s.push(4);
    s.push(400);
    a.push("ali");
    a.push("mamad");
    int sum = 0;
    stack<int>::Iterator it = s.get_iterator();
    while (it.has_more_elements()){
        sum += it.next_element();
    }
    cout << sum << endl;
}
```

## نحوه‌ی تحویل

فایل برنامه‌ی خود را با نام **HO14-SID.cpp** در صفحه‌ی **CECM** درس بارگذاری کنید. **SID** شماره‌ی دانشجویی شماست؛ برای مثال، اگر شماره‌ی دانشجویی شما **۸۱۰۱۱۲۳۴۵** است، نام فایل شما باید **HO14-810112345.cpp** باشد.