بسم الله الرحمن الرحیم

دانشگاه صنعتی شاهرود، دانشکده برق

درس برنامه نویسی پیشرفته

تمرین سری سوم – موعد تحویل ۲۷ اسفند

۱. می‌خواهیم نسخه‌ی توسعه یافته ای از کلاس CStats را بنویسیم. برای این منظور، متغیرهای عضو کلاس CStats تمرین دوم را به حالت protected تغییر داده و کلاسی با نامه CStatsEx ایجاد کنید که از کلاس CStats مشتق شود کلاس جدید باید شامل تغییرات زیر باشد:

- یک متغیر خصوصی جدید با عنوان name از نوع std::string اضافه کنید. این متغیر حاوی نامی برای هر شیء CStatsEx خواهد بود
- دو متغیر خصوصی دیگر با عناوین last_min و last_max اضافه کنید. اینها کمینه و بیشینه را در خود نگه خواهند داشت.
- تابع update را به گونه‌ای تغییر دهید که علاوه بر کاری که قبلا انجام می‌داد، مقادیر last_min و last_max را هم به روز رسانی کند. برای این کار نیاز نیست last_mean و last_std دوباره حساب شود، فقط کافیست ابتدای تابع update کلاس فرزند (CStatsEx) تابع update کلاس پدر (CStats) را فراخوانی کنید:

```
CStatsEx<_T, _SZ>::update()
{
	CStats<_T, _SZ>::update();// فراخوانی تابع کلاس پدر از داخل کلاس فرزند!
    //از اینجا به بعد برنامه محاسبه کمینه و بیشینه را بنویسید
}
```

- تابع print را هم طوری تغییر دهید که علاوه بر نمایش داده‌ها، نام شیء، کمینه و بیشینه‌ی داده‌های آن را هم نشان دهد.
- تابع سازنده‌ای به صورت زیر بسازید:

```
CStatsEx(float* input_data, int size, std::string name)
```

- تابع جدید با نام sort بنویسید که داده‌های کلاس را از کوچک به بزرگ مرتب کند.

برنامه ای بنویسید که تعدادی نمره (بین ۰ و ۲۰) از کاربر دریافت کرده و در اشاره گری ذخیره کند (تعداد از کاربر دریافت شود). سپس با استفاده از اشاره گر یک نمونه از کلاس **CStatsEx** بسازید (با استفاده از عملگر **new** و تابع سازنده دوم؛ نام شیء را students' scores بگذارید) و داده های دریافتی را به این کلاس ارسال کنید. در نهایت به ترتیب توابع **print**، **update** و **sort** و **print** را فراخوانی کنید.

۲. **بازی سنگ کاغذ قیچی!** با مراجعه به فایل rps.cpp، کلاسی به نام Tool پیاده سازی کنید. این کلاس باید یک متغیر int به نام strength و متغیر char به نام type داشته باشد. اینها می توانند خصوصی (private) یا محافظت شده (protected) باشند. کلاس Tool همچنین باید تابعی با الگوی void setStrength (int) داشته باشد که strength را تنظیم می کند.

سه کلاس دیگر به نامهای Rock، Paper و Scissors ایجاد کنید که از Tool ارث می برند. هر یک از این کلاس ها به یک سازنده با یک ورودی int نیاز دارند که مقدار strength را مقداردهی اولیه می کند. سازنده همچنین باید متغیر type را با 'r' برای سنگ، 'p' برای کاغذ و 's' برای قیچی مقداردهی اولیه کند.

این کلاسها همچنین به یک تابع عمومی با الگوی bool fight(Tool) نیاز دارند که مقدار strength آنها را به روش زیر مقایسه کند:

قدرت سنگ هنگام مبارزه با قیچی دو برابر می شود (به طور موقت) و در هنگام مبارزه با کاغذ نصف می شود (موقت).

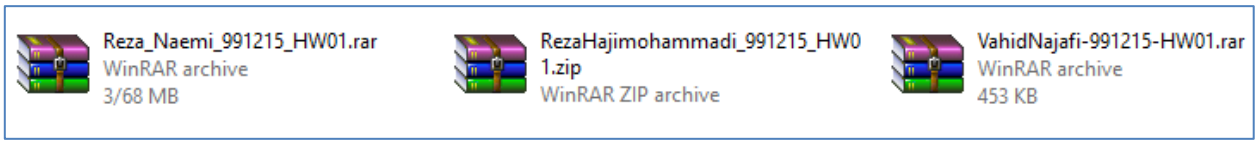
به همین ترتیب ، کاغذ در برابر سنگ و قیچی در برابر کاغذ برتری دارد.

تابع fight اگر کلاس اصلی برنده شود مقدار true وگرنه false برمی گرداند. پارامتر strength نباید در تابع مذکور تغییر کند.

شما همچنین می توانید در هر یک از این کلاس ها توابع اضافی و یا متغیرهای کمکی اضافی داشته باشید. برنامه main را بدون تغییر اجرا کنید و صحت نتایج را تأیید کنید.

نحوه ارسال تمرین

- **یک فایل word ایجاد کنید** و پس از ذکر نام و شماره دانشجویی و شماره تمرین، کد نوشته شده به همراه یک نمونه اجرای برنامه را در آن قرار دهید (از خروجی برنامه با فشردن همزمان دکمه‌های Alt و Print Screen عکس بگیرید و داخل فایل word کپی کنید).
- فایلهای حاوی کد و فایل پروژه (پسوندهای *.cpp, *.h, *.vcxproj) را به همراه فایل word، زیپ کنید. دقت کنید که پوشه های debug و release و فایل اجرایی برنامه (exe) و نیز فایلهای حجیم دیگر مثل *.db یا *.sdf و ... را **انتخاب نکنید**.
- عنوان فایل باید شامل **اسم خودتان**، **تاریخ ارسال** و **شماره تمرین** باشد مثلا VahidNajafi-991215-HW01.rar



- فایل زیپ شده را در سامانه LMS ارسال کنید.

نکته مهم: بخشی از نمره تمرین مربوط به تهیه فایل word به صورت خواسته شده، است.

تاخیر تا ۱ روز قابل اغماض است، تا ۵ روز، ۲۵٪ کسر نمره، بیش از ۵ روز، پذیرفته نخواهد شد

امام رضا علیه السلام

من فرج عن مؤمن فرج الله عن قلبه یوم القیامة

هر کس اندوه و مشکلی را از مؤمنی برطرف نماید ، خداوند در روز قیامت اندوه را از قلبش برطرف سازد. امام رضا (ع)

موفق باشید؛ حسین خسروی