

补充:

1、（单继承练习）在 14-b5 的 TString 类的基础上，定义 TStringAdv 类

(1) TStringAdv 继承 TString 类

(2) 增加下列函数

- assign 函数（赋新值，等价于=操作）

```
TStringAdv& assign(const TStringAdv &ts2);
```

```
TStringAdv& assign(const char *s);
```

```
TStringAdv& assign(const char &c);
```

- append 函数（在尾部追加，等价于+=操作）

```
TStringAdv& append(const TStringAdv &ts2);
```

```
TStringAdv& append(const char *s);
```

```
TStringAdv& append(const char &c);
```

- insert 函数（插入到 pos 开始的位置， $pos \in [1..len+1]$ ，非法则不插入，返回原值即可）

```
TStringAdv& insert(const TStringAdv &ts2, int pos);
```

```
TStringAdv& insert(const char *s, int pos);
```

```
TStringAdv& insert(const char &c, int pos);
```

- erase 函数（删除第一个匹配的子串，等价于-）

```
TStringAdv& erase(const TStringAdv &ts2);
```

```
TStringAdv& erase(const char *s);
```

```
TStringAdv& erase(const char &c);
```

- substr 函数（从 pos 开始，返回长度为 len 的子串， $pos \in [1..len]$ ，非法则返回空，len 可缺省，表示取到尾部为止）

```
TStringAdv substr(const int pos, const int len = **); 【**：缺省值该填什么？】
```

【注：】返回类型为 TStringAdv，不是引用!!!

- at 函数（返回指定位置字符的值，等价于[]操作）

```
char& at(const int n);
```

(3) 自己写测试用例 16-b1-main.cpp

- 将 14-b5-main.cpp 的内容复制进来，全部 TString 改为 TStringAdv，要能通过

- 为了提高调试效率，两个 100MB 累加可删去

- 在后面加入 (2) 中函数的测试用例（部分函数的测试应包含不正确条件）

- main 函数需要相互验证（甲的 main 函数提供给乙，验证乙的三个类），每人需要将 main 函数的源码提供给至少 5 人，将名单放在 main 函数的最开始，用注释说明即可（正常情况双向查验都应该正确，如果查验不正确则要连环扣分，即提供者检查者同步扣分）

(4) 本次作业由下列文件组成

16-b1-TString.h : 直接复制原 14-b5.h 即可（如因为之前有错需要修改，允许）

16-b1-TString.cpp : 直接复制原 14-b5.cpp 即可（如因为之前有错需要修改，允许）

16-b1-TStringAdv.h : TStringAdv 类的定义

16-b1-TStringAdv.cpp : TStringAdv 类的实现

16-b1-main.cpp : 测试用例

2、（多继承练习）按要求完成下列三个类的定义及实现

- (1) 定义 Date 类，有年月日三个数据成员，要求定义类的构造函数及 set 函数，重载 cin/cout/+/-/++/--/>/>=</<==/!=等运算符（构造及 set 在三个成员有任何错误时均设置为 1900-01-01，运算以天为单位，下限 1900-01-01 和上限 2099-12-31 互为+1 关系）
 - 原 14-b4 的要求中 set 约定 0 表示维持原值不变，缺省为 2000-01-01，本题认为 0 错误，默认设为 1900-01-01
 - Date 和整型的+/-，遵循互为+1 关系，即 $2099-12-31 + 5 = 1900-01-05$ / $1900-01-01 - 5 = 2099-12-27$
 - Date 和 Date 的-，值可为正负， $2099-12-31 - 1900-01-01$ 为正，反之为负，值为相差的天数
- (2) 定义 Time 类，有时分秒三个数据成员，要求定义类的构造函数及 set 函数，重载 cin/cout/+/-/++/--/>/>=</<==/!=等运算符（构造及 set 在三个成员有任何错误时均设置为 00:00:00，运算以秒为单位，下限 00:00:00 和上限 23:59:59 互为+1 关系）
 - Time 和整型的+/-，遵循互为+1 关系，即 $23:59:59 + 5 = 00:00:04$ / $00:00:01 - 5 = 23:59:56$
 - Time 和 Time 的-，值可为正负， $14:15:23 - 12:13:14$ 为正，反之为负，值为相差的秒数
- (3) 定义 DateTime 类，表示为 yyyy-mm-dd hh:mm:ss 形式，要求从 Date 类和 Time 类中继承这 6 个数据成员（即 DateTime 类中不能再重复定义），定义类的构造函数及 set 函数，重载 cin/cout/+/-/++/--/>/>=</<==/!=等运算符（构造及 set 在三个成员有任何错误时均设置为 1900-01-01 00:00:00，运算以秒为单位，下限 1900-01-01 00:00:00 和上限 2099-12-31 23:59:59 互为+1 关系）
 - 数据范围问题，DateTime 类和整型的转换，表示为 long long 型（Linux 中虽然用 long 即可，但为了统一，多编译器均要求采用 long long）
 - long long 和其它整型数据的转换及计算，按系统定义规则即可
 - DateTime 和整型的+/-，遵循互为+1 关系，即 $2099-12-31\ 23:59:59 + 5 = 1900-01-01\ 00:00:04$ / $1900-01-01\ 00:00:04 - 5 = 2099-12-31\ 23:59:59$
 - DateTime 和 DateTime 的-，值可为正负， $2021-05-15\ 17:18:19 - 2021-05-14\ 14:15:16$ 为正，反之为负，值为相差的秒数

【基本要求：】

- (1) DateTime 类的实现过程中，遵循两个基本原则，一是尽量从其两个父类中继承函数而少定义自己的成员函数；二是自己的成员函数实现时尽可能调用父类的成员函数
- (2) 构造 main 函数里面的测试用例，要求对 DateTime 类进行测试（要求考虑测试项的合理性及其可检出差错的范围，main 函数的构造占本题分数的 35%）
- (3) main 函数需要相互验证（甲的 main 函数提供给乙，验证乙的三个类），每人需要将 main 函数的源码提供给至少 5 人，将名单放在 main 函数的最开始，用注释说明即可（正常情况双向查验都应该正确，如果查验不正确则要连环扣分，即提供者检查者同步扣分）
- (4) 本次作业由下列文件组成（头文件名以此为准），按要求编写各自的内容（每个类，h 中存放定义，cpp 中给出体外形式的实现，main 中放测试用例），提交时依次提交即可

16-b2-date.h	: Date 类的定义
16-b2-date.cpp	: Date 类的实现
16-b2-time.h	: Time 类的定义
16-b2-time.cpp	: Time 类的实现
16-b2-datetime.h	: DateTime 类的定义
16-b2-datetime.cpp	: DateTime 类的实现
16-b2-main.cpp	: main 函数，用于测试，三个类的测试用例都要有

【编译器要求:】

		编译器VS	编译器Dev	编译器Linux
16-b1-TString.h	TStringAdv类-基类.h	Y	Y	Y
16-b1-TString.cpp	TStringAdv类-基类.cpp	Y	Y	Y
16-b1-TStringAdv.h	TStringAdv类-派生类.h	Y	Y	Y
16-b1-TStringAdv.cpp	TStringAdv类-派生类.cpp	Y	Y	Y
16-b1-main.cpp	TStringAdv类-main.cpp	Y	Y	Y
16-b2-date.h	DateTime类-date.h	Y	Y	Y
16-b2-date.cpp	DateTime类-date.cpp	Y	Y	Y
16-b2-time.h	DateTime类-time.h	Y	Y	Y
16-b2-time.cpp	DateTime类-time.cpp	Y	Y	Y
16-b2-datetime.h	DateTime类-datetime.h	Y	Y	Y
16-b2-datetime.cpp	DateTime类-datetime.cpp	Y	Y	Y
16-b2-main.cpp	DateTime类-main.cpp	Y	Y	Y

【作业要求:】

- 1、**5月20日前**网上提交本次作业
- 2、每题所占平时成绩的具体分值见网页
- 3、超过截止时间提交作业会自动扣除相应的分数，具体见网页上的说明