第1关：（\*）圆

100

* 任务要求
* 评论
* [任务描述](https://www.educoder.net/tasks/laqm4gt32nfx?coursesId=14604#%E4%BB%BB%E5%8A%A1%E6%8F%8F%E8%BF%B0)
* [编程要求](https://www.educoder.net/tasks/laqm4gt32nfx?coursesId=14604#%E7%BC%96%E7%A8%8B%E8%A6%81%E6%B1%82)
* [测试说明](https://www.educoder.net/tasks/laqm4gt32nfx?coursesId=14604#%E6%B5%8B%E8%AF%95%E8%AF%B4%E6%98%8E)

**任务描述**

本关任务：定义圆的类Circle，包含三个属性：圆心(x,y)和半径r

**编程要求**

根据提示，在右侧编辑器补充代码，定义圆的类Circle，包含三个属性：圆心(x,y)和半径r，成员函数见main函数。用户先输入整数x, y, r，表示圆心（x,y）和半径r，程序输出圆的圆心坐标，以及圆的半径。用户再输入整数move\_x, move\_y，表示圆心的偏移量。程序输出移动后的圆的圆心坐标。用户最后输入整数new\_r，表示新的半径，程序输出修改后的圆的半径。

**测试说明**

平台会对你编写的代码进行测试：

测试输入： 2 3 5 -1 1 10

预期输出： 2 3 5 1 4 10

开始你的任务吧，祝你成功！

第2关：(\*)银行账户

100

* 任务要求
* 评论
* [任务描述](https://www.educoder.net/tasks/bo6f7rz2pcfk?coursesId=14604#%E4%BB%BB%E5%8A%A1%E6%8F%8F%E8%BF%B0)
* [编程要求](https://www.educoder.net/tasks/bo6f7rz2pcfk?coursesId=14604#%E7%BC%96%E7%A8%8B%E8%A6%81%E6%B1%82)
* [测试说明](https://www.educoder.net/tasks/bo6f7rz2pcfk?coursesId=14604#%E6%B5%8B%E8%AF%95%E8%AF%B4%E6%98%8E)

**任务描述**

本关任务：定义账户类SavingAccount，包含账号，存款金额和月利率。

**编程要求**

根据提示，在右侧编辑器补充代码，要求账号自动生成，第一个生成的对象账号为1，第二个生成的对象账号为2，依此类推。所需的操作包括修改月利率，每月计算新的存款额（原金额加上本月利息）和显示账户金额。不得使用全局变量。月利率初始值为0.05。本题已给定main()函数，输入共三次：第一次是第一个账户和第二个账户的存款金额；第二次是第二个月以后新的月利率；第三次是第三个账户的存款金额。输出是四个月的所有账户的账号、存款金额和月利率。(存款金额四舍五入保留两位小数)

**测试说明**

平台会对你编写的代码进行测试：

测试输入： 20000 10000 0.01 30000

预期输出： 1 20000.00 0.05 2 10000.00 0.05 1 21000.00 0.05 2 10500.00 0.05 1 21210.00 0.01 2 10605.00 0.01 1 21422.10 0.01 2 10711.05 0.01 3 31500.00 0.05

开始你的任务吧，祝你成功！

第3关：LongLongInt

100

* 任务要求
* 评论
* [任务描述](https://www.educoder.net/tasks/tfrcb54gfho8?coursesId=14604#%E4%BB%BB%E5%8A%A1%E6%8F%8F%E8%BF%B0)
* [编程要求](https://www.educoder.net/tasks/tfrcb54gfho8?coursesId=14604#%E7%BC%96%E7%A8%8B%E8%A6%81%E6%B1%82)
* [测试说明](https://www.educoder.net/tasks/tfrcb54gfho8?coursesId=14604#%E6%B5%8B%E8%AF%95%E8%AF%B4%E6%98%8E)

**任务描述**

本关任务：定义可处理任意大的正整数类LongLongInt

**编程要求**

根据提示，在右侧编辑器补充代码，定义可处理任意大的正整数类LongLongInt，用一个动态字符数组存放任意长度的正整数，数组的每个元素存放整型数的一位。用户输入两个任意长度的正整数，输出这两个数的和。

**测试说明**

平台会对你编写的代码进行测试：

测试输入： 9934 567

预期输出： 10501

开始你的任务吧，祝你成功！

第4关：(\*)时钟

100

* 任务要求
* 评论
* [任务描述](https://www.educoder.net/tasks/cag9yxr64bp7?coursesId=14604#%E4%BB%BB%E5%8A%A1%E6%8F%8F%E8%BF%B0)
* [编程要求](https://www.educoder.net/tasks/cag9yxr64bp7?coursesId=14604#%E7%BC%96%E7%A8%8B%E8%A6%81%E6%B1%82)
* [测试说明](https://www.educoder.net/tasks/cag9yxr64bp7?coursesId=14604#%E6%B5%8B%E8%AF%95%E8%AF%B4%E6%98%8E)

**任务描述**

本关任务：设计一个hh:mm:ss格式的时钟类，支持时间的修改和计算两个时间的差值(后者减前者的绝对值)

**编程要求**

根据提示，在右侧编辑器补充代码，完成时钟类的定义。用户输入两个时间（24小时制），计算其差值（两个时间不保证先后顺序）

**测试说明**

平台会对你编写的代码进行测试：

测试输入： 10:30:25 14:20:34

预期输出： 03:50:09

开始你的任务吧，祝你成功！

第5关：(\*)栈

100

* 任务要求
* 评论
* [任务描述](https://www.educoder.net/tasks/mifo9uawxyv8?coursesId=14604#%E4%BB%BB%E5%8A%A1%E6%8F%8F%E8%BF%B0)
* [编程要求](https://www.educoder.net/tasks/mifo9uawxyv8?coursesId=14604#%E7%BC%96%E7%A8%8B%E8%A6%81%E6%B1%82)
* [测试说明](https://www.educoder.net/tasks/mifo9uawxyv8?coursesId=14604#%E6%B5%8B%E8%AF%95%E8%AF%B4%E6%98%8E)

**任务描述**

本关任务：栈是一种只能在一端进行插入和删除操作的数据结构，按照先进后出的原则存储数据，先进入的数据被压入栈底，最后的数据在栈顶，需要读数据的时候从栈顶开始弹出数据（最后一个数据被第一个读出来）。定义栈类mystack，用一个动态整型数组存放栈的数据，数据成员包括指向动态数组的指针，栈的最大规模（缺省值为100）和栈顶指针，要求能够判别栈满和栈空，数据进栈函数push，出栈函数pop等。

**编程要求**

根据提示，在右侧编辑器补充代码，main()函数已隐藏以避免被修改，内容如下：

int main() { int num[]={1,3,5,7,9,11},i,data; mystack s(5); cout<<"The stack is empty: "<<s.isempty()<<endl; cout<<"The stack is full: "<<s.isfull()<<endl; for(i=0;i<=5;i++) { if(s.push(num[i])) cout<<"Push "<<num[i]<<" success!"<<endl; else cout<<"Push "<<num[i]<<" fail!"<<endl; } cout<<"The stack is empty: "<<s.isempty()<<endl; cout<<"The stack is full: "<<s.isfull()<<endl; for(i=0;i<=5;i++) { if(s.pop(data)) cout<<"Pop "<<data<<" success!"<<endl; else cout<<"Pop fail!"<<endl; } return 0; } 请从给出的main()函数推断出要实现的功能和函数。给出的main()函数不一定代表真实的测试程序的方式。

**测试说明**

平台会对你编写的代码进行测试： 对应于所给main函数，其输出为： The stack is empty: 1 The stack is full: 0 Push 1 success! Push 3 success! Push 5 success! Push 7 success! Push 9 success! Push 11 fail! The stack is empty: 0 The stack is full: 1 Pop 9 success! Pop 7 success! Pop 5 success! Pop 3 success! Pop 1 success! Pop fail!

开始你的任务吧，祝你成功

第4关：(\*)时钟

100

* 任务要求
* 评论
* [任务描述](https://www.educoder.net/tasks/cag9yxr64bp7?coursesId=14604#%E4%BB%BB%E5%8A%A1%E6%8F%8F%E8%BF%B0)
* [编程要求](https://www.educoder.net/tasks/cag9yxr64bp7?coursesId=14604#%E7%BC%96%E7%A8%8B%E8%A6%81%E6%B1%82)
* [测试说明](https://www.educoder.net/tasks/cag9yxr64bp7?coursesId=14604#%E6%B5%8B%E8%AF%95%E8%AF%B4%E6%98%8E)

**任务描述**

本关任务：设计一个hh:mm:ss格式的时钟类，支持时间的修改和计算两个时间的差值(后者减前者的绝对值)

**编程要求**

根据提示，在右侧编辑器补充代码，完成时钟类的定义。用户输入两个时间（24小时制），计算其差值（两个时间不保证先后顺序）

**测试说明**

平台会对你编写的代码进行测试：

测试输入： 10:30:25 14:20:34

预期输出： 03:50:09

开始你的任务吧，祝你成功！

第5关：(\*)栈

100

* 任务要求
* 评论
* [任务描述](https://www.educoder.net/tasks/mifo9uawxyv8?coursesId=14604#%E4%BB%BB%E5%8A%A1%E6%8F%8F%E8%BF%B0)
* [编程要求](https://www.educoder.net/tasks/mifo9uawxyv8?coursesId=14604#%E7%BC%96%E7%A8%8B%E8%A6%81%E6%B1%82)
* [测试说明](https://www.educoder.net/tasks/mifo9uawxyv8?coursesId=14604#%E6%B5%8B%E8%AF%95%E8%AF%B4%E6%98%8E)

**任务描述**

本关任务：栈是一种只能在一端进行插入和删除操作的数据结构，按照先进后出的原则存储数据，先进入的数据被压入栈底，最后的数据在栈顶，需要读数据的时候从栈顶开始弹出数据（最后一个数据被第一个读出来）。定义栈类mystack，用一个动态整型数组存放栈的数据，数据成员包括指向动态数组的指针，栈的最大规模（缺省值为100）和栈顶指针，要求能够判别栈满和栈空，数据进栈函数push，出栈函数pop等。

**编程要求**

根据提示，在右侧编辑器补充代码，main()函数已隐藏以避免被修改，内容如下：

int main() { int num[]={1,3,5,7,9,11},i,data; mystack s(5); cout<<"The stack is empty: "<<s.isempty()<<endl; cout<<"The stack is full: "<<s.isfull()<<endl; for(i=0;i<=5;i++) { if(s.push(num[i])) cout<<"Push "<<num[i]<<" success!"<<endl; else cout<<"Push "<<num[i]<<" fail!"<<endl; } cout<<"The stack is empty: "<<s.isempty()<<endl; cout<<"The stack is full: "<<s.isfull()<<endl; for(i=0;i<=5;i++) { if(s.pop(data)) cout<<"Pop "<<data<<" success!"<<endl; else cout<<"Pop fail!"<<endl; } return 0; } 请从给出的main()函数推断出要实现的功能和函数。给出的main()函数不一定代表真实的测试程序的方式。

**测试说明**

平台会对你编写的代码进行测试： 对应于所给main函数，其输出为： The stack is empty: 1 The stack is full: 0 Push 1 success! Push 3 success! Push 5 success! Push 7 success! Push 9 success! Push 11 fail! The stack is empty: 0 The stack is full: 1 Pop 9 success! Pop 7 success! Pop 5 success! Pop 3 success! Pop 1 success! Pop fail!

开始你的任务吧，祝你成功