

第一题：链表合并（A卷）

题目 | #969

题目描述

给定一个单链表，请将奇数编号且值为奇数的节点和偶数编号且值为偶数的节点分别聚集，并输出合并后的新链表。要求必须使用链表结构实现，并通过运算符重载完成输入输出。

- 1. 节点编号规则：首个节点视为奇数节点（编号1），第二个节点视为偶数节点（编号2），以此类推
- 2. 转换输出：奇数链表在前，偶数链表在后，输出合并之后的结果链表，奇数节点组和偶数节点组需保持原始相对顺序
- 3. 特殊处理：链表为空时输出"NULL"

实现要求

- 1. 必须自定义链表结构，包含节点类Node和链表类LinkedList
- 2. 必须重载>>运算符实现链表输入，重载<<运算符实现链表输出
- 3. 必须实现rearrange()方法完成奇偶节点重排

输入

格式为：

M A1 A2 ... AM

(M为节点个数，A1~AM为各节点整数，M=0时表示空链表)

输出

输出重组后的链表，节点间用空格分隔，末尾无空格。

空链表输出"NULL"。

样例输入

7 1 31 15 27 9 10 12

样例输出

1 15 9 10

题目配置 »

题目名：第一题：链表合并（A卷）

编号：969

测试点：10

时间限制：1000 ms

空间限制：65536 KiB

完成状态：已通过

通过率：155 / 1348

评测全部测试点：是

Special Judge：未启用

快速跳转 »

A.第一题：链表合并（A卷）

提交题目

提交记录

第一题：链表合并（B卷）

题目 | #970

题目描述

给定一个单链表，请将奇数编号且值为奇数的节点和偶数编号且值为偶数的节点分别聚集，并输出合并后的新链表。要求必须使用链表结构实现，并通过运算符重载完成输入输出。

- 1. 节点编号规则：首个节点视为奇数节点（编号1），第二个节点视为偶数节点（编号2），以此类推
- 2. 转换输出：偶数链表在前，奇数链表在后，输出合并之后的结果链表，奇数节点组和偶数节点组需保持原始相对顺序
- 3. 特殊处理：链表为空时输出"NULL"

实现要求

- 1. 必须自定义链表结构，包含节点类Node和链表类LinkedList
- 2. 必须重载>>运算符实现链表输入，重载<<运算符实现链表输出
- 3. 必须实现rearrange()方法完成奇偶节点重排

输入

格式为：

M A1 A2 ... AM

(M为节点个数，A1~AM为各节点整数，M=0时表示空链表)

输出

输出重组后的链表，节点间用空格分隔，末尾无空格。

空链表输出"NULL"。

样例输入

7 1 31 15 27 9 10 12

样例输出

10 1 15 9

题目配置 »

题目名：第一题：链表合并（B卷）

编号：970

测试点：10

时间限制：1000 ms

空间限制：65536 KiB

完成状态：未提交

通过率：111 / 1250

评测全部测试点：是

Special Judge：未启用

快速跳转 »

B.第一题：链表合并（B卷）

提交题目

提交记录

第二题：交通工具租赁系统（A卷）

题目 | #971

题目描述：

设计一个交通工具租赁系统，该系统能够管理不同类型交通工具的租赁信息和费用计算。系统需要支持以下功能：

一、基类设计：创建一个名为Vehicle的抽象基类，包含以下成员：

1、纯虚函数double calculateRentalFee(int days)：计算租赁费用、保护成员变量：brand（品牌，string类型）、model（型号，string类型）；

2、构造函数和虚析构函数；

3、纯虚函数void displayInfo（）用于输出每种交通工具的信息。

二、派生类实现：从Vehicle派生出以下具体类：

1、Car（汽车）：包含座位数属性（seats，int型），租赁费用为基础费用300元/天+座位数×10元/天；

2、Motorcycle（摩托车）：包含排量属性（cc，int型），租赁费用为排量×0.5元/天；

3、Bicycle（自行车）：包含是否电动属性（isElectric，int型，0表示非电动，1表示电动），电动自行车租赁费用为50元/天，普通自行车为20元/天。

请根据题目要求，计算不同种类交通工具的租赁费用。

输入：

第一行一个整数N，代表有N行交通工具的信息，接下来的N行，每行有5个数据，分别是接通工具的种类（C代表汽车，M代表摩托车，B代表自行车）、该种交通工具的品牌、型号、私有属性、租赁天数（int型），中间用空格分隔。

输出：

N行，按照题目要求调用函数displayInfo输出每个交通工具的种类（Car、Motorcycle、Bicycle）、品牌、型号，以及私有属性信息（只有是Bicycle时才输出），其中是否为电动属性，是输出true，否则输出false，调用calculateRentalFee函数输出租赁费用，各个数据之间用空格分隔，行位无空格。

样例输入：

3
C byd qin 5 2
M haojue AFR125X 150 1
B Giant Quick-E 0 3

样例输出：

Car byd qin 700
Motorcycle haojue AFR125X 75
Bicycle Giant Quick-E false 60

要求：必须实现继承和多态，必须实现抽象基类和动态联编，否则计0分。

题目配置

题目名：第二题：交通工具租赁系统（A卷）

编号：971

测试点：10

时间限制：1000 ms

空间限制：65536 KiB

完成状态：已通过

通过率：220 / 1686

评测全部测试点：是

Special Judge：未启用

快速跳转

E.第二题：交通工具租赁系统（A卷）

提交题目

提交记录

第二题：交通工具租赁系统（B卷）

题目 | #972

题目描述：

设计一个交通工具租赁系统，该系统能够管理不同类型交通工具的租赁信息和费用计算。系统需要支持以下功能：

一、基类设计：创建一个名为Vehicle的抽象基类，包含以下成员：

1、纯虚函数double calculateRentalFee(int days)：计算租赁费用、保护成员变量：brand（品牌，string类型）、model（型号，string类型）；

2、构造函数和虚析构函数；

3、纯虚函数void displayInfo（）用于输出每种交通工具的信息。

二、派生类实现：从Vehicle派生出以下具体类：

1、Car（汽车）：包含座位数属性（seats，int型），租赁费用为基础费用200元/天+座位数×15元/天；

2、Motorcycle（摩托车）：包含排量属性（cc，int型），租赁费用为排量×2元/天；

3、Bicycle（自行车）：包含是否电动属性（isElectric，int型，0表示非电动，1表示电动），电动自行车租赁费用为40元/天，普通自行车为20元/天。

请根据题目要求，计算不同种类交通工具的租赁费用。

输入：

第一行一个整数N，代表有N行交通工具的信息，接下来的N行，每行有5个数据，分别是接通工具的种类（C代表汽车，M代表摩托车，B代表自行车）、该种交通工具的品牌、型号、私有属性、租赁天数（int型），中间用空格分隔。

输出：

N行，按照题目要求调用函数displayInfo输出每个交通工具的种类（Car、Motorcycle、Bicycle）、品牌、型号，以及私有属性信息（只有是Bicycle时才输出），其中是否为电动属性，是输出True，否则输出False，调用calculateRentalFee函数输出租赁费用，各个数据之间用空格分隔，行位无空格。

样例输入：

3
C byd qin 5 2
M haojue AFR125X 150 1
B Giant Quick-E 0 3

样例输出：

Car byd qin 550
Motorcycle haojue AFR125X 300
Bicycle Giant Quick-E False 60

要求：必须实现继承和多态，必须实现抽象基类和动态联编，否则计0分。

题目配置

题目名：第二题：交通工具租赁系统（B卷）

编号：972

测试点：10

时间限制：1000 ms

空间限制：65536 KiB

完成状态：未提交

通过率：253 / 1624

评测全部测试点：是

Special Judge：未启用

快速跳转

F.第二题：交通工具租赁系统（B卷）

提交题目

提交记录

第三题：容器模板类（A卷）

题目 | #973

题目描述：

请编写实现一个容器模板类MyContainer，该模板类实现了一个能够存储常见数据类型（int,float和char）的基本容器，该模板类的主要形式和成员函数如下：

```
template<class T>
class MyContainer{
...
MyContainer(T array[], int len);           //构造函数，根据数组构造
void add( const &T value);                 //添加元素到容器末尾
bool remove(int idx);                     //删除指定位置的元素，删除成功返回true，位置越界返回false
bool empty();                             //空容器判断
int size();                               //返回当前容器大小
void display();                           //打印容器，打印容器中所有元素，格式：[元素1 元素2 ...]
void clear();                             //清空容器
};
```

完成该模板类及成员函数之后，请在main()函数中验证，验证要求和方式如下：

1、定义三个数组，整数类型数组、浮点数数据类型和字符类型数组，分别初始化为固定数值，然后利用这两个数组进行构造类对象，数组如下格式：

```
int iarray[] = {0,1,1,2,3,5,8,13,21,34,55,89}; //整数数组，初始数据固定
float farray[]={0,0.125,0.25,0.375,0.5,0.625,0.75,0.875,1.0}; //浮点数数组，初始为固定值
char carray[] = "I_love_Nankai_University"; //字符数组，初始为固定值
```

2、从输入端读取操作标记符和操作数，操作符和操作标记如下（第一个字符为操作标记符，后面的x为操作数）：

| 输入操作要求 | 说明： |
|-------------------|--|
| d 打印容器信息 | 执行display()方法打印容器数据，容器为空输出“None” |
| s 终止操作此容器 | 无输出信息 |
| i x 把操作数追加到容器末尾 | 无输出信息 |
| o x 从容器中删除位置为x的元素 | 无直接输出信息，若处理到空容器则输出“None”，若x越界输出“BoundsLimit” |
| c 清空当前容器 | 无输出信息 |

程序输入：

程序输入为两行，第一行为对int类型容器的操作序列，第二行为对float类型容器的操作序列，第三行为对字符类型容器操作的序列（字符类型操作序列中的操作数不会出现空格字符' '），每一行均以s结尾。

程序输出：

依次为输入操作码对应的输出信息。

样例输入：

```
d o 4 d s
i 1.25 d o 13 s
d i s d i d d c o 27 s
```

样例输出：

```
[0,1,1,2,3,5,8,13,21,34,55,89]
[0,1,1,2,5,8,13,21,34,55,89]
[0,0.125,0.25,0.375,0.5,0.625,0.75,0.875,1,1.25]
BoundsLimit
[I,_1,o,v,e,_N,a,n,k,a,i,_U,n,i,v,e,r,s,i,t,y]
[I,_1,o,v,e,_N,a,n,k,a,i,_U,n,i,v,e,r,s,i,t,y,s]
[I,_1,o,v,e,_N,a,n,k,a,i,_U,n,i,v,e,r,s,i,t,y,s,d]
None
```

题目配置 »

题目名：第三题：容器模板类（A卷）

编号：973

测试点：10

时间限制：1000 ms

空间限制：65536 KiB

完成状态：已通过

通过率：64 / 1352

评测全部测试点：是

Special Judge：未启用

快速跳转 »

C.第三题：容器模板类（A卷）

提交题目

提交记录

第三题：容器模板类（B卷）

题目 | #974

题目描述：

请编写实现一个容器模板类MyContainer，该模板类实现了一个能够存储常见数据类型（int,float和char）的基本容器，该模板类的主要形式和成员函数如下：

```
template<class T>
class MyContainer{
...
MyContainer(T array[], int len);           //构造函数，根据数组构造
void append( const &T value);              //追加元素到容器末尾
void insert(const T& value, int idx);       //插入元素value到指定的位置idx，idx为位置编号，容器的位置编号从0开始
bool empty();                             //空容器判断
int size();                               //返回当前容器大小
void display();                           //打印容器，打印容器中所有元素，格式：[元素1 元素2 ...]
void clear();                             //清空容器
};
```

完成该模板类及成员函数之后，请在main()函数中验证，验证要求和方式如下：

1、定义三个数组，整数类型数组、浮点数数据类型和字符类型数组，分别初始化为固定数值，然后利用这两个数组进行构造类对象，数组如下格式：

```
int iarray[] = {0,1,1,2,3,5,8,13,21,34,55,89}; //整数数组，初始数据固定
float farray[]={0,0.125,0.25,0.375,0.5,0.625,0.75,0.875,1.0}; //浮点数数组，初始为固定值
char carray[] = "I_love_Nankai_University"; //字符数组，初始为固定值
```

2、从输入端读取操作标记符和操作数，操作符和操作标记如下（第一个字符为操作标记符，后面的x为操作数）：

| 输入 操作要求 | 说明 |
|--------------------------|---------------------------------|
| d 打印容器信息 | 执行show()方法打印容器数据，容器为空输出“None” |
| s 终止操作此容器 | 无输出信息 |
| a x 把操作数x追加到容器末尾 | 无输出信息 |
| i x idx 把操作数x追加到容器的idx位置 | 无输出信息，若idx位置越界，则输出“BoundsLimit” |
| c 清空当前容器 | 无输出信息 |

程序输入：

题目配置 »

题目名：第三题：容器模板类（B卷）

编号：974

测试点：10

时间限制：1000 ms

空间限制：65536 KiB

完成状态：未提交

通过率：75 / 1004

评测全部测试点：是

Special Judge：未启用

快速跳转 »

D.第三题：容器模板类（B卷）

提交题目

提交记录

程序输入：

程序输入为三行，第一行为对int类型容器的操作序列，第二行为对float类型容器的操作序列，第三行为对字符类型容器操作的序列（字符类型操作序列中的操作数不会出现空格字符' '），每一行均以s结尾。

程序输出：

依次为输入操作码对应的输出信息。

样例输入：

```
d a 4 d c d s
i 1.25 9 d i 0.1 0 d s
i ! 28 d s
```

样例输出：

```
[0,1,1,2,3,5,8,13,21,34,55,89]
[0,1,1,2,3,5,8,13,21,34,55,89,4]
None
[0,0.125,0.25,0.375,0.5,0.625,0.75,0.875,1,1.25]
[0.1,0,0.125,0.25,0.375,0.5,0.625,0.75,0.875,1,1.25]
BoundsLimit
[I,_,l,o,v,e,_,N,a,n,k,a,i,_,U,n,i,v,e,r,s,i,t,y]
```

附加题：Knight of Nights

题目 | #975

题目描述

作为拥有操纵时间程度的能力的女仆，十六夜咲夜可以将大量飞刀丢出后暂停时间，从而形成一个完美的飞刀阵。

在坐标轴上，飞刀的轨迹可以看作是无限多等间距的、与y轴平行的直线，而飞刀阵则是一个简单多边形（可能是凹的）。

咲夜想在阵中布满飞刀，于是她想知道：在飞刀阵的圈定范围内，所有飞刀的轨迹的长度之和是多少。

为防止飞刀出现在阵的边缘导致攻击无效，保证多边形的任意一条边均不与y轴平行。同时为方便起见，总有一条轨迹直线恰好与y轴重合。

输入

第一行输入一个数字 n ，代表多边形的边数。

接下来的 n 行，每行用空格隔开两个实数，按照顺时针顺序给出多边形的 $n(3 \leq n \leq 10^5)$ 个顶点的坐标 $x_i, y_i (-10^4 \leq x_i, y_i \leq 10^4)$ ：第 i 个点与第 $i + 1$ 个点连接 ($1 \leq i \leq n$)，第 n 个点与第1个点连接。

最后一行给出轨迹的间距 $d(0.1 \leq d \leq 100)$ 。

输入数据的浮点数，均保留4位小数。

输出

对于每个样例输出一行，包含一个数字，表示多边形内包含的线段长度和。

若你的答案 ans 的与标准答案 std 满足： $\frac{|ans - std|}{\max(1.0, std)} \leq 10^{-6}$ ，则被认为正确，因此请保留较多的小数位。

样例输入1

```
4
2.0000 2.0000
2.5000 -2.0000
-2.5000 -2.0000
-2.0000 2.0000
1.0000
```

样例输入2

```
10
3.1309 5.0818
9.7876 -0.7907
1.4695 -8.4136
1.3539 -4.7559
-4.5584 -3.3204
-1.3199 0.5740
-9.1664 -6.0000
-5.7476 -0.2324
-8.1701 0.4321
-2.1978 4.3864
1.4741
```

样例输出2

```
85.2737861924
```

题目配置 »

题目名：附加题：Knight of Nights

编号：975

测试点：5

时间限制：1000 ms

空间限制：256000 KiB

完成状态：未提交

通过率：2 / 77

评测全部测试点：否

Special Judge：比较模式

快速跳转 »

G.附加题：Knight of Nights

提交题目

提交记录