互评示例

1

题目：W4 P1

M个好人和M 个非好人在深海上遇险，必须将一半的人投入海中，其余的人才能幸免于难，于是想 了一个办法:2M个人围成一圆圈，从第一个人开始依次报数，每数到第K个人就将他扔入大海，如 此循环进行直到仅余M个人为止。问怎样排法，才能使每次投入大海的都是坏人。(0<M<1000, 1<K<1000)

代码：

```
#include <iostream>
#include <ctime>
using namespace std;
/* run this program using the console pauser or add your own getch,
system("pause") or input loop */
int m,k;
const int maxn=1005;
char a[2*maxn];
int main(int argc, char** argv) {
  cin>>m;
  if(cin.fail())
  {
    cerr<<"wrong input!";
    exit(1);
  }
  cin>>k;
  if(cin.fail())
  {
    cerr<<"wrong input!";
    exit(1);
  }
  int pos=0;
  for(int i=1;i<=m;i++)//每次插入一个坏人，共m个
  {
    int count=k-1;//从当前位置开始报数，1-k意味着前进k-1个位置
    while(count)
    {
      pos++;
      pos=pos%(2*m);
      if(a[pos]!='@')//这个位置还没被坏人填充
      {
        count--;
      }
    }
```

```
      a[pos]='@';//填充
      while(a[pos]=='@')//找到下一个没有被扔下去的位置
      {
        pos++;
        pos=pos%(2*m);
      }
    }
    for(int i=0;i<2*m;i++)
    {
      if(a[i]!='@')
      {
        a[i]='+';
      }
      cout<<a[i];
    }
    cout<<endl;
    return 0;
}
```

评价示例：

7==程序结果正确，但有以下缺点：1.所有代码实现都写在主函数里，2.命名风格不完全统一，其中数组命名直接用a，意义不明3.直接定义容量为2010的数组，首先我觉得没必要定义2010，2000即可，其次这样当m较小时，会造成大量的空间浪费，动态分配内存会更好

优点：

1. 格式正确
2. 对程序正确性进行了评价
3. 能够指出代码中存在的具体问题
4. 能够给出建议的改进方式

建议改进的点：

1. 对于代码优点的评价，可以考虑从注释、代码健壮性、代码简洁性等方面进行评价

2

题目：W5 P2

撰写一个程序，从标准输入装置读取n个整数，并将读入的整数依次置入array及vector，然后遍历 这两种容器，求取数值总和，将总和及平均值输出至标准输出装置。

代码：

```
#include <iostream>
#include <vector>

template <typename elemType>//遍历数组，求sum与average
void obtainSumAndAverage(const elemType first, const long long len, long long&
sum, double& average)
{
sum = 0;
average = 0;
for (long long i = 0; i < len; i++)
{
sum += *(first + i);
}
average = static_cast<double>(sum) / static_cast<double>(len);//强制类型转换，比
（double）sum的形式要安全
}

int main()
{
long long n;
std::cin >> n;
if (n <= 0||std::cin.fail()==true)//读入检查
{
std::cout << "input Error" << std::endl;
exit(0);
}
long long* containerArray = new long long[n];//创建大小为n的containerArray
std::vector <long long > containerVector;
for (long long i = 0; i < n; i++)
{
std::cin >> containerArray[i];//读入containerArray
if (std::cin.fail() == true)//读入检查
{
std::cout << "input Error" << std::endl;
exit(0);
}
containerVector.push_back(containerArray[i]);//读入containerVector
}
long long sum = 0;
double average = 0;
```

```
obtainSumAndAverage(containerArray, n, sum, average);
obtainSumAndAverage(containerVector.begin(), n, sum, average);
std::cout << sum << " " << average << std::endl;
delete[]containerArray;//销毁containerArray
return 0;
}
```

评价示例：

10==代码相应正确，健壮性强。代码风格：1. [+]命名风格良好，思路清晰，鲁棒性良好；正确使用 C++ 风格类型转换，正确释放内存。2. [-]为了让每行不那么长，可以考虑给 long long 类型取一个别名；行内注释可以考虑对齐。3. [-]个人认为 std::cin.fail() == true 的 ==true 有些多余，"如果输入流失效"比"如果输入流失效为真"更加易读。4. [+]整体完成良好，体现了对知识的掌握，给满分。

优点：

1. 格式正确
2. 较为全面地从多个角度对程序进行了评价
3. 能够指出代码中存在的具体问题
4. 能够给出建议的改进方式

建议改进的点：

1. 将代码的优点和需要改进的点分开写可能更清晰

3

题目：W5 P2

撰写一个程序，从标准输入装置读取n个整数，并将读入的整数依次置入array及vector，然后遍历 这两种容器，求取数值总和，将总和及平均值输出至标准输出装置。

代码：

```
#include<iostream>
#include<vector>

using namespace std;

int evaluateSum(vector<int>Numbers) //一个只可以调用vector的函数，
                  //求解怎么写可以array和vector都能调用(т_т)
{
  int i = 0;
  long sum = 0;

  for ( i = 0; i < Numbers.size(); i++)
    sum += Numbers[i];

  return sum;
}

int main()
{
  static int num = 0;
  long sum = 0;
  int i = 0;

  cin >> num;

  if (num > 2147483647 || num < -2147483647) //输入判断，不能超过int范围
    return -1;

  int* myArray = new int[num + 1];
  vector<int> myVector(num + 1);

  for ( i = 0; i < num; i++)
  {
    cin >> myArray[i];
    myVector[i] = myArray[i];
  }

  sum = evaluateSum(myVector);

  cout << sum << " " << sum / num;
```

```
    delete[]myArray;

    return 0;
}
```

评价示例：

8==变量命名规范，代码可读性好，平均值=sum/num如果是小数会自动取整，建议改成(double)sum / num；(对于evaluateSum的调用的话，我可能会选择overloading的方式，但这可能不是最好的方法

优点：

1. 格式正确
2. 能够发现代码存在的优缺点
3. 能够给出建议的改进方式

建议改进的点：

1. 可以从更多角度对代码进行评价
2. 将评价分点写可能更清晰

4

题目：W5 P3

给定一个长度为 2n 的数组，将这些数分成 n 对，例如 (a1, b1), (a2, b2), ..., (an, bn) ， 使得从1 到 n 的 min(ai, bi) 总和最大，请计算上述min(ai, bi)总和的最大值。

代码：

```
//根据题意，先将数组排序，然后求第0，2，4...（2n-2） 项的和即可
#include<iostream>
#include<vector>
#include<algorithm>//调用sort所需
using namespace std;
int main()
{
  int array[10000];
  int i=0;
  int num=0;//数组元素个数
  int n;//n=num/2
  int res=0;//所求结果
  int input;
  char c;
  while(cin>>input)
  {
    array[i]=input;
    i++;
    c=getchar();
    if(c=='\n')
    break;
  }
  num=i;
  n=num/2;
  vector<int>v;//开启一个vector
  for(i=0;i<num;i++)
  {
    v.push_back(array[i]);
  }
  sort(v.begin(),v.end());//调用vector函数对元素进行从小到大排序
  for(i=0;i<num;i+=2)
  {
    res+=v[i];
  }
  cout<<res;
  system("pause");
  return 0;
}
```

评价示例：

8==注释清晰，结果正确。几点建议：1.只能处理在10000以内的结果，且可以在输入时直接将数据存储如vector提升效率，建议改进输入形式；2.可以考虑以更大的变量类型如long等存储加总结果以应对总和较大的情况

优点：

1. 格式正确
2. 能够发现代码存在的优缺点
3. 能够给出建议的改进方式
4. 能够考虑到数据范围的问题

建议改进的点：

1. 可以从更多角度对代码进行评价，如命名、冗余性等方面

作业评分说明

本学期每周作业采取教师打分+学生互评的方式给分，具体说明如下：

1. 教师和助教会对同学们提交的代码进行评价，给出分数。
2. 同学们需要相互评价他人的代码，评价时应尽可能地指出代码正确性、代码风格、鲁棒性等方面存在的问题，并给出分数。
3. 每位同学需要评价其他5位同学的代码。
4. 学生互评不规定字数，能够清晰指出代码存在的缺陷即可。
5. 每次作业的最终得分中，教师打分占50%，另50%将根据每位同学对其他同学作业的和合理性、完整性给出。即每次作业的得分，取决于学生本人代码的质量，以及学生对其他学生代码的评价的质量。
6. 每次作业将给出一周的时间完成代码，以及额外一周的时间完成评价，即每次作业期限为两周。

作业使用 GPT 说明：

大家也可以借助 GPT 等工具帮助你们进行代码走查，但不要简单的把 GPT 的回复照搬作为你的代码走查结果。进行代码 code review 可以参考如下的 Prompt：

You are an expert c++ code reviewer, providing feedback on the code below. As a code reviewer, your task is: Review the C++ code to follow the google c++ style guide and provide feedback. If there are any bugs, highlight them. Do not highlight minor issues and nitpicks. Use bullet points if you have multiple comments. If no suggestions are provided, please give good feedback. Please use Chinese to give feedback.