

青少年软件编程（C 语言）等级考试试卷（八级）

分数：100 题数：4

一、编程题(共 4 题，共 100 分)

1. City 不 City

“City 不 City”因为一位外国友人保保熊直播旅游时总是用奇怪的腔调说“好 city，啊！”而走红中国社交网络，成为网络热梗。事实上，有一些叛逆的年轻人在旅行时会刻意避开网红打卡点，选择一些小众的特色地方小城镇，不追求 city，而喜欢说“好 country，啊”。

下面给定各个城镇的旅游热度和城镇间的旅行花销，请你为前来咨询的旅行者规划一条最经济的路线，并且尽可能避开热度很高的网红点。

时间限制：6000

内存限制：65536

输入

输入第一行首先给出 4 个正整数： n 和 m ($1 < n \leq 10^3$, $1 \leq m \leq 5n$)，依次为城镇数量（于是城镇编号从 1 到 n ）和城镇间的通路条数； s 和 t 依次为旅行者的出发地和目的地的城镇编号。随后一行给出 n 个不超过 100 的正整数，依次为 n 个城镇的旅游热度。再后面是 m 行，每行给出一条通路连接的两个城镇的编号、这条通路的最小花销（其数值为不超过 10^3 的正整数）。通路是双向的，题目保证任一对城镇间至多给出一条通路。同一行的数字间均以空格分隔。

输出

题目要求从 s 到 t 的最小花销路线；若这样的路线不唯一，则取途经城镇的最高旅游热度值最小的那条路线。在一行中输出从 s 到 t 的最小花销、以及途经城镇的最高旅游热度值（若没有途经的城镇，则热度值为 0）。数值间以 1 个空格分隔，行首尾不得有多余空格。若从 s 根本走不到 t ，则在一行中输出 `Impossible`。

样例输入

样例 1：

8 14 7 8

100 20 30 10 50 80 100 100

7 1 1

7 2 2

7 3 1

7 4 2

1 2 1

1 5 2

2 5 1

3 4 1

3 5 3

3 6 2

4 6 1

5 6 1

5 8 1

6 8 2

样例 2：

3 1 1 2

10 20 30

1 3 1

样例输出

样例 1：

4 50

样例 2：

Impossible

提示

样例 1 解释：从 7 到 8 的最短路径有 3 条，其中 2 条都经过城镇 1，于是对应的最高旅游热度值是城镇 1 的热度值 100。解路径为 7->2->5->8，途径城镇 2 和 5，对应的最高旅游热度值是城镇 5 的热度值 50。在最短路径长度相等的情况下，取热度值小的解，故输出的热度值为 50。

试题编号：20241207-8-01

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：[点击浏览](#)

考生答案：（此题已作答）

考生得分：0

是否评分：已评分

评价描述：

2. 集体照

拍集体照时队形很重要，这里对给定的 N 个人 K 排的队形设计排队规则如下：

- 每排人数为 N/K （向下取整），多出来的人全部站在最后一排；
- 后排所有人的个子都不比前排任何人矮；
- 每排中最高者站中间（中间位置为 $m/2+1$ ，其中 m 为该排人数，除法向下取整）；
- 每排其他人以中间人为轴，按身高非增序，先右后左交替入队站在中间人的两侧（例如 5 人身高为 190、188、186、175、170，则队形为 175、188、190、186、170。这里假设你面对拍照者，所以你的左边是中间人的右边）；
- 若多人身高相同，则按名字的字典序升序排列。这里保证无重名。

现给定一组拍照人，请编写程序输出他们的队形。

时间限制：6000

内存限制：65536

输入

每个输入包含 1 个测试用例。每个测试用例第 1 行给出两个正整数 N ($\leq 10^4$ ，总人数) 和 K (≤ 10 ，总排数)。随后 N 行，每行给出一个人的名字（不包含空格、长度不超过 8 个英文字母）和身高 ($[30, 300]$ 区间内的整数)。

输出

输出拍照的队形。即 K 排人名，其间以空格分隔，行末不得有多余空格。注意：假设你面对拍照者，后排的人输出在上方，前排输出在下方。

样例输入

```
10 3
Tom 188
Mike 170
Eva 168
Tim 160
Joe 190
Ann 168
Bob 175
Nick 186
Amy 160
```

John 159

样例输出

Bob Tom Joe Nick

Ann Mike Eva

Tim Amy John

试题编号：20241207-8-02

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：[点击浏览](#)

考生答案：

```
#include<iostream>
#include<string>
#include<algorithm>
using namespace std;
struct people {
    string name;
    int height;
};
people p[10001], row[10001];
bool compare(people p1, people p2)
{
    if (p1.height != p2.height)
        return p1.height > p2.height;
    else
        return p1.name < p2.name;
}
void lineup(int begin, int number) {
    int i, j;
    if (number % 2)
    {

        lineup(begin, number - 1);
        row[number] = p[begin + number - 1];
    }
    else
    {
        for (j = begin + number - 1, i = 1; i <= number / 2; i++, j -= 2)
```

```

        row[i] = p[j];
        for (j = begin, i = number / 2 + 1; i <= number; i++, j += 2)
            row[i] = p[j];
    }
}
int main()
{
    int N, K, i, j, last_num, each_num;
    cin >> N >> K;
    for (i = 1; i <= N; i++)
        cin >> p[i].name >> p[i].height;
    each_num = N / K;
    last_num = N - N / K * K + each_num;
    sort(p+1, p + N+1, compare);

    lineup(1, last_num);
    for (i = 1; i < last_num; i++)
        cout << row[i].name << " ";
    cout << row[last_num].name << endl;

    for (i = 1; i < K; i++)
    {
        lineup(last_num + (i-1)*each_num + 1, each_num);
        for (j = 1; j < each_num; j++)
            cout << row[j].name << " ";
        cout << row[each_num].name;
        if(i!=K-1)
            cout << endl;
    }
    return 0;
}

```

考生得分：25

是否评分：已评分

评价描述：

3. 千手观音

人类喜欢用 10 进制，大概是因为人类有一双手 10 根手指用于计数。于是在千手观音的世界里，数字都是 10 000 进制的，因为每位观音有 1 000 双手 ……

千手观音们的每一根手指都对应一个符号（但是观音世界里的符号太难画了，我们暂且用小写英文字母串来代表），就好像人类用自己的 10 根手指对应 0 到 9 这 10 个数字。同样的，就像人类把这 10 个数字排列起来表示更大的数字一样，他们也把这些名字排

列起来表示更大的数字，并且也遵循左边高位右边低位的规则，相邻名字间用一个点`.`分隔，例如`pat.pta.cn`表示千手观音世界里的一个 3 位数。

人类不知道这些符号代表的数字的大小。不过幸运的是，人类发现了千手观音们留下的一串数字，并且有理由相信，这串数字是从小到大有序的！于是你的任务来了：请你根据这串有序的数字，推导出千手观音每只手代表的符号的相对顺序。

注意：有可能无法根据这串数字得到全部的顺序，你只要尽量推出能得到的结果就好了。当若干根手指之间的相对顺序无法确定时，就暂且按它们的英文字典序升序排列。例如给定下面几个数字：

```
pat
cn
lao.cn
lao.oms
pta.lao
pta.pat
cn.pat
```

我们首先可以根据前两个数字推断`pat` < `cn`；根据左边高位的顺序可以推断`lao` < `pta` < `cn`；再根据高位相等时低位的顺序，可以推断出`cn` < `oms`，`lao` < `pat`。综上我们得到两种可能的顺序：`lao` < `pat` < `pta` < `cn` < `oms`；或者`lao` < `pta` < `pat` < `cn` < `oms`，即`pat`和`pta`之间的相对顺序无法确定，这时我们按字典序排列，得到`lao` < `pat` < `pta` < `cn` < `oms`。

时间限制：7000

内存限制：65536

输入

输入第一行给出一个正整数 N ($\leq 10^5$)，为千手观音留下的数字的个数。随后 N 行，每行给出一个千手观音留下的数字，不超过 10 位数，每一位的符号用不超过 3 个小写英文字母表示，相邻两符号之间用`.`分隔。我们假设给出的数字顺序在千手观音的世界里是严格递增的。题目保证数字是 10^4 进制的，即符号的种类肯定不超过 10^4 种。

输出

在一行中按大小递增序输出可推理出的符号，符号间仍然用`.`分隔。

样例输入

```
7
pat
cn
```

```
lao.cn
lao.oms
pta.lao
pta.pat
cn.pat
```

样例输出

```
lao.pat.pta.cn.oms
```

试题编号：20241207-8-03

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：[点击浏览](#)

考生答案：（此题已作答）

考生得分：0

是否评分：已评分

评价描述：

4. 最多删除两个字符

给定一个仅由小写英文字母组成的字符串，**最多删两个字符**后，能得到多少种不同的字符串？

时间限制：5000

内存限制：65536

输入

输入在一行中给出长度在区间 $[3, 10^6]$ 内的仅由小写英文字母组成的字符串。

输出

在一行中输出最多删两个字符后所能得到的不同字符串的个数。

样例输入

```
ababcc
```

样例输出

```
15
```

提示

样例解释：

- 1、删除 0 个字符后得到 `ababcc`；
- 2、删除 1 个字符后得到 `babcc`, `aabcc`, `abbcc`, `abacc`, `ababc`；
- 3、删除 2 个字符后得到 `abcc`, `bbcc`, `bacc`, `babcc`, `aacc`, `aabc`, `abbc`, `abac`, `abab`。

试题编号：20241207-8-04

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：[点击浏览](#)

考生答案：

```
#include <bits/stdc++.h>
#define int long long
using namespace std;
const int N = 1e6 + 10;
int dp[N][4];

signed main()
{
    ios::sync_with_stdio(false);
    cin.tie(0), cout.tie(0);
    string s, stemp;
    cin >> stemp;
    s.assign("0");
    s.append(stemp);
    dp[0][0] = 1;
    for (int i = 1; i < s.size(); ++i)
    {
        for (int j = 0; j <= 3; ++j)
        {
            dp[i][j] = dp[i - 1][j];
            if (j)
                dp[i][j] = dp[i][j] + dp[i - 1][j - 1];
            for (int k = i - 1; k >= 1 && i - k <= j; --k)
            {
                if (s[k] == s[i])
                {
                    dp[i][j] = dp[i][j] - dp[k - 1][j - (i - k)];
                    break;
                }
            }
        }
    }
}
```



```
    }  
}  
int ans = 0;  
for (int i = 0; i <= 2; ++i)  
{  
    ans = ans + dp[s.size() - 1][i];  
}  
cout << ans << endl;  
return 0;  
}
```

考生得分：25

是否评分：已评分