

## C-比赛的冠亚季军

### 题目描述：

有  $N$  ( $3 \leq N < 10000$ ) 个运动员，他们的  $id$  为  $0$  到  $N-1$ ，他们的实力由一组整数表示。他们之间进行比赛，需要决出冠亚军。比赛的规则是  $0$  号和  $1$  号比赛， $2$  号和  $3$  号比赛，以此类推，每一轮，相邻的运动员进行比赛，获胜的进入下一轮；实力值大的获胜，实力值相等的情况， $id$  小的情况下获胜；轮空的直接进入下一轮。

### 输入描述：

输入一行  $N$  个数字代表  $N$  的运动员的实力值 ( $0 \leq \text{实力值} \leq 10000000000$ )。

### 输出描述：

输出冠亚军的  $id$ ，用空格隔开。

### 补充说明：

#### 示例 1

##### 输入：

2 3 4 5

##### 输出：

3 1 2

##### 说明：

第一轮比赛， $id$  为  $0$  实力值为  $2$  的运动员和  $id$  为  $1$  实力值为  $3$  的运动员比赛， $1$  号胜出进入下一轮争夺冠亚军， $id$  为  $2$  的运动员和  $id$  为  $3$  的运动员比赛， $3$  号胜出进入下一轮争夺冠亚军；冠亚军比赛， $3$  号胜  $1$  号；故冠军为  $3$  号，亚军为  $1$  号。 $2$  号与  $0$  号，比赛进行季军的争夺， $2$  号实力值为  $4$ ， $0$  号实力值  $2$ ，故  $2$  号胜出，得季军。冠亚季军为  $3\ 1\ 2$ 。

```
#include <stdio.h>
#include<string.h>
char inp[10000010];
struct peo {
    int index;
    int fen;
} p[100010], q[100010], s, L, s1, L1;
int main() {
    gets(inp);
    int now = 0;
    int k = 0;
    for (int i = 0; i < strlen(inp); i++) {
        if (inp[i] == ' ') {
            p[k].fen = now;
            now = 0;
            p[k].index = k;
        }
    }
}
```

```

        k++;
    } else {
        now = now * 10 + inp[i] - '0';
    }
}

p[k].fen = now;
p[k].index = k;
k++;
while (k > 4) {
    int kk = 0;
    for (int i = 0; i < k; i += 2) {
        if (i + 1 < k) {
            if (p[i].fen >= p[i + 1].fen) {
                q[kk].fen = p[i].fen;
                q[kk].index = p[i].index;
            } else {
                q[kk].fen = p[i + 1].fen;
                q[kk].index = p[i + 1].index;
            }
        } else {
            q[kk].fen = p[i].fen;
            q[kk].index = p[i].index;
        }
        kk++;
    }
    k = kk;
    for (int i = 0; i < k; i++) {
        p[i].fen = q[i].fen;
        p[i].index = q[i].index;
    }
}

```

```

int G, Y, J;
if (k == 3) {
    s = p[0].fen >= p[1].fen ? p[0] : p[1];
    L = p[0].fen < p[1].fen ? p[0] : p[1];
    J = L.index;
    G = s.fen >= p[2].fen ? s.index : p[2].index;
    Y = s.fen < p[2].fen ? s.index : p[2].index;
} else {
    s = p[0].fen >= p[1].fen ? p[0] : p[1];
    L = p[0].fen < p[1].fen ? p[0] : p[1];
    s1 = p[2].fen >= p[3].fen ? p[2] : p[3];

```

```
    L1 = p[2].fen < p[3].fen ? p[2] : p[3];
    G = s.fen >= s1.fen ? s.index : s1.index;
    Y = s.fen < s1.fen ? s.index : s1.index;
    J = L.fen >= L1.fen ? L.index : L1.index;

}
printf("%d %d %d", G, Y, J);

}
```