

题目描述

给定一个射击比赛成绩单，包含多个选手若干次射击的成绩分数，请对每个选手按其最高3个分数之和进行降序排名，输出降序排名后的选手ID序列。

条件如下

- 1. 一个选手可以有多个射击成绩的分数，且次序不固定。
- 2. 如果一个选手成绩少于3个，则认为选手的所有成绩无效，排名忽略该选手。
- 3. 如果选手的成绩之和相等，则成绩之和相等的选手按照其ID降序排列。

输入描述

- 输入第一行，一个整数N，表示该场比赛总共进行了N次射击，产生N个成绩分数（ $2 \leq N \leq 100$ ）。
- 输入第二行，一个长度为N整数序列，表示参与每次射击的选手ID（ $0 \leq ID \leq 99$ ）。
- 输入第三行，一个长度为N整数序列，表示参与每次射击的选手对应的成绩（ $0 \leq \text{成绩} \leq 100$ ）。

输出描述

符合题设条件的降序排名后的选手ID序列。

用例

输入	13 3,3,7,4,4,4,4,7,7,3,5,5,5 53,80,68,24,39,76,66,16,100,55,53,80,55
输出	5,3,7,4
说明	<p>该场射击比赛进行了13次 参赛的选手为3,4,5,7</p> <ul style="list-style-type: none">• 3号选手成绩：53,80,55，最高3个成绩的和为： 80+55+53=188。• 4号选手成绩：24,39,76,66，最高3个成绩的和为： 76+66+39=181。• 5号选手成绩：53,80,55，最高3个成绩的和为： 80+55+53=188。• 7号选手成绩：68,16,100，最高3个成绩的和为： 100+68+16=184。 <p>比较各个选手最高3个成绩的和，有3号=5号>7号>4号，由于3号 和5号成绩相等且ID号5>3，所以输出为：5,3,7,4</p>

题目解析

简答的排序问题，按照题目要求写排序规则即可。

Java算法源码

```
1 import java.util.*;
2
3 public class Main {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         int n = Integer.parseInt(sc.nextLine());
8
9         Integer[] ids =
10             Arrays.stream(sc.nextLine().split(",")).map(Integer::parseInt).toArray(Integer[]::new);
11
12         Integer[] scores =
13             Arrays.stream(sc.nextLine().split(",")).map(Integer::parseInt).toArray(Integer[]::new);
14
15         System.out.println(getResult(n, ids, scores));
16     }
17 }
```

```
18 public static String getResult(int n, Integer[] ids, Integer[] scores) {
19     HashMap<Integer, ArrayList<Integer>> players = new HashMap<>();
20
21     for (int i = 0; i < n; i++) {
22         players.putIfAbsent(ids[i], new ArrayList<>());
23         players.get(ids[i]).add(scores[i]);
24     }
25
26     ArrayList<int[]> ans = new ArrayList<>();
27
28     for (int id : players.keySet()) {
29         ArrayList<Integer> idScores = players.get(id);
30         if (idScores.size() >= 3) {
31             int total =
32                 idScores.stream().sorted((a, b) -> a - b).limit(3).reduce(Integer::sum).orElse(0);
33
34             ans.add(new int[] {id, total});
35         }
36     }
37
38     ans.sort((a, b) -> a[1] != b[1] ? b[1] - a[1] : b[0] - a[0]);
39
40     StringJoiner sj = new StringJoiner(",");
41     for (int[] player : ans) sj.add(player[0] + "");
42
43     return sj.toString();
44 }
45 }
```

JS算法源码

```
1  /* JavaScript Node ACM模式 控制台输入获取 */
2  const readline = require("readline");
3
4  const rl = readline.createInterface({
5    input: process.stdin,
6    output: process.stdout,
7  });
8
9  const lines = [];
10 rl.on("line", (line) => {
11   lines.push(line);
12   if (lines.length == 3) {
13     const n = lines[0] - 0;
14     const ids = lines[1].split(",").map(Number);
15     const scores = lines[2].split(",").map(Number);
16
17     console.log(getResult(n, ids, scores));
18
19     lines.length = 0;
20   }
21 });
22
23 function getResult(n, ids, scores) {
24   const player = {};
25
26   for (let i = 0; i < n; i++) {
27     const id = ids[i];
28     const score = scores[i];
29
30     if (!player[id]) player[id] = [];
31     player[id].push(score);
32   }
33
34   const ans = [];
35
36   for (let id in player) {
37     if (player[id].length >= 3) {
38       const total = player[id]
39         .sort((a, b) => b - a)
40         .slice(0, 3)
41         .reduce((a, b) => a + b);
42
43       ans.push([id, total]);
44     }
45   }
46
47   return ans
48     .sort((a, b) => (a[1] != b[1] ? b[1] - a[1] : b[0] - a[0]))
49     .map((x) => x[0])
50     .join(",");
51 }
```

Python算法源码

```
1 # 输入获取
2 n = int(input())
3 ids = list(map(int, input().split(",")))
4 scores = list(map(int, input().split(",")))
5
6
7 # 算法入口
8 def getResult():
9     players = {}
10
11     for i in range(n):
12         players.setdefault(ids[i], [])
13         players.get(ids[i]).append(scores[i])
14
15     ans = []
16
17     for pid in players:
18         if len(players[pid]) >= 3:
19             players[pid].sort(reverse=True)
20             ans.append((pid, sum(players[pid][:3])))
21
22     ans.sort(key=lambda x: (-x[1], -x[0]))
23
24     return ",".join(map(lambda x: str(x[0]), ans))
25
26
27 # 算法调用
28 print(getResult())
```