

青少年软件编程（C 语言）等级考试试卷（二级）

分数：100 题数：5

一、编程题(共 5 题，共 100 分)

1. 不同的数字

题目描述

给定 n 个正整数 $a_1, a_2, \dots, a_n$ ，请统计出这个数列中含有多少个不同的整数。

输入格式

- 第一行：一个整数 n ；
- 第二行： n 个整数 $a_1, a_2, \dots, a_n$ 。

输出格式

一个整数，数列中含有多少个不同的整数。

输入样例#1

```
6
1 6 5 6 3 1
```

输出样例#1

```
4
```

输入样例#2

```
10
1 2 3 4 5 6 7 8 9 10
```

输出样例#2

```
10
```

说明提示

【数据范围】

$$1 \leq n, a_i \leq 10^5。$$

限制

时间限制：1000ms

内存限制：256MiB

试题编号：20250906-2-01

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](#)

参考代码：

```
#include <iostream>
```

```
int main()
{
    int n ,ans = 0;
    std::cin >> n;
    bool a[100006] = {0};
    for(int i = 1; i <= n ;++i){
        int x;
        std::cin >> x;
        if(a[x] == 0){
            a[x] = true;
            ans++;
        }
    }
    std::cout << ans ;
}
```

考生得分：20

是否评分：已评分

评价描述：

2. 丢失的数字

题目描述

给定整数 n 和一个包含 $n - 1$ 个整数的序列，该序列由区间 $[1, n]$ 内的 $n - 1$ 个互异整数组成。请找出该区间内缺失的唯一整数。

输入格式

- 第一行：一个整数 n ；
- 第二行： $n - 1$ 个整数。

输出格式

一个整数，表示丢失的数字。

输入样例^{#1}

```
6
1 2 5 6 3
```

输出样例^{#1}

```
4
```

输入样例^{#2}

```
10
7 2 3 9 4 8 1 6 10
```

输出样例^{#2}

```
5
```

说明提示

【数据范围】

$1 \leq n \leq 2 \times 10^4$, 输入序列保证合法。

限制

时间限制：1000ms

内存限制：256MiB

试题编号：20250906-2-02

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](#)

参考代码：

```
#include<iostream>
int main(){
    int n,s,d;
    std::cin>>n;
    s=(1+n)*n/2;
    for(int i=1;i<=n-1;i++){
        std::cin>>d;
        s-=d;
    }
    std::cout<<s;
    return 0;
}
```

考生得分：20

是否评分：已评分

评价描述：

3. 维修任务

题目描述

题目描述

在一条商业街上，有 100 个店铺，店铺从左到右编号为 1 到 100。工人从店铺 x 移动到 y 发生的移动距离为 $|y - x|$ 。

现在需要完成 n 个修理任务。其中第 i 个任务要求工人移动到店铺 a_i ，报修任务分两类，用 L 表示任务分配给小李，用 R 表示任务分配给小任。

必须按照报修的顺序来完成这些任务。请计算，完成所有修理任务后，两位工人的总移动的总距离。工人最开始的位置可以按照最理想的情况安排。

输入格式

- 第一行：一个整数 n 表示搬运次数；
- 接下来 n 行，每行一个整数 a_i 表示商店位置，一个字符 S_i 表示哪位工人完成。

输出格式

输出所有任务完成时的最小总距离。

输入样例#1

```
4
3 L
6 R
9 L
2 R
```

输出样例#1

10

输入样例#2

3
2 L
2 L
99 L

输出样例#2

97

输入样例#3

8
22 L
75 L
26 R
45 R
72 R
81 R
47 L
29 L

输出样例#3

154

说明提示

- $1 \leq N \leq 100$;
- $1 \leq a_i \leq 100$;
- $s_i \in \{ 'L', 'R' \}$ 。

限制

时间限制：1000ms

内存限制：256MiB

试题编号：20250906-2-03

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

参考代码：

```
#include <iostream>

int main()
{
    int n , a , l = -1, r = -1;
    std::cin >> n;
    int ans = 0;
    for (int i = 0; i < n; ++i) {
        int a;
        char c;
        std::cin >> a >> c;
        if (c == 'L') {
            if (l != -1) {
                int dis = a - l;
                if(dis < 0) dis = -dis;
                ans += dis;
            }
            l = a;
        }
        else {
            if (r != -1) {
                int dis = a - r;
                if(dis < 0) dis = -dis;
                ans += dis;
            }
            r = a;
        }
    }
    std::cout << ans ;
    return 0;
}
```

考生得分：20

是否评分：已评分

评价描述：

4. 硬币

题目描述

可以使用任意数量的 a 元硬币、 b 元硬币和 c 元硬币。

请找出恰好凑出 n 元所需的最小硬币总数。

若无法凑出，则输出 -1 。

输入格式

- 第一行，整数 n ；
- 第二行，三个整数表示 a, b, c 。

输出格式

输出最小硬币总数（若无法凑出则输出 -1 ）。

输入样例^{#1}

```
100
20 40 50
```

输出样例^{#1}

```
2
```

输入样例^{#2}

```
99
```

1 5 10

输出样例#2

14

说明提示

【数据范围】

- $1 \leq N \leq 10^9$;
- $1 \leq A \leq B \leq C \leq 10^9$;
- 保证最终最小硬币总数不超过 10^4 。

限制

时间限制：1000ms

内存限制：256MiB

试题编号：20250906-2-04

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：（此题已作答）

考生得分：3

是否评分：已评分

评价描述：

5. 游戏

题目描述

游戏中有 N 个敌人，第 i 个敌人有 h_i 的血量，当敌人血量降至 0 或以下时该敌人被消灭。

按照顺序从第 1 个敌人开始打，时间 T 从 0 开始，每次攻击前，先将 T 增加 1，若此时 T 为 3 的倍数，当前敌人掉 3 滴血，否则掉 1 滴血。

求消灭全部敌人所需时间。

输入格式

- 第一行输入一个正整数 N ；
- 第二行输入 N 个正整数 $H_1, H_2, \dots, H_n$ 。

输出格式

输出一行答案 T 。

输入样例#1

```
3
3 1 4
```

输出样例#1

```
6
```

输入样例#2

```
6
3 14 15 9 2 6
```

输出样例#2

```
33
```

说明提示

【数据范围】

- $1 \leq N \leq 2 \times 10^5$;
- $1 \leq H_i \leq 10^9$ 。

限制

时间限制：1000ms

内存限制：256MiB

试题编号：20250906-2-05

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](#)

参考代码：

```
#include <iostream>
int main()
{
    int n, a;
    std::cin >> n;
    long long cnt = 0;
    for (int i = 1; i <= n; i++)
    {
        std::cin >> a;
        int k = a / 5;
        int x = cnt % 3;
        cnt += k * 3;
        k = a % 5;
        if (!k) continue;
        if (x == 0)
        {
            if (k <= 2) cnt += k;
            else cnt += 3;
        }
    }
}
```

```
    }  
    else if (x == 1)  
    {  
        if (k <= 1) cnt++;  
        else cnt += 2;  
    }  
    else if (x == 2)  
    {  
        if (k <= 3) cnt++;  
        else cnt += 2;  
    }  
  
}  
std::cout << cnt ;  
return 0;  
}
```

考生得分：0

是否评分：已评分

评价描述：