

# 热身赛题解

---

## A、夹数

出题人：李嘉龙

因为 $n$ 的范围较大，直接枚举 $1\sim n$ 的所有数字再判断其是不是夹数是不现实的。

所以我们可以枚举平方数或者立方数，然后判断其相邻的数字是不是夹数，这样枚举的量就只有 $\sqrt{n}$ 或者 $\sqrt[3]{n}$ 。

枚举写法需要注意的点有：

- 1、枚举过程中数字的大小一定会超出int的范围，所以要使用longlong型变量。
- 2、尽量避免使用pow函数，其糟糕的精度问题极易导致答案出错。

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
int main()//暴力写法
{
    long long n;
    scanf("%lld",&n);
    int ans=0;
    for(long long i=1;i*i*i<=n+1;i++)
    {
        long long b=i*i*i;
        long long x=b-1,a=b-2;
        long long z=sqrt(a);
        if(z*z==a) ans++;
    }
    printf("%d\n",ans);
    return 0;
}
```

实际上还有一个更快的写法：

可以证明，夹数仅存在一个，即26，所以只需要判断 $n$ 与26的大小即可。

相关证明可以通过搜索引擎查到，该结论亦可，这说明线上比赛利用搜索引擎的能力也是很重要的。（笑）

## B、石子分堆游戏

出题人：高言峰

假如一开始的堆数为  $n$ ，石子总数为  $S$ ，那么最后的状态必然是  $S$  堆石子，每堆各一颗。无论怎么分，都需要进行恰好  $S - n$  轮操作。如果  $S - n$  是奇数，则最后一次操作是由小G完成的，小G胜；否则，小G的朋友胜。

```
#include <stdio.h>
int main()
{
    int t, n, x;
    scanf("%d", &t);
    while (t--)
    {
        scanf("%d", &n);
        int sum = 0;
        for (int i = 0; i < n; ++i)
        {
            scanf("%d", &x);
            sum += x;
        }
        puts((sum - n) % 2 ? "win" : "Lose");
    }
    return 0;
}
```

## C、达拉崩吧

出题人：高言峰

这道题没有思维难度，直接按要求排序即可。因为数据比较小，可以使用冒泡排序。比较时，先用 `strlen` 比较字符串的长度，如果相等，则用 `strcmp` 比较字典序。

```
#include <stdio.h>
#include <string.h>
char names[505][505], *name_ptrs[505];
int less_than(char *s1, char *s2)
{
    int len1 = strlen(s1), len2 = strlen(s2);
    if (len1 < len2)
        return 1;
    else if (len1 > len2)
        return 0;
    return strcmp(s1, s2) < 0;
}
void sort(char *name_ptrs[], int n)
{
    for (int i = 0; i < n - 1; ++i)
        for (int j = 0; j < n - 1; ++j)
```

```

        if (less_than(name_ptrs[j], name_ptrs[j + 1]))
        {
            char *t = name_ptrs[j];
            name_ptrs[j] = name_ptrs[j + 1];
            name_ptrs[j + 1] = t;
        }
    }
}

int main()
{
    int n;
    scanf("%d", &n);
    for (int i = 0; i < n; ++i)
        scanf("%s", names[i]);
    for (int i = 0; i < n; ++i)
        name_ptrs[i] = names[i];
    sort(name_ptrs, n);
    for (int i = 0; i < n; ++i)
        printf("%s\n", name_ptrs[i]);
    return 0;
}

```

## D、Disco Elysium I

出题人：李泽政

不需要任何的思维，只需要不断地尝试即可得到答案。

但是：

- 1、可以通过搜索引擎查找题面相关内容，直接得到答案。
- 2、也可以通过每次二分枚举，来减少枚举的次数，更快地得到答案。

## 总结

热身赛的全部赛题都没有考到任何的算法，只需要学会基础的C语言语法即可完成。重要的在于参赛者的思考，以及细节的处理。

当然，热身赛的题目也会显得有些非正式，比如夹数只有一个、以及像猜数字这样略显滑稽的题目，但毕竟热身赛只是用于让参赛者适应OJ的。

可以保证，周赛的题目会更加正式、规范化。