

# C语言刷题训练营-第十二讲

比特就业课@提供课程讲解和技术支持

题目来自: [牛客网](#)

## BC130-小乐乐查找数字

### 题目描述

给定n个整数和一个整数x，小乐乐想从n个整数中判断x出现几次，请编程帮他实现。

### 输入描述:

共3行  
第一行，一个整数，表示n ( $1 \leq n \leq 100$ )。  
第二行，共输入n个整数，两个整数之间用空格分隔。  
第三行，输入一个整数，表示要查找的整数x。

### 输出描述:

一行，表示整数x出现的次数。

### 示例1

#### 输入

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

#### 输出

```
3
```

### 参考代码:

```
#include <stdio.h>
int main()
{
    int arr[101] = {0};
    int n = 0;
    scanf("%d", &n);
    int i = 0;
    for(i=0; i<n; i++)
    {
        int m = 0;
        scanf("%d", &m);
        arr[m]++;
    }
}
```

```
int x = 0; 比特就业课，400小时就业课，保姆式就业服务
scanf("%d", &x);
printf("%d", arr[x]);
return 0;
}
```

## BC131-kiki学程序设计基础

### 题目描述

BoBo老师教了KiKi学习程序设计基础，他知道C++是带类的C语言，这个“++”主要包含三部分内容：对C语言进行语法上的扩展、面向对象（封装、继承和多态），STL(即模板)。这学期KiKi学习了C和C++，学好他们，其他编程语言都可以很轻松掌握。C和C++的输入有区别，请帮KiKi输出C和C++输出Hello world的代码。

### 输入描述:

无

### 输出描述:

```
printf("Hello world!\n");
cout << "Hello world!" << endl;
```

### 参考代码:

```
#include <stdio.h>

int main()
{
    //难点: 要让转义字符不是转义字符
    printf("printf(\"Hello world!\\n\");\n");
    printf("cout << \"Hello world!\" << endl;");

    return 0;
}
```

## BC135-kiki求质数的个数

### 题目描述

KiKi知道了什么是质数（只能被1和他自身整除的数），他现在想知道所有三位整数中，有多少个质数。

### 输入描述:

无

### 输出描述:

### 参考代码：

```
#include <stdio.h>

int is_prime(int n)
{
    int i = 0;
    for(i=2; i<n; i++)
    {
        if(n%i == 0)
            return 0;
    }
    return 1;
}

int main()
{
    int count = 0;
    int i = 0;
    for(i=100; i<=999; i++)
    {
        //判断i是否为质数
        if(is_prime(i))
        {
            count++;
        }
    }
    printf("%d\n", count);
    return 0;
}
```

## BC136-kiki去重整数并排序

### 题目描述

给定一个整数序列，KiKi想把其中的重复的整数去掉，并将去重后的序列从小到大排序输出。

### 输入描述:

第一行，输入一个整数n，表示序列有n个整数。

第二行输入n个整数（每个整数大于等于1，小于等于1000），整数之间用空格分隔。

### 输出描述:

去重并且从小到大排列的整数序列，整数之间用空格分隔。

### 示例1

#### 输入

6  
5 3 3 4 2 2

输出

2 3 4 5

参考代码：

```
//方法1
#include <stdio.h>
int main()
{
    int n = 0;
    int arr[100];
    scanf("%d", &n);
    int i = 0;
    for (i = 0; i < n; i++)
    {
        scanf("%d", &arr[i]);
    }
    //排序
    for (i = 0; i < n - 1; i++)
    {
        int j = 0;
        for (j = 0; j < n - 1 - i; j++)
        {
            if (arr[j] > arr[j + 1])
            {
                int tmp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = tmp;
            }
        }
    }
    //去重-比较n-1对
    int j = 0;
    for (i = 0; i < n - 1; i++)
    {
        //把i+1后边的元素往前移动
        if (arr[i] == arr[i + 1])
        {
            int k = 0;
            for (k = i; k < n - 1; k++)
            {
                arr[k] = arr[k + 1];
            }
            n--;
            i--;
        }
    }
    //打印
    for (i = 0; i < n; i++)
    {
        printf("%d ", arr[i]);
    }
}
```

```

        return 0;
    }

    //方法2:
    int main()
    {
        int n = 0;
        int arr[1001] = {0};
        scanf("%d", &n);
        int i = 0;
        int m = 0;
        for(i=0; i<n; i++)
        {
            scanf("%d", &m);
            arr[m] = m;
        }
        for(i=0; i<1001; i++)
        {
            if(arr[i] != 0)
                printf("%d ", arr[i]);
        }
        return 0;
    }

```

## BC138-KIKI学结构体和指针

### 题目描述

KiKi学习了结构体和指针，他了解了结构体类型可以定义包含多个不同类型成员，而指针本质是内存地址，是引用数据的另外一种方式。现在他想将多个输入的数据通过结构体和指针的方式连接在一起，形成一个单向链表，即：每个结点是结构体类型，包括整型数据成员（data）和结构体指针类型成员（next），每个结点的指针部分指向下一个输入的结点。具体建立过程如下：先输入n个整数，按照数据输入的顺序建立一个带头结点的单链表，再输入一个数据m,将单链表中的值为m的结点全部删除。输出删除后的单链表信息。

### 输入描述:

包括三行：  
第一行输入数据个数n（ $3 \leq n \leq 100$ ）；第二行依次输入n个整数，用空格分隔；第三行输入欲删除数据m。

### 输出描述:

包括两行：  
第一行输出完成删除后的单链表长度；第二行依次输出完成删除后的单链表数据。

### 示例1

#### 输入

```

5
1 2 2 3 4
2

```

```
3
1 3 4
```

## 参考代码：

```
#include <stdio.h>
#include <stdlib.h>

struct Node
{
    int data;
    struct Node* next;
};

int main()
{
    int n = 0;
    struct Node* list = NULL;
    struct Node* tail = NULL;
    //输入结点
    int i = 0;
    scanf("%d", &n);
    for (i = 0; i < n; i++)
    {
        int data = 0;
        scanf("%d", &data);
        struct Node *n = (struct Node*)malloc(sizeof(struct Node));
        n->data = data;
        n->next = NULL;

        if (list == NULL)
        {
            list = n;
            tail = list;
        }
        else
        {
            tail->next = n;
            tail = tail->next;
        }
    }

    //删除结点
    int del = 0;
    scanf("%d", &del);
    struct Node *cur = list;
    struct Node *prev = NULL;
    while (cur)
    {
        //找到了
        if (cur->data == del)
        {
            struct Node* pd = cur;
```

```
//如果删除的是第一个结点
if (cur == list)
{
    list = list->next;
    cur = list;
}
else
{
    //删除的不是第一个结点
    prev->next = cur->next;
    cur = prev->next;
}
n--;
free(pd);
}
else//找不到
{
    prev = cur;
    cur = cur->next;
}
}
cur = list;
printf("%d\n", n);
while (cur)
{
    printf("%d ", cur->data);
    cur = cur->next;
}
//释放空间
cur = list;
while (cur)
{
    del = cur;
    cur = cur->next;
    free(del);
}

return 0;
}
```

## BC139-KIKI定义电子日历类

### 题目描述

KiKi学习了面向对象技术，学会了通过封装属性（变量）和行为（函数）定义类，现在他要设计一个电子日历类TDate。

它有3个私有数据成员：Month,Day,Year和若干个公有成员函数，要求：

- (1) 带有默认形参值的构造函数，默认值为0, 0, 0;
- (2) 输出日期函数，用“日/月/年”格式输出日期;
- (3) 设置日期函数，从键盘输入年、月、日。

### 输入描述:

## 输出描述:

一行，用“日/月/年”格式输出日期。

## 示例1

### 输入

2019 12 30

### 输出

30/12/2019

参考代码:

```
#include <iostream>

using namespace std;

class TDate
{
private:
    int Year;
    int Month;
    int Day;
public:
    TDate(int year=0, int month=0, int day = 0)
    {
        Year = year;
        Month = month;
        Day = day;
    }
    void Print()
    {
        cout<<Day<<"/"<<Month<<"/"<<Year<<endl;
    }
};

int main()
{
    int year = 0;
    int month = 0;
    int day = 0;
    cin>>year>>month>>day;
    TDate d(year, month, day);
    d.Print();
    return 0;
}
```



比特就业课，400小时就业课，保姆式就业服务  
比特IT读书会@提供课程讲解和技术支持

联系鹏哥：15596668862（同微信）

[完整的C语言教程链接](#)

比特就业课