

第四章 函数

第 4 6 课 自己定义函数

《信息学奥赛一本通·编程启蒙 C++版》

函数的定义的语法形式

数据类型 函数名（形式参数表）

{

函数体

//执行语句

}

【例 46.1】 完全数

【题目描述】

一个数如果恰好等于它不包括自身的因子之和,这个数就称为"完全数"。例如, 6 的因子为 1、2、3, 而 $6 = 1 + 2 + 3$, 因此 6 是"完全数"。 编程序找出 N 之内的所有完全数, 并按下面格式输出其因子 (这里的因子里面, 不包含自己)。

【输入格式】

输入一个整数 $n(0 \leq n \leq 1000)$ 。

【输出格式】

形式如: 一个整数, 然后 `its factors are` 后面一些整数, 具体可参考样例输出

`? its factors are ? ? ?`

【样例输入】

`10`

【样例输出】

`6 its factors are 1 2 3`

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n;
4. void check(int x){
5.     int sum=0;
6.     for(int i=1;i<x;i++){
7.         if(x%i==0)sum+=i;
8.     }
9.     if(sum==x){
10.         cout<<x<<" its factors are ";
11.         for(int i=1;i<x;i++){
12.             if(x%i==0)cout<<i<<' ';
13.         }
14.         cout<<endl;
15.     }
16.}
17.int main(){
18.    cin>>n;
19.    for(int i=1;i<=n;i++){
20.        check(i);
21.    }
22.    return 0;
23.}
```

【例 46.2】 数字统计

【题目描述】

请统计某个给定范围 $[L, R]$ 的所有整数中，数字 7 出现的次数。

比如给定范围 $[60, 80]$ 中，7 一共出现 12 次。分别是 67, 77 的个位，以及 70 ~ 79 的十位。

【输入格式】

一行两个数 L, R 表示范围，用空格分隔。

【输出格式】

一个整数表示数字 7 出现的次数。

【样例输入】

60 80

【样例输出】

12

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int l,r,cnt;
4. int sum(int x){
5.     int cnt=0;
6.     while(x>0){
7.         if(x%10==7) cnt++;
8.         x/=10;
9.     }
10.    return cnt;
11.}
12.int main(){
13.    cin>>l>>r;
14.    for(int i=1;i<=r;i++)
15.        cnt+=sum(i);
16.    cout<<cnt;
17.    return 0;
18.}
```

【例 46.3】 素数回文数的个数

【题目描述】

求 11 到 n 之间（包括 n ），既是素数又是回文数的整数有多少个。

【输入格式】

一个大于 11 小于 1000 的整数 n 。

【输出格式】

11 到 n 之间的素数回文数个数。

【样例输入】

23

【样例输出】

1

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,cnt;
4. bool is_p(int x){
5.     if(x<2) return false;
6.     if(x==2) return true;
7.     for(int i=2;i*i<=x;i++){
8.         if(x%i==0) return false;
9.     }
10.    return true;
11.}
12.bool is_r(int x){
13.    int a[10],cnt=0;
14.    memset(a,0,sizeof(a));
15.    while(x>0){
16.        cnt++;
17.        a[cnt]=x%10;
18.        x/=10;
19.    }
20.    for(int i=1;i<=cnt/2;i++){
21.        if(a[i]!=a[cnt-i+1]) return false;
22.    }
23.    return true;
24.}
25.int main(){
26.    cin>>n;
27.    for(int i=1;i<=n;i++){
28.        if(is_p(i)&&is_r(i)) cnt++;
29.    }
30.    cout<<cnt;
31.    return 0;
32.}
```


练 46.1 求 π 的值

【题目描述】

根据公式：

$$\arctan x (x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots \text{ 和 } \pi = 6 \arctan x\left(\frac{1}{\sqrt{3}}\right)$$

定义函数 $\arctan x(x)$ ，求当最后一项小于 10^{-6} 时 π 的值。

【输入格式】

无

【输出格式】

π 的值。保留到小数点后 10 位。

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. double f(double x,int y){
4.     double ans=1;
5.     for(int i=1;i<=y;i++) ans*=x;
6.     return ans;
7. }
8. double arctanx(double x){
9.     double ans=0;
10.    for(int i=1;;i++){
11.        double fz=f(x,2*i-1);
12.        double fm=2*i-1;
13.        double n=fz/fm;
14.        if(n<0.000001) break;
15.        if(i%2==1) ans+=fz/fm;
16.        else ans-=fz/fm;
17.    }
18.    return ans;
19.}
20.int main(){
21.    double pi=6*arctanx(1.0/sqrt(3));
22.    printf("%.10lf",pi);
23.    return 0;
24.}
```

练 46.2 求丑数(趣味编程)

【题目描述】

我们把只包含因子 2、3 和 5 的数成为丑数。例如 6、8 都是丑数。例如 6、8 都是丑数，但 14 不是，因为它包含因子 7。习惯上，我们把 1 当作是第一个丑数。

先定义一个判断丑数的函数，利用它输出 1 ~ 100 之间所有的丑数，并统计出个数。

【输入格式】

无

【输出格式】

输出分为两部分：

第一部分，输出 1 ~ 100 之间所有的丑数，一行 10 个整数，每个整数占 6 个字符。

第二部分，输出 1 ~ 100 之间的丑数个数。

【样例输入】

无

【样例输出】

1	2	3	4	5	6	8	9	10	12
15	16	18	20	24	25	27	30	32	36
40	45	48	50	54	60	64	72	75	80
81	90	96	100						

个数： 34

【代码实现】

```
1. #include<iostream>
2. #include<iomanip>
3. using namespace std;
4. bool choushu(int n){ //求丑数
5.     while(n%2==0)    //包含因子2
6.         n/=2;
7.     while(n%3==0)    //包含因子3
8.         n/=3;
9.     while(n%5==0)    //包含因子5
10.        n/=5;
11.    return (n==1); //是否是丑数
12.}
13.int main(){
14.    int num=0;
15.    for(int i=1;i<=100;i++)
16.        if(choushu(i))
17.        {
18.            cout<<setw(6)<<i; //每个数占6 个字符
19.            num++;
20.            if(!(num%10)) cout<<endl; //换行
21.        }
22.    cout<<endl;
23.    cout<<"个数: "<<num<<endl; //输出个数
24.    return 0;
25.}
```

练 46.3 哥德巴赫猜想(趣味编程)

【题目描述】

哥德巴赫猜想是近代三大数学难题之一，即任何一个大于 2 的偶数，都可表示成两个素数之和。如 $4 = 2 + 2$ ， $6 = 3 + 3$ ， $8 = 3 + 5$ ， $10 = 3 + 7$ 。

请定义一个判断素数的函数，利用它验证 $4 \sim n$ 之间的偶数都能够分解为两个素数之和，其中 $n \geq 4$ 。

【输入格式】

一行一个整数，为 n 。若 $n < 4$ ，则重新读入。

【输出格式】

输出若干行，每行输出一个等式，表示 $4 \sim n$ 之间的偶数被分解为两个素数之和，若偶数不符合该整式，则输出“验证失败!”。

【样例输入】

10

【样例输出】

$4=2+2$

$6=3+3$

$8=3+5$

$10=3+7$

【代码实现】

```
1. #include<iostream>
2. using namespace std;
3.
4. int sushu(int x)           //求素数
5. {
6.     if(x<=1) return 0;     //0 或 1 不是素数
7.     if(x==2) return 1;     //2 是素数
8.     for(int i=2;i<=x-1;i++) //大于 2 的时候
9.         if(x%i==0) return 0;
10.    return 1;
11.}
12.
13.int main()
14.{
15.    int i,j,n;
16.    //cout<<"n=";
17.    do //当 n<4 时, 不输出, 只读入
18.    {
19.        cin>>n;
20.    }while(n<4);
21.    for(i=4;i<=n;i+=2)
22.    {
23.        for(j=2;j<i;j++)
24.            if(sushu(j)) //枚举 j 是素数
25.                if(sushu(i-j)) //判断 i-j 是不是素数
26.                {
27.                    cout<<i<<'='<<j<<'+ '<<i-j<<endl; //输出整式
28.                    break;
29.                }
30.        if(i==j) cout<<i<<"验证失败! "<<endl;
31.    }
32.    return 0;
33.}
```

练 46.4 第 N 个智慧数

【题目描述】

一个正整数如果能表示成两个正整数的平方差,则称这个数为一个“智慧数”比如 **16** 就等于 **5** 的平方减去 **3** 的平方,所以 **16** 就是一个智慧数,从 **1** 开始的自然数列中,将“智慧数”从小到大编号为 **1,2,3,...,n**。现输入一个正整数 **n**,输出第 **n** 个“智慧数”。

【输入格式】

输入仅包含一个正整数 **n** ($1 \leq n \leq 100$)。

【输出格式】

输出仅包含一个正整数,表示编号为 **n** 的“智慧数”。

【样例输入】

3

【样例输出】

7

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,cnt;
4. bool check(int x){
5.     for(int i=2;i<=x;i++)
6.         for(int j=1;j<=i-1;j++)
7.             if(i*i-j*j==x) return true;
8.     return false;
9. }
10.int main(){
11.    cin>>n;
12.    for(int i=1;;i++){
13.        if(check(i)) cnt++;
14.        if(cnt==n){
15.            cout<<i;
16.            break;
17.        }
18.    }
19.    return 0;
20.}
```

谢谢！