Ex1

源代码

```
1
    #include <stdio.h>
2
3
    int main() {
4
5
        int x, isPrime;
6
        printf("input a positive number: ");
7
        scanf("%d", &x);
8
9
        isPrime = 1;
10
        for (int i = 2; i < x / 2; i++) {
11
            if (x \% i == 0) {
                isPrime = 0;
12
13
                break;
14
            }
15
        }
16
        // 如果是素数,则输出本身并结束程序
17
        if (isPrime) {
18
            printf("%d", x);
19
20
            return 0;
21
        }
22
        // 如果不是,输出所有因数
23
24
        for (int i = 2; i \le x / 2; i++) {
25
            if (x \% i == 0) {
26
                printf("%d ", i);
27
            }
28
        }
29
        return 0;
30
31 }
```

运行结果

```
PS E:\Code_C++\hw06> cd "e:\Code_C+-\Ex1 }
input a positive number: 91
7 13
PS E:\Code_C++\hw06> cd "e:\Code_C+-\Ex1 }
input a positive number: 93
3 31
PS E:\Code_C++\hw06> cd "e:\Code_C+-\Ex1 }
input a positive number: 97
97
```

源代码

```
1 #include <stdio.h>
    #include <math.h>
 2
 3
4
   int main() {
6
        double x;
7
        printf("input a number: ");
8
        scanf("%1f", &x);
9
        double item = 1, s = 1;
10
        for (int i = 1; fabs(item) >= 0.000001; i++) {
11
            item = item * (0.5 - i + 1) * x / i;
12
13
            s += item;
14
        }
15
16
        printf("%1f", s);
17
18
        return 0;
19
    }
```

运行结果

```
PS E:\Code_C++\hw06> cd "e input a number: 0.5
1.224745
PS E:\Code_C++\hw06> cd "e input a number: -0.5
0.707107
```

Ex4

源代码

```
1 #include <stdio.h>
 2
 3
    int main() {
4
 5
        int n;
 6
        double s = 1;
 7
        printf("input a number: ");
        scanf("%d", &n);
 8
9
10
        for (double i = 1; i < n; i++) {
11
            s += (i + 1) / i;
12
        }
        printf("%1f", s);
13
14
15
       return 0;
16 }
```

Ex7

源代码

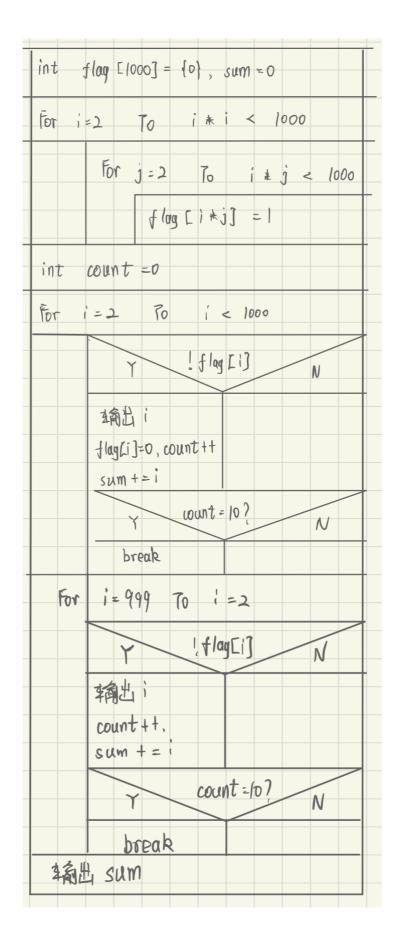
```
#include <stdio.h>
2
3
    int main() {
4
 5
        double x, sumPos = 0, sumNeg = 0;
        int countPos = 0, countNeg = 0;
 6
7
        for (int i = 0; i < 30; i++) {
8
            scanf("%1f", &x);
9
            if (x > 0) {
10
                sumPos += x;
11
                countPos++;
12
            }
13
            else {
14
                sumNeg += x;
15
                if (x != 0)
16
                    countNeg ++;
17
            }
18
        }
19
20
        printf("所有正数之和为: %lf\n", sumPos);
21
        printf("所有负数之和为: %lf\n", sumNeg);
        printf("所有数的绝对值之和为: %lf\n", sumPos - sumNeg);
22
23
        printf("正数的个数为: %d\n", countPos);
24
        printf("负数的个数为: %d\n", countNeg);
25
26
        return 0;
27
```

运行结果

```
PS E:\Code_C++\hw06> cd "e:\Code_C++\hw06\"; if ($?) { gcc Ex7.c 0 0 0 0 0 0 0 12 123 3123 3123 432 9978978 -123123 -12351 -857 -1241234 -3245234 -55 0.123 234.65 456.76 999.999 0.934567 0.87126 -0 -0 -0 -0 -0 0 所有正数之和为: 9987484.337827 所有负数之和为: -4622854.000000 所有数的绝对值之和为: 14610338.337827 正数的个数为: 12 负数的个数为: 6
```

Ex10

```
#include <stdio.h>
    #include <math.h>
 3
4
    int main() {
5
6
        int flag[1000] = {0}; // 0: prime 1: composite number
7
        int sum = 0;
8
9
        // 筛法找质数
10
        for (int i = 2; i < sqrt(1000); i++) {
            for (int j = 2; i * j <= 1000; j++) {
11
12
                flag[i * j] = 1;
13
            }
14
        }
15
        printf("最小素数: ");
16
17
        int count = 0;
18
        for (int i = 2; i \le 1000; i++) {
19
            if (!flag[i]) {
                                           // flag[i]为0即是素数,为1则是合数
20
                printf("%d, ", i);
                                           // 把flag[i]当作合数,保证后续不会被输出
21
                flag[i] = 1;
                count++;
22
23
                sum += i;
24
25
           if (count == 10)
26
                break;
27
        }
28
        printf("\n");
29
30
        printf("最大素数: ");
31
        count = 0;
32
        for (int i = 999; i >= 2; i--) {
33
            if (!flag[i]) {
34
                printf("%d, ", i);
35
                count++;
36
                sum += i;
37
            if (count == 10)
38
39
                break;
40
        }
        printf("\n");
41
42
        printf("素数之和: %d\n", sum);
43
44
45
        return 0;
46 }
```



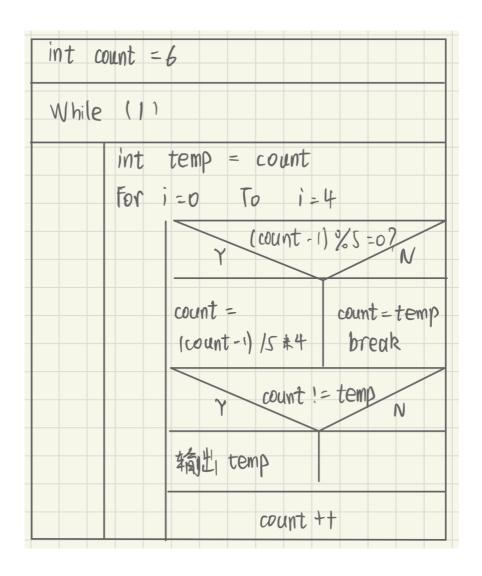
运行结果

```
PS E:\Code_C++> cd "e:\Code_C++\hw06\"; if ($?) { gcc Ex10.c -fexec-char:最小素数: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29,最大素数: 997, 991, 983, 977, 971, 967, 953, 947, 941, 937,素数之和: 9793
```

源代码

```
1 | #include <stdio.h>
2
3 int main() {
4
5
     int count = 6;
6
     while (1) {
        7
8
   该步骤能重复5次
9
       if ((count - 1) % 5 == 0) {
10
              count = (count - 1) / 5 * 4;
11
12
          else {
              count = temp; // 该数量不符合条件,将数量还原
13
14
             break;
15
       }
16
17
       if (count != temp) { // count不等于temp意味着count满足条件
   了
18
           printf("the minimum quantity is: %d", temp);
19
           break;
       }
20
   }
21
                          // 不满足条件,数量+1,继续尝试
        count++;
22
23
    return 0;
24
25 }
```

结构化流程



运行结果

```
PS E:\Code_C++\hw06> cd "e:\Code_C++\hw06\"; the minimum quantity is: 3121
```

Ex12

源代码

```
1
   #include <stdio.h>
2
   int main() {
3
4
       int a, b, c, d, e, f;
5
6
       // 0表示不去,1表示去
7
       for (a = 0; a \ll 1; a++) {
8
           for (b = 0; b \le 1; b++) {
9
10
              for (c = 0; c <= 1; c++) {
11
                  for (d = 0; d \le 1; d++) {
12
                      for (e = 0; e <= 1; e++) {
                          for (f = 0; f \le 1; f++) {
13
14
                             if ((c || (!c && !b)) // 若C不去则B不去
                             && (c + d == 1)
15
                                                   // C和D两人去一个
16
                             && (d == e)
                                                   // D和E要么都去,要么都不去
                             && (a + b + f == 2)  // A, B, F三人去两个
17
                             && (c + f != 2)
18
                                                  // C和F不能一起去
                             && (e + f != 0)) { // E和F至少去一个
19
```

```
20
                                 printf("去的人有: %c %c %c %c %c",
                                        a == 1 ? 'A' : ' ',
21
                                       b == 1 ? 'B' : ' ',
22
                                        c == 1 ? 'C' : ' ',
23
                                       d == 1 ? 'D' : ' ',
24
25
                                        e == 1 ? 'E' : ' ',
                                       f == 1 ? 'F' : ' ');
26
27
                             }
28
                         }
                     }
29
30
                 }
31
             }
32
          }
33
       }
34
35
      return 0;
36 }
```

结构化流程

int a, b, c, d, e, f
for a=0 To a=1
For b=0 To b=1
For c=0 To c=1
For d=0 To d=1
For e=o To e=1
For $f = 0$ To $f = 1$
C (!c &&!b) && c+d ==
RQ d = = e 7
&& a+b+f==2.
&& e+f!=0
输出为一

运行结果