

一、分工

范娇娇：数据成分 1、2 题及运算成分 1、2、3 题；设计排版

郑春瑞：数据成分第 3 题及审核；

合作完成：实验分析总结

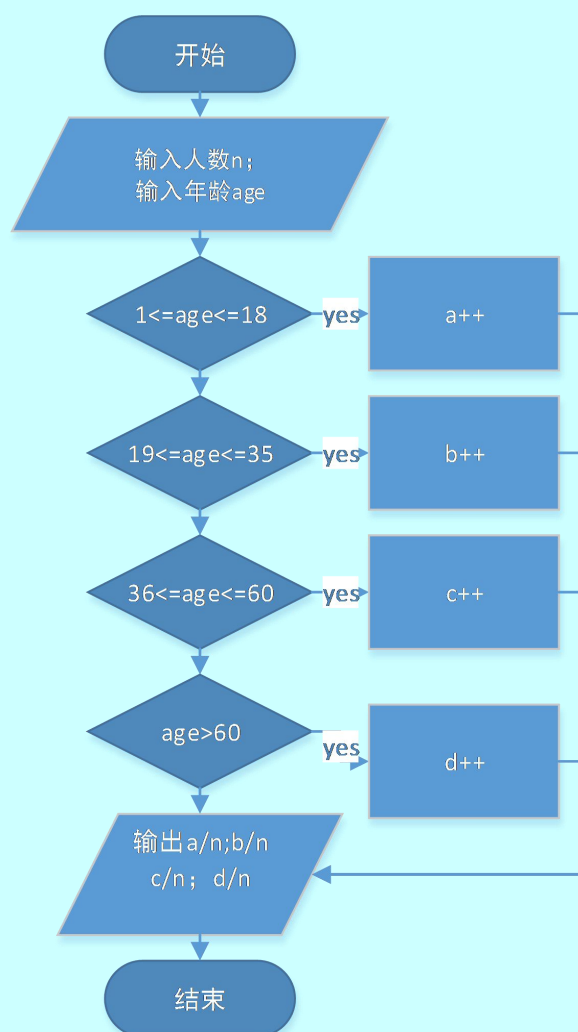
二、数据成分

1. 年龄与疾病

(1) 问题分析：统计某项疾病的获得是否与年龄有关。

(2) 解决方案：分不同年龄段，统计不同年龄段患病人数占总患病人数的百分比。

(3) 流程图及编程实现



```

1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4  int main()
5  {
6      int age;
7      double n;
8      double a=0,b=0,c=0,d=0;
9      cin>>n;
10     for (int i=0;i<n;i++)
11     {
12         cin>>age;
13         if(age>=1&&age<=18) a++;
14         if(age>=19&&age<=35) b++;
15         if(age>=36&&age<=60) c++;
16         if(age>60) d++;
17     }
18     cout<<"1-18: "<<fixed<<setprecision(2)<<a*100/n<<"%"<<endl;
19     cout<<"19-35: "<<fixed<<setprecision(2)<<b*100/n<<"%"<<endl;
20     cout<<"36-60: "<<fixed<<setprecision(2)<<c*100/n<<"%"<<endl;
21     cout<<"60-: "<<fixed<<setprecision(2)<<d*100/n<<"%"<<endl;
22     return 0;
23 }
24

```

样例输入

```

10
1  11  21  31  41  51  61  71  81  91

```

样例输出

```

1-18 : 20.00%
19-35 : 20.00%
30-60 : 20.00%
60- :40.00%

```

(4) 总结分析

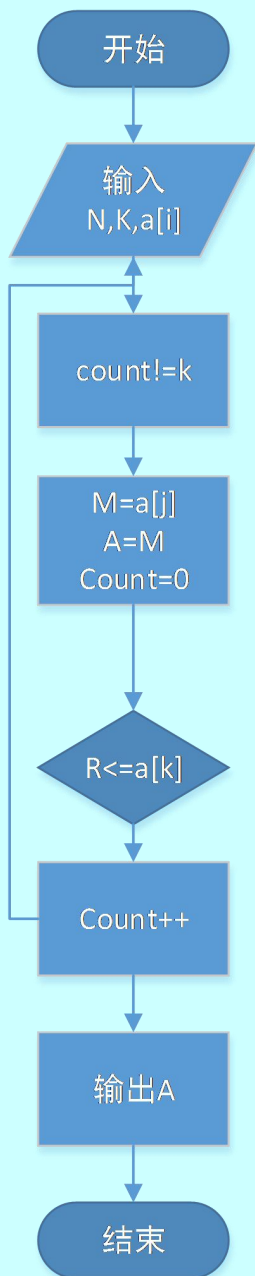
- 对于 $1 \leq \text{age} \leq 18$ 这样的表达，要学会用逻辑与或非来转换；
- 学习使用 `cout<<fixed<<setprecision(2)<<f` 来保留 f 后面的两位小数并进行推广使用。

2.找出第K大的数

(1) 问题分析：输入 N 个无序的正整数，在不排序的情况下找出第 K 大的数

(2) 解决方案：浏览所有的数，找出有 k-1 个数小于 A，则 A 为第 K 大

(3) 流程图及编程实现



```

1  #include<iostream>
2  using namespace std;
3  int main()
4  {
5      int N,K,A;
6      cin>>N>>K;
7      int a[10000];
8      int m=0, count=0;
9      for (int i=0;i<N;i++)
10     {
11         cin>>a[i];
12     }
13     for (int j=0;j<N&&count!=K;j++)
14     {
15         m=a[j];
16         A=m;
17         count=0;
18         for (int j=0;j<N;j++)
19             if (m<=a[j])
20             {
21                 count++;
22             }
23     }
24     cout<<A<<endl;
25     return 0;
26 }
  
```

样例输入

```
5
2
32 3 12 5 89
```

样例输出

```
32
```

(4) 总结分析

- a.思路清晰很重要
- b.注意变量的初始化

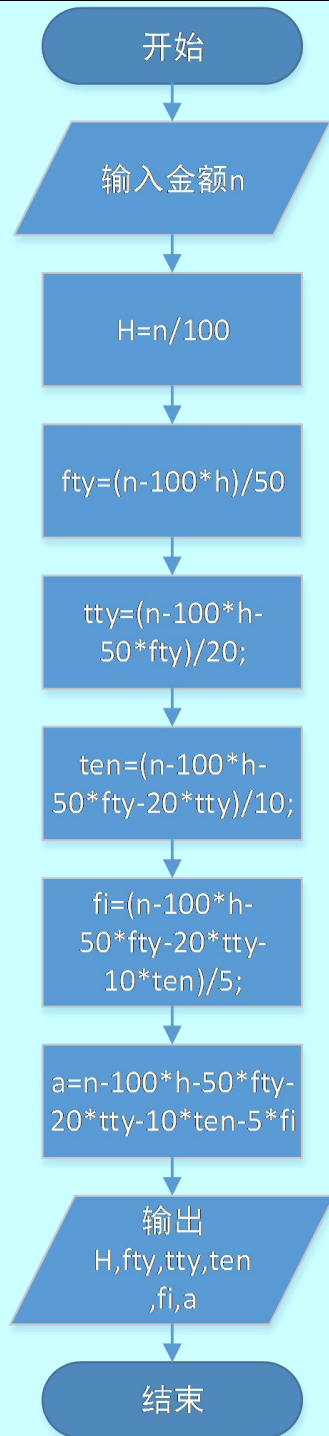
3.人民币支付

(1) 问题分析：输入制定金额，输出支付该金额的各种面额的人民币数量，尽量使用大面额的钞票。

(要求输入小于 1000 的正整数，输入面额显示 100 元，50 元，20 元，10 元，5 元，1 元)

(2) 解决方案：利用基本数学方法结合取整来解决。

(3) 流程图及编程实现



```

1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int n;
6      cin>>n;
7      int h,fty,tty,ten,fi,a;
8      h=n/100;
9      fty=(n-100*h)/50;
10     tty=(n-100*h-50*fty)/20;
11     ten=(n-100*h-50*fty-20*tty)/10;
12     fi=(n-100*h-50*fty-20*tty-10*ten)/5;
13     a=n-100*h-50*fty-20*tty-10*ten-5*fi;
14     cout<<h<<endl;
15     cout<<fty<<endl;
16     cout<<tty<<endl;
17     cout<<ten<<endl;
18     cout<<fi<<endl;
19     cout<<a<<endl;
20
21     return 0;
22 }
  
```

样例输入

735

样例输出

7
0
1
1
1
1
0

(4) .总结分析

注意对取整，取余的理解运用。

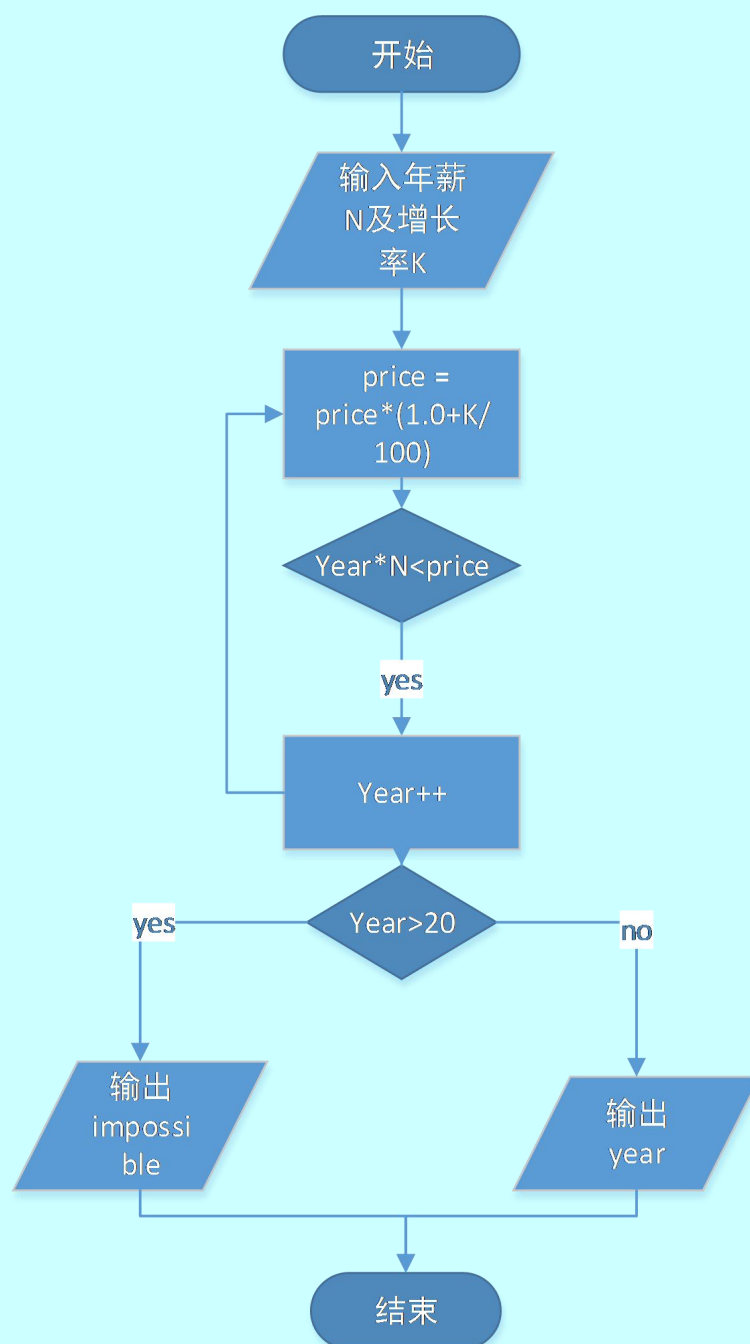
二、运算成分

1. 买房子

(1) 问题分析：某套房子现价 200 万，每年以百分之 K 增长；某人年薪 N 万为纯收入，输入 N、K，计算此人第几年才能买得起这套房子。

(2) 解决方案：利用基本数学算法和简单循环解决。

(3) 流程图及编程实现



```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int N;
6      double K;
7      double price = 200;
8      int year =1;
9      cin>>N>>K;
10     while( year*N<price )
11     {
12         price = price*(1.0+K/100);
13         year++;
14         if(year>20)
15             break;
16     }
17     if(year>20)
18         cout<<"Impossible"<<endl;
19     else if(year<=20)
20         cout<<year<<endl;
21     return 0;
22 }
```

样例输入

```
50 10
40 10
40 8
```

样例输出

```
8
Impossible
10
```

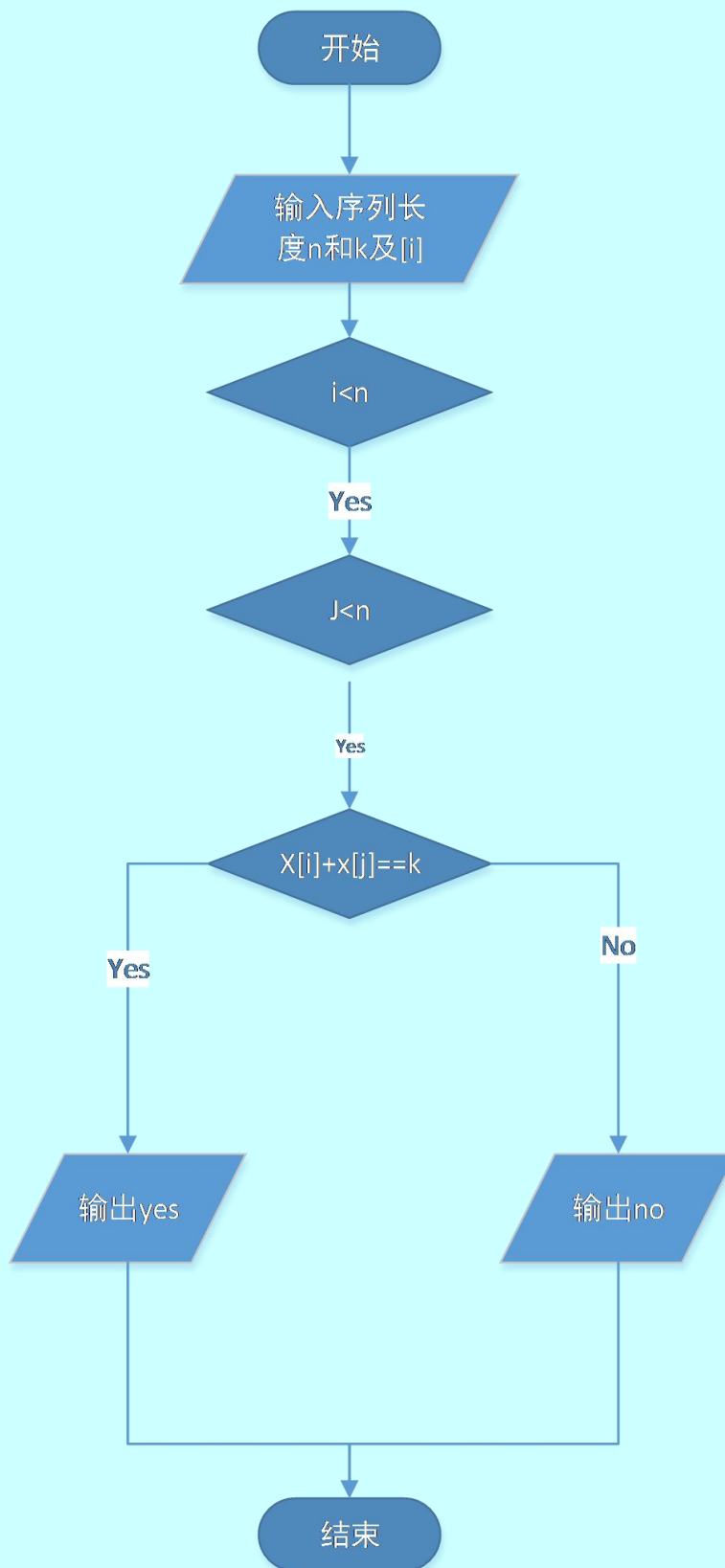
(4) 问题分析：注意对循环的正确使用。在 openjudge 上出现过超时问题，在于循环的使用不当。

2.找和为 K 的两个元素

(1) 问题分析：在一个长度为 n ($n < 1000$) 的整数序列中，判断是否存在某两个元素之和为 k 。

(2) 解决方案：利用循环，令每个数和其他数依次相加，判断是否存在符合题意的数。

(3) 流程图及编程实现




```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int n,k;
6      cin >> n >> k;
7      int x[1000];
8      for (int i = 0; i < n; i++)
9      {
10         cin >> x[i];
11     }
12     for (int i = 0; i < n-1; i++)
13     {
14         for (int j = i + 1; j < n; j++)
15         {
16             if (x[i] + x[j] == k)
17             {
18                 cout << "yes" << endl;
19                 return 0;
20             }
21         }
22     }
23     cout << "no" << endl;
24     return 0;
25 }
```

样例输入

9	10							
1	2	3	4	5	6	7	8	9

样例输出

yes

(4) 总结分析

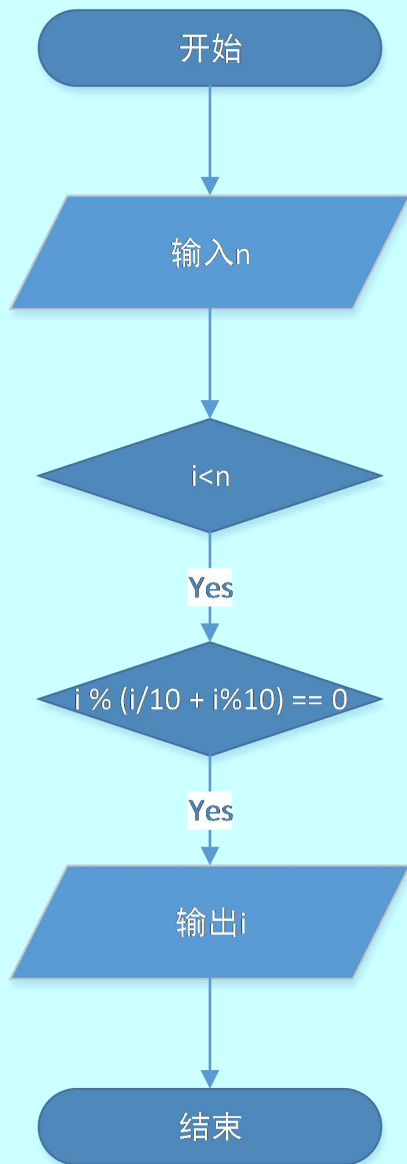
注意对{}的正确使用与对其用法的理解

3. 自除整数

(1) 问题分析： 对一个整数 n ，若其各个位数的数字相加的数 m 能整除 n ，则称 n 为自除整数。现需求出从 10 到 n ($n < 100$) 之间的自除整数。

(2) 解决方案： 根据题意结合数学公式和循环结构解决问题。

(3) 流程图及编程实现



```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int n,i;
6      cin >> n;
7      for(i = 10; i <= n; i++)
8          if(i % (i/10 + i%10) == 0)
9              cout << i << endl;
10     return 0;
11 }
```

样例输入

47

样例输出

```
10
12
18
20
21
24
27
30
36
40
42
45
```

（4）问题分析

需掌握并灵活运用取整和取余

三、实验总结分析

- 1.通过抄写程序和自己编写程序，发现并总结共同点，从而总结经验，有利于提升对 c 语言的认识和理解；
- 2.需更多的了解并学习运用 c++的格式化输入；
- 3.需更深层次地理解{}的作用并灵活运用；
- 4.程序的编写需注重其简洁性和可读性；（openjudge 有一项错误提醒 Time Limit Exceeded，可能原因包括使用了不必要的二次循环）
- 5.程序中注意条件语句的大小判定应灵活运用“&&，||，！”（注意此类错误表达：如 $25 < A < 89$ ）
- 6.需要灵活运用循环语句，区别 while，for 的不同；
- 7.注意“;”的用法；