

计算机与程序设计基础 (C)

实验课堂

第三讲 函数&数组



任务一 函数名的本质

```
1  #include <stdio.h>
2  int anyFun();
3  int main()
4  {
5      anyFun();
6      printf("%d\n", anyFun());
7      printf("%p\n", anyFun);
8      return 0;
9  }
10 int anyFun()
11 {
12     printf("In anyFunc()!\n");
13     return 1;
14 }
```

函数名代表函数的起始地址或入口地址

In anyFunc()!

In anyFunc()!

1

00000000000401593

任务二 函数声明

```
1  #include <stdio.h>
2
3  int main()
4  {
5      printf("hello\n");
6      test();
7      return 0;
8  }
9
10 Long test() //当没有函数原型声明时
11 {
12     return 1;
13 }
```

error: conflicting types for 'test'

不要这样写代码



```
16 #include <stdio.h>
17 Long test()
18 {
19     return 1;
20 }
21 int main()
22 {
23     printf("hello\n");
24     test();
25     return 0;
26 }
27 }
```

hello

任务三 函数调用参数类型不符

```
1  #include <stdio.h>
2  void funA(int, int);
3  int main()
4  {
5      funA(1.2, 3.8);
6      funA(1, 3);
7      return 0;
8  }
9
10 void funA(int a, int b)
11 {
12     double result1, result2;
13     result1=a;
14     result2=b;
15     printf("%f, %f\n", result1, result2);
16 }
```

1.000000, 3.000000
1.000000, 3.000000

不要这样写代码



任务四 函数返回类型不符

```
1  #include <stdio.h>
2  int TestFun(void)
3  {
4      return 9.7;
5  }
6  int main()
7  {
8      printf("%d\n", TestFun());
9      printf("%f\n", TestFun());
10 }
```

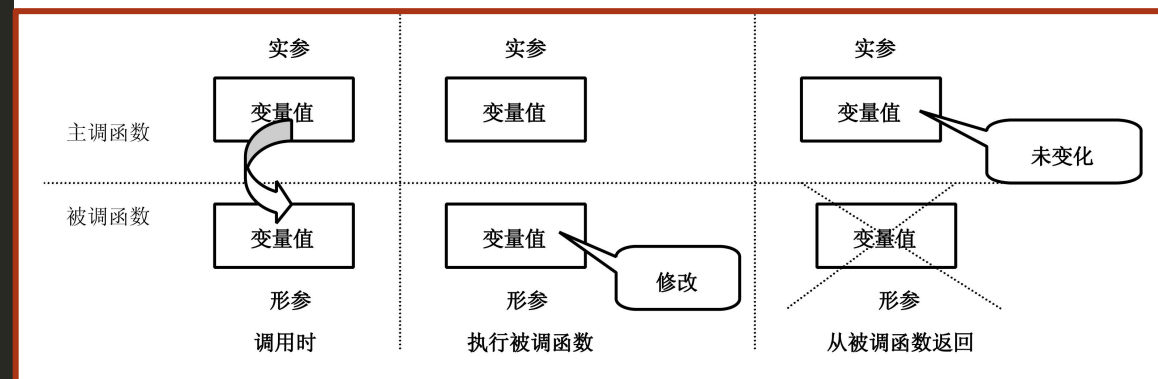
9
0.000000

不要这样写代码



任务五 内容值传递

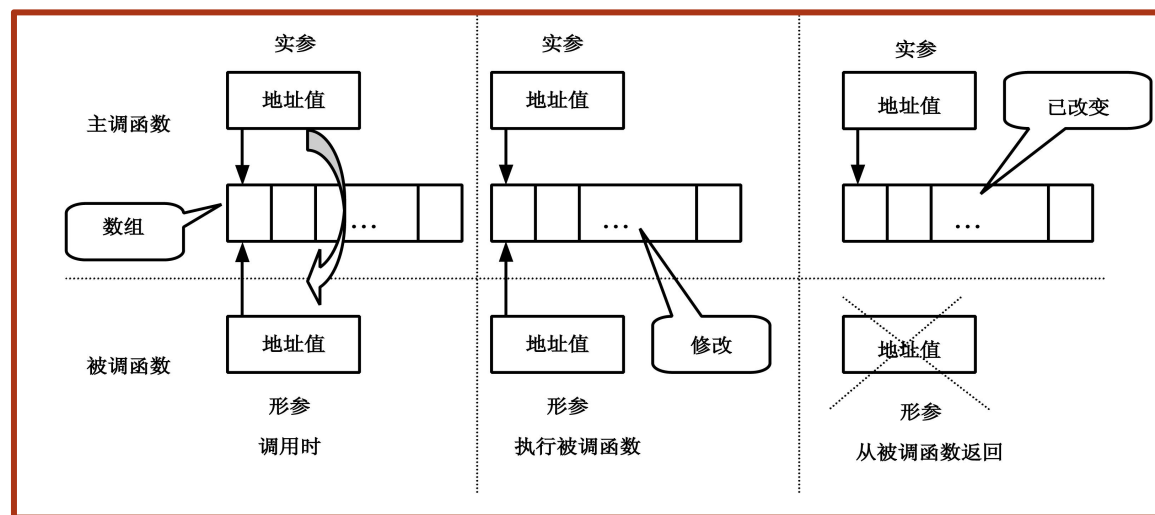
```
1 #include <stdio.h>
2 void main()
3 {
4     int a =3, b=5;
5     printf("main-->a @ %p, b @ %p\n", &a, &b);
6     printf("a=%d, b=%d\n", a, b);
7     swap( a, b);
8     printf("a=%d, b=%d\n", a, b);
9 }
10
11 void swap(int a,int b)
12 {
13     int temp;
14     temp =a;
15     a = b;
16     b = temp;
17     printf("swap--> a @ %p, b @ %p\n", &a, &b);
18 }
```



```
main-->a @ 000000000061FE1C, b @
000000000061FE18
a=3, b=5
swap--> a @ 000000000061FDF0, b @
000000000061FDF8
a=3, b=5
```

任务六 地址值传递

```
1  #include <stdio.h>
2  void swap(int *x, int *y);
3  void main()
4  {
5      int a, b, c;
6      scanf("%d%d%d", &a, &b, &c);
7      if (a > b)
8          swap(&a, &b);
9      if (a > c)
10         swap(&a, &c);
11         if (b > c)
12             swap(&b, &c);
13         printf("%d\t%d\t%d\n", a, b, c);
14     }
15
16     void swap(int *x, int *y)
17     {
18         int temp;
19         temp = *x;
20         *x = *y;
21         *y = temp;
22         return;
23     }
```



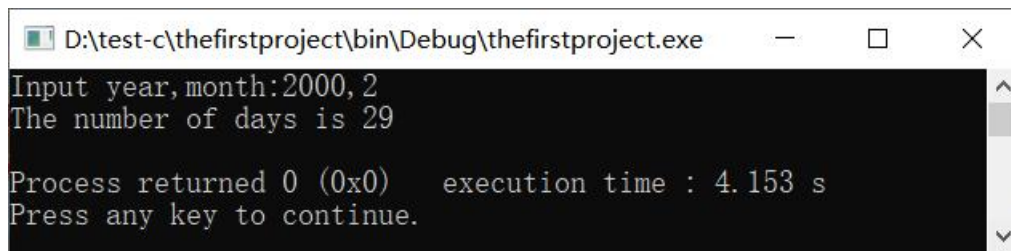
12	15	13
12	13	15

11	5	2
2	5	11

任务七 数组定义和初始化

编程实现显示用户输入的月份的天数

```
1  #include <stdio.h>
2  #define MONTHS 12
3  int main()
4  {
5      int days[2][MONTHS] = {{31,28,31,30,31,30,31,31,30,31,30,31},{31,29,31,30,31,30,31,31,30,31,30,31}};
6      int year, month;
7      do{
8          printf("Input year,month:");
9          scanf("%d,%d", &year, &month);
10     } while(month < 1 || month > 12); /* 处理不合法数据的输入 */
11     if (((year%4 == 0) && (year%100 != 0)) || (year%400 == 0))/*闰年*/
12         printf("The number of days is %d\n", days[1][month-1]);
13     else /*非闰年*/
14         printf("The number of days is %d\n", days[0][month-1]);
15     return 0;
16 }
```



```
D:\test-c\thefirstproject\bin\Debug\thefirstproject.exe
Input year,month:2000,2
The number of days is 29

Process returned 0 (0x0)   execution time : 4.153 s
Press any key to continue.
```


任务八 数组越界访问

```
1  #include <stdio.h>
2  int main()
3  {
4      int a = 1, c = 2, b[5] = {0}, i;
5      printf("%p, %p, %p\n", b, &c, &a);
6      for (i=0; i<=8; i++)
7      {
8          b[i] = i;
9          printf("%d  ", b[i]);
10     }
11     printf("\nc=%d, a=%d, i=%d\n", c, a, i);
12     return 0;
13 }
```

D:\tmp\c-coding\test\bin\Debug\test.exe

000000000061FE00, 000000000061FE14, 000000000061FE18
0 1 2 3 4 5 6 7 8
c=5, a=6, i=9

Process returned 0 (0x0) execution time : 0.250 s
Press any key to continue.

b[0]	0	40
b[1]	1	44
b[2]	2	48
b[3]	3	4c
b[4]	4	50
c	5	54
a	6	58
i	9	5c
b[8]	8	60
		64
	...	68
		6c

任务九 数组逆序

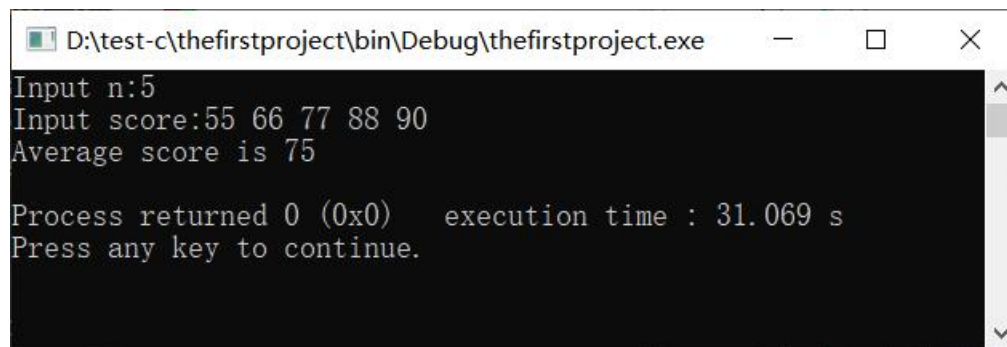
4
input the numbers:
2 5 8 1
now the numbers are:
1 8 5 2

```
1  #include <stdio.h>
2  void main( )
3  {
4      int a[100];
5      int i,j,n,temp;
6      scanf("%d",&n);        //输入整数个数
7      printf("input the numbers:\n");
8      for(i=0;i<n;i++)        //输入整数序列
9      {
10         scanf("%d",&a[i]);
11     }
12     /*将整数序列依次从首尾向中间交换元素，从而实现逆序排列*/
13     for(i=0,j=n-1; i<j; i++,j--)
14     {
15         temp=a[j];
16         a[j]=a[i];
17         a[i]=temp;
18     }
19     printf("now the numbers are:\n");
20     //输出重排后的整数序列
21     for(i=0;i<n;i++)
22     {
23         printf("%5d",a[i]);
24     }
25 }
```

任务十 向函数传递一维数组

```
1  #include <stdio.h>
2  #define N 40
3  int Average(int score[], int n);    /* Average()函数原型 */
4  void ReadScore(int score[], int n); /* ReadScore()函数原型 */
5  int main()
6  {
7      int score[N], aver, n;
8      printf("Input n:");
9      scanf("%d", &n);
10     ReadScore(score, n);           /* 数组名作为函数实参调用函数ReadScore() */
11     aver = Average(score, n);       /* 数组名作为函数实参调用函数Average() */
12     printf("Average score is %d\n", aver);
13     return 0;
14 }
15 /* 函数功能： 计算n个学生成绩的平均分 */
16 int Average(int score[], int n) /* Average()函数定义 */
17 {
18     int i, sum = 0;
19     for (i=0; i<n; i++)
20     {
21         sum += score[i];
22     }
23     return sum / n;
24 }
25 /* 函数功能： 输入n个学生的某门课成绩 */
26 void ReadScore(int score[], int n) /* ReadScore()函数定义 */
27 {
28     int i;
29     printf("Input score:");
30     for (i=0; i<n; i++)
31     {
32         scanf("%d", &score[i]);
33     }
34 }
```

编程实现计算用户输入分数的平均值



```
D:\test-c\thefirstproject\bin\Debug\thefirstproject.exe
Input n:5
Input score:55 66 77 88 90
Average score is 75

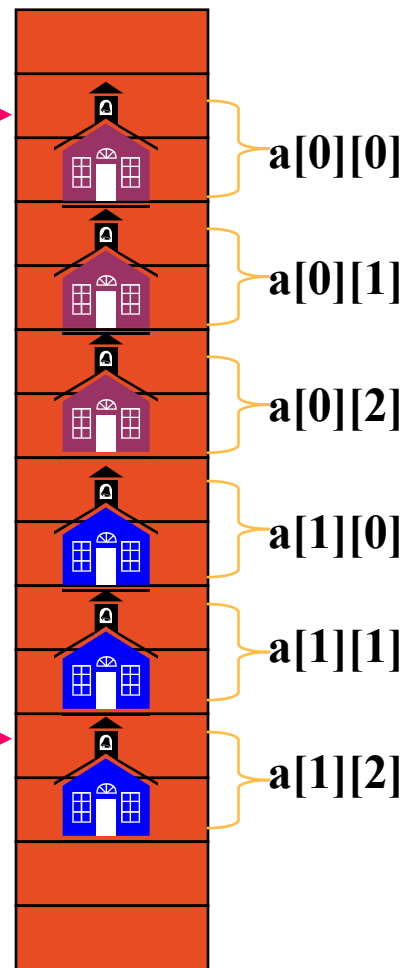
Process returned 0 (0x0)   execution time : 31.069 s
Press any key to continue.
```

向函数传递二维数组

实际传送的是数组第一个元素的地址

- 在声明二维数组形参时，不能省略数组第二维的长度（列数）
- 元素 $a[i][j]$ 在数组 a 中的位置是： $i * N + j$
- 元素地址： 首地址 + 偏移量

偏移 $1 * 3 + 2$



任务十一 向函数传递二维数组

计算学生的平均分

```
D:\test-c\thefirstproject\bin\Debug\thefirstproject.exe
Input student's ID and score as: MT EN PH:
50 60 70
70 80 90
90 90 90
100 90 80
88 78 68
student 0, adverage score=60.0
student 1, adverage score=80.0
student 2, adverage score=90.0
student 3, adverage score=90.0
student 4, adverage score=78.0

Process returned 0 (0x0)   execution time : 27.196 s
Press any key to continue.
```

```
1  #include <stdio.h>
2  #define STUD_N 20 /* 最多学生人数 */
3  #define COURSE_N 3 /* 考试科目数 */
4  void ReadScore(int score[][COURSE_N], int n);
5  void AverforStud(int score[][COURSE_N], float aver[], int n);
6
7  int main()
8  {
9      int score[STUD_N][COURSE_N], n;
10     float averS[STUD_N];
11     printf("Input the total number of the students(n<=20):");
12     scanf("%d", &n); /* 输入参加考试的学生人数 */
13     ReadScore(score, n); /* 读入n个学生的学号和成绩 */
14     AverforStud(score, averS, n); /* 计算每个学生的平均分 */
15     return 0;
16 }
17 /* 函数功能：输入n个学生的学号及其三门课的成绩 */
18 void ReadScore(int score[][COURSE_N], int n)
19 {
20     int i, j;
21     printf("Input student's ID and score as: MT EN PH:\n");
22     for (i=0; i<n; i++) /* 对所有学生进行循环 */
23     {
24         for (j=0; j<COURSE_N; j++) /* 对所有课程进行循环 */
25         {
26             scanf("%d", &score[i][j]); /* 输入每个学生的各门课成绩 */
27         }
28     }
29 }
30 /* 函数功能：计算每个学生的平均分 */
31 void AverforStud(int score[][COURSE_N], float aver[], int n)
32 {
33     int i, j;
34     int sum;
35     for (i=0; i<n; i++)
36     {
37         sum=0;
38         for (j=0; j<COURSE_N; j++) /* 对所有课程进行循环 */
39         {
40             sum = sum + score[i][j]; /* 计算第i个学生的总分 */
41         }
42         aver[i] = (float)sum / COURSE_N; /* 计算第i个学生的平均分 */
43         printf("student%2d, adverage score=%4.1f\n", i, aver[i]);
44     }
45 }
```

任务十二 函数返回数组

C 语言不允许函数返回一个完整的数组

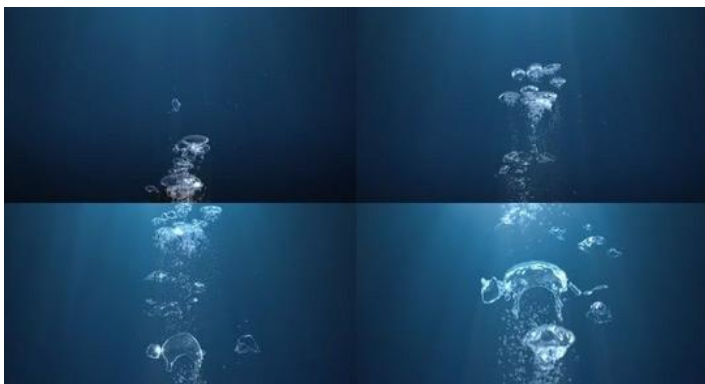
但可通过**数组名**来返回一个指向数组的指针

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  /* 要生成和返回随机数的函数 */
6  int * getRandom( )
7  {
8      static int r[10];
9      int i;
10     /* 设置种子 */
11     srand( (unsigned)time( NULL ) );
12     for ( i = 0; i < 10; ++i)
13     {
14         r[i] = rand();
15         printf( "r[%d] = %d\n", i, r[i]);
16     }
17     return r;
18 }
19
20 /* 要调用上面定义函数的主函数 */
21 int main ( )
22 {
23     /* 一个指向整数的指针 */
24     int *p;
25     int i;
26     p = getRandom();
27     for ( i = 0; i < 10; i++ )
28     {
29         printf( "p[%d] : %d\n", i, p[i]);
30     }
31     return 0;
32 }
33
```

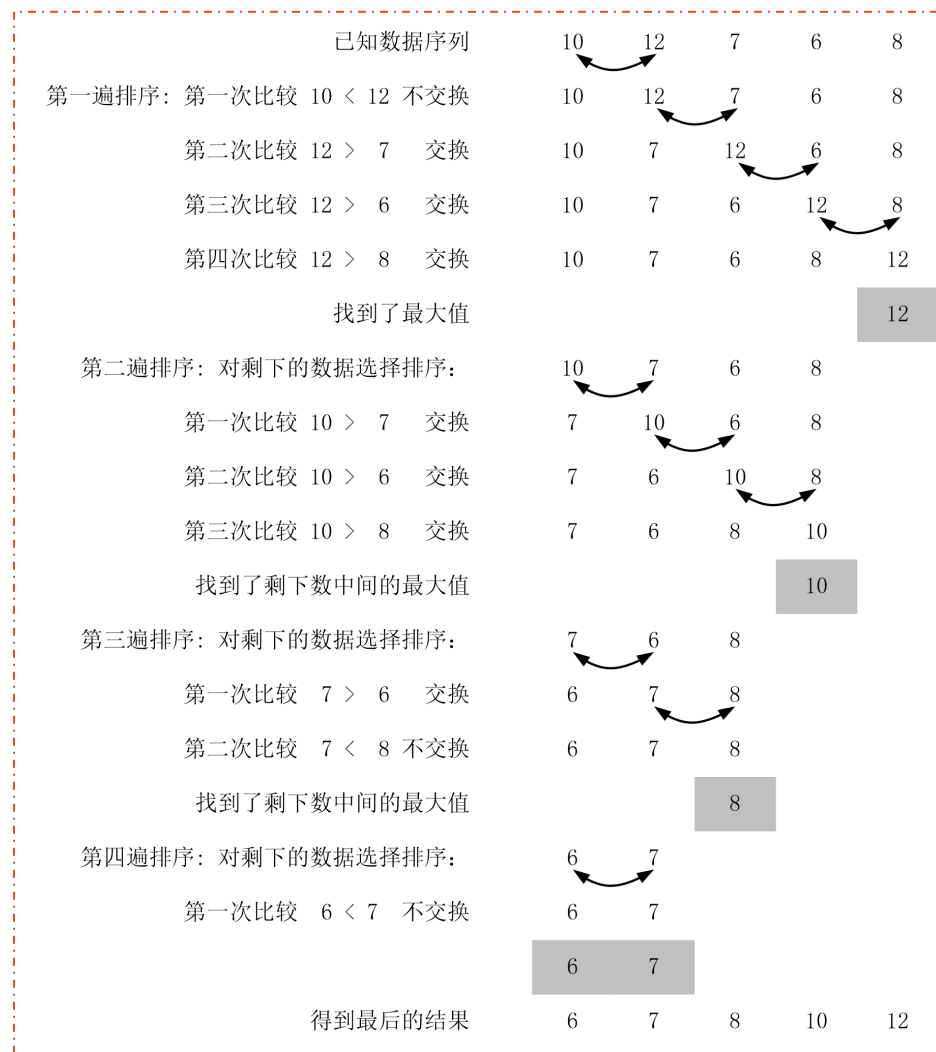

任务十三 冒泡法排序

思想: 首先进行第一遍排序, 第一个元素开始, 将相邻的两个数比较, 将小的数字放在前面, 大的放在后面, 依次往后比较, 比较到最后一组后, 产生一个最大值; 第二遍排序是除去产生的最大值, 对剩下的序列进行类似第一遍冒泡排序。依此类推, 进行N-1遍排序就能将N个数按从小到大排序; 同样的方法也能按从大到小排序。

对N个数要进行N-1遍排序, 每一遍比较次数随遍数的增加而减小, 比较的顺序是从前往后。



任务十三 冒泡法排序



```
1  #include <stdio.h>
2  void BubSort(int a[], int n);
3  void main()
4  {
5      int i,j,t;
6      int a[10];
7
8      for ( i=0 ;i < 10 ;i++)
9          scanf("%d",&a[i]);
10
11     BubSort(a, 10);
12
13     for ( i=0 ;i < 10 ;i++)        printf("%d\t", a[i]);
14 }
15
16 void BubSort(int a[], int n )
17 {
18     int i,j,t;
19
20     for ( i=0 ;i<n-1; i++)
21     {
22         for (j=0;j<n-1-i;j++)
23         {
24             if(a[j]>a[j+1])
25             {
26                 t = a[j];
27                 a[j] = a[j+1];
28                 a[j+1] = t;
29             }
30         }
31     }
32 }
```

```
D:\test-c\thefirstproject\bin\Debug\thefirstproject.exe
3 4 7 5 1 3 11 200 100 0
0      1      3      4      5      7      11      100      200
Process returned 4 (0x4)   execution time : 25.556 s
Press any key to continue.
```

任务十四 数组中插入元素

- 用函数编程实现在一个按升序排序的数组中查找x应该插入的位置，将x插入数组中，使得数组元素仍然按升序排列

Please input array size:

10

Please input array:

1 3 4 5 7 9 10 20 21 22

Please input x:

8

The changed array is:

1 3 4 5 7 8 9 10 20 21 22

任务十四 数组中插入元素

Please input array size:

10

Please input array:

1 3 4 5 7 9 10 20 21 22

Please input x:

8

The changed array is:

1 3 4 5 7 8 9 10 20 21 22

```
1  #include <stdio.h>
2  #define N 20 //插入x前数组最大元素个数
3  void Insert(int a[], int n, int x);
4  int main(void)
5  {
6      int a[N+1], x, i, n;
7      printf("Please input array size:\n");
8      scanf("%d", &n);
9      printf("Please input array:\n");
10     for (i=0; i<n; i++) {scanf("%d", &a[i]);}
11     printf("Please input x:\n");
12     scanf("%d", &x);
13     Insert(a,n,x);
14     printf("The changed array is:\n");
15     for (i=0; i<n+1; i++) {printf("%4d",a[i]);}
16     printf("\n");
17     return 0;
18 }
19
20 void Insert(int a[], int n, int x)
21 {
22     int i = 0, pos;
23     while(i<n&& x>a[i]) {i++;}
24     pos = i;
25     for (i=n-1; i>=pos; i--) //从尾部开始移动pos后所有元素
26     {
27         a[i+1]=a[i];
28     }
29     a[pos]=x;
30 }
```

任务十五 C99允许这样定义数组

i=0, @000000000061FD80, a[0]=4206072
i=1, @000000000061FD84, a[1]=0
i=2, @000000000061FD88, a[2]=80
i=3, @000000000061FD8C, a[3]=0
i=4, @000000000061FD90, a[4]=1
i=5, @000000000061FD94, a[5]=0
i=6, @000000000061FD98, a[6]=4200002
i=7, @000000000061FD9C, a[7]=0
i=8, @000000000061FDA0, a[8]=0
i=9, @000000000061FDA4, a[9]=0
i=10, @000000000061FDA8, a[10]=268501009
i=11, @000000000061FDAC, a[11]=0
i=12, @000000000061FDB0, a[12]=6752096
i=13, @000000000061FDB4, a[13]=0
i=14, @000000000061FDB8, a[14]=-1367549882
i=15, @000000000061FDBC, a[15]=32762
i=16, @000000000061FDC0, a[16]=8
i=17, @000000000061FDC4, a[17]=0
i=18, @000000000061FDC8, a[18]=0
i=19, @000000000061FDCC, a[19]=0

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int n=20;
6      int a[n];
7      for (int i=0; i<20;i++)
8      {
9          printf("i=%d, @%p, a[%d]=%d\n", i, &a[i], i, a[i]);
10     }
11     return 0;
12 }
```


任务十五 C99允许这样定义数组

4

array's size=16

i=0, at 000000000061FDC0, a[0]=50529027

i=1, at 000000000061FDC4, a[1]=50529027

i=2, at 000000000061FDC8, a[2]=50529027

i=3, at 000000000061FDCC, a[3]=50529027

i=4, at 000000000061FDD0, a[4]=4200064

i=5, at 000000000061FDD4, a[5]=4

i=6, at 000000000061FDD8, a[6]=6421952

二进制

八进制

十进制

十六进制

2进制

转换数字

00000011000000110000001100000011

二进制

八进制

十进制

十六进制

10进制

▼

转换结果

50529027

```
1  #include <stdio.h>
2
3  int main(void)
4  {
5      int n;
6      scanf("%d",&n);
7      //int a[n]={0};
8      int a[n];
9      printf("array's size=%d\n",sizeof(a));
10     memset(a, 3, sizeof(a));
11     for (int i=0; i<n+3;i++)
12     {
13         printf("i=%d, at %p, a[%d]=%d\n", i, &a[i], i, a[i]);
14     }
15     return 0;
16 }
```


任务十六 模拟骰子

- 模拟骰子的6000次投掷，编程统计并输出骰子的6个面各自出现的概率。



远离赌博陋习,倡导文明风尚

任务十六 模拟骰子

Face	Frequency
1	999
2	1013
3	1055
4	994
5	993
6	946

```
1  #include <stdlib.h>
2  #include <time.h>
3  #include <stdio.h>
4  int main(void)
5  {
6      int face, roll, frequency[7] = {0}; //7个元素
7      srand(time(NULL));
8      for (roll=1; roll<=6000; roll++)
9      {
10         face = rand()%6+1;
11         frequency[face]++;
12     }
13     printf("%4s%17s\n", "Face", "Frequency");
14     for(face=1; face<=6; face++)
15     {
16         printf("%4d%17d\n", face, frequency[face]);
17     }
18     return 0;
19 }
```

任务十七 兔子在哪里？

- 一只小兔子躲进了10个环形分布的洞中的一个。狼在第一个洞中没有找到兔子，就隔一个洞，到第三个洞去找；也没有找到，就隔两个洞，到第六个洞去找；以后每次多一个洞去找小兔子.....这样下去，如果一直找不到兔子，请问兔子可能在哪个洞中？

