

习 题

一、选择题

1.D 2. C 3. C 4. C 5.D 6. D 7. A 8. D 9. B 10. D
11.D 12.C 13. A 14.B 15.B 16.A 17.C 18.A

二、编程题

1.

```
#include "stdlib.h"
#include "time.h"
#define N 10
main()
{
    int i,j,a[N],temp;
    float Me;
    srand((unsigned int)time(0)); //srand 函数就用来初始化这个发生器
    for(i=0;i<N;i++){
        a[i]=rand()%90+10;
        printf("%d ",a[i]);
    }
    for(i=0;i<N-1;i++)
        for(j=i+1;j<N;j++)
            if(a[i]>a[j]){
                temp=a[i];a[i]=a[j];a[j]=temp;
            }
    if(N % 2 ==0)
    {
        Me=(a[N/2]+a[N/2-1])/2.0;
    }
    else
        Me=a[N/2];
    printf("\nmax=%d,min=%d,Me=%.1f",a[N-1],a[0],Me);
}
```

2.

```
main()
{ int i,j;
  for(i=1;i<=11;i++){
      for(j=1;j<=11-i;j++)
          printf(" ");
      for(j=1;j<=i;j++)
          printf("*");
      printf("\n");
  }
}
```

```
}
```

3. 已知 A 是一个 3X4 的矩阵，B 一个 4X3 的矩阵，编程求 AXB 得到的新矩阵 C，并输出 C 矩阵。

```
#define M 3
#define N 4
#define O 3
main()
{int i,j,k,c[M][O];
 int a[M][N]={ {1,1,1},{2,2,2},{3,3,3}};
 int b[M][N]={ {1,1,1},{2,2,2},{3,3,3}};
 for(i=0;i<M;i++)
     for(j=0;j<O;j++)
         for(k=0,c[i][j]=0;k<N;k++)
             c[i][j]+=a[i][k]*b[k][j];
 for(i=0;i<M;i++)
 {   for(j=0;j<O;j++)
     printf("%d  ",c[i][j]);
     printf("\n");
 }
}
```

4. 输入 5 个字符串，将其中最小的打印出来。（gets）

```
main()
{ char str[10],temp[10];
  int i;
  gets(temp);
  for(i=0;i<4;i++)
  { gets(str);
    if(strcmp(temp,str)>0)
      strcpy(temp,str);
  }
  printf("\n The least string is: %s",temp);
}
```

5. (附加题)

```
#include <stdio.h>
#define M 10000
#define N 1997
main()
{int i,j, a[M]={0};
 int t,sign = 0;
 a[0]=1;
 for(i=1;i<=N;i++){
     for(j=0;j<M;j++){
         a[j]=a[j]*i;
```

```

    }
    for(j=0;j<M;j++){
        t=a[j];
        a[j]=t % 10;
        if(t >= 10)a[j+1] += t/10%10;
        if(t >= 100)a[j+2] += t/100%10;
        if(t >= 1000)a[j+3] += t/1000%10;
        if(t >= 10000)a[j+4] += t/10000%10;
    }
}
for(i=M-1;i>=0;i--){
    if(sign==1||a[i]!=0){
        printf("%d",a[i]);
        sign=1;
    }
}
}
}

```

6. (附加题)

方法一：

```

#include <stdio.h>
#define N 10
void main()
{ int s[N],i=1,m=2,j,count=1;
  for (j=0;j<N;j++)
  { s[j]=j+1;
    printf("%d ",s[j]);
  }
  printf("\n");
  printf("%d ",s[i-1]);
  s[i-1]=0;
  while(count<N){
    j=0;
    do{
        if(s[i-1]!=0)j++;
        i++;
        if(i>N)i=1;
    }while(j<m);
    count++;
    if(i-2<0){printf("%d ",s[i-2+N]);s[i-2+N]=0;}
    else {printf("%d ",s[i-2]);s[i-2]=0;s[i-2]=0;}
  }
}

```

方法二：（方法一中的 m 与方法二中的 m 差 1）

考查一维数组的使用。

用一个数组 A 存放车站编号 (0~n-1)，这里用 0 表示编号为 n 的车站。当检查了某个车站后，将其编号置为 -1。

```
#include<stdio.h>
#define N 100
void main()
{int A[N],B[N],n,m,i,j,k=0,k1; /*k 记录检查车站的个数*/
  printf("n,i,m:");
  scanf("%d,%d,%d",&n,&i,&m);
  for(j=0;j<n;j++)
    A[j]=j;
  B[k++]=i;
  A[i]=-1;j=i;
  while(k<n) {
    k1=0; /*求间隔车站个数*/
    while(k1<=m){
      j=(j+1)%n; /*间隔车站是环形排列的*/
      if(A[j]!=-1)k1++;
    }
    B[k++]=A[j];
    if(B[k-1]==0)B[k-1]=n; /*B 数组从 0 开始，用 n 代替 0 */
    A[j]=-1;
  }
  printf("the result is:"); /*输出结果*/
  for(j=0;j<n;j++)
    printf("%3d",B[j]);
  printf("\n");
}
```