

C 语言必背 15 个经典程序

1、/*古典问题：有一对兔子，从出生后第 3 个月起每个月都生一对兔子，小兔子长到第三个月后每个月又生一对兔子，假如兔子都不死，问每个月的兔子总数为多少？

兔子的规律为数列 1,1,2,3,5,8,13,21....*/

主要涉及知识点：循环选择的嵌套、输出的格式控制（可转变方式用数组、递归的方式实现）

```
#include <stdio.h>
int main()
{
    long f1,f2;
    int i;
    f1=f2=1;
    for(i=1;i<=20;i++)
    {
        printf("%12ld %12ld",f1,f2);
        if(i%2==0)
            printf("\n");/*控制输出，每行四个*/
        f1=f1+f2; /*前两个月加起来赋值给第三个月*/
        f2=f1+f2; /*前两个月加起来赋值给第三个月*/
    }
    return 0;
}
```

2、/*输出 9*9 口诀。共 9 行 9 列，i 控制行，j 控制列。*/

主要涉及知识点：循环嵌套、输出的格式控制

```
#include <stdio.h>
int main()
{
    int i,j,result;
    for (i=1;i<10;i++)
    {
        for(j=1;j<=i;j++)
        {
            result=i*j;
            printf("%d*%d=%-3d",i,j,result); /*-3d 表示左对齐，占 3 位*/
        }
        printf("\n"); /*每一行后换行*/
    }
    return 0;
}
```

3、已知数组 a 中的元素已按由小到大顺序排列，以下程序的功能是将输入的一个数插入数组 a 中，插入后，数组 a 中的元素仍然由小到大顺序排列*/

主要涉及知识点：一维数组、插入算法、数组元素的移动

```
#include <stdio.h>
int main()
```

```

{
    int a[10]={0,12,17,20,25,28,30};          /*a[0]为工作单元，从 a[1]开始存放数据*/
    int x, i, j=6;                          /*j 为元素个数*/
    printf("Enter a number: ");
    scanf("%d",&x);
    a[0]=x;
    i=j;                                    /*从最后一个单元开始*/
    while(a[i]>x)
    {
        a[i+1]=a[i];
        i--;
    } /*将比 x 大的数往后移动一个位置*/
    a[++i]=x;
    j++;                                  /*插入 x 后元素总个数增加*/
    for(i=1;i<=j;i++)
        printf("%8d",a[i]);
    printf("\n");
    return 0;
}

```

4、/*下面程序的功能是将一个 4×4 的数组进行逆时针旋转 90 度后输出，要求原始数组的数据随机输入，新数组以 4 行 4 列的方式输出。*/

主要涉及知识点：二维数组的输入及输出、循环嵌套与二维数组的结合、输出的格式控制

```
#include <stdio.h>
```

```
int main()
```

```

{
    int a[4][4],b[4][4],i,j;          /*a 存放原始数组数据，b 存放旋转后数组数据*/
    printf("input 16 numbers: ");
    /*输入一组数据存放到数组 a 中，然后旋转存放到 b 数组中*/
    for(i=0;i<4;i++)
    {
        for(j=0;j<4;j++)
        {
            scanf("%d",&a[i][j]);
            printf("%6d",a[i][j]);
            b[3-j][i]=a[i][j];
        }
        printf("\n");
    }
    printf("array b:\n");
    for(i=0;i<4;i++)
    {
        for(j=0;j<4;j++)
            printf("%6d",b[i][j]);
        printf("\n");
    }
}

```

```

    }
    return 0;
}

```

5、/*编程打印直角杨辉三角形*/

主要涉及知识点：二维数组的输入及输出、循环嵌套与二维数组的结合、输出的格式控制

```

#include <stdio.h>
int main()
{
    int i,j,a[6][6];
    for(i=0;i<=5;i++)
    {
        a[i][i]=1;
        a[i][0]=1;
    }
    for(i=2;i<=5;i++)
        for(j=1;j<=i-1;j++)
            a[i][j]=a[i-1][j]+a[i-1][j-1];
    for(i=0;i<=5;i++)
    {
        for(j=0;j<=i;j++)
            printf("%4d",a[i][j]);
        printf("\n");
    }
    return 0;
}

```

6、/*通过键盘输入 3 名学生 4 门课程的成绩，分别求每个学生的平均成绩和每门课程的平均成绩。要求所有成绩均放入一个 4 行 5 列的数组中，输入时同一人数据间用空格,不同人用回车其中最后一列和最后一行分别放每个学生的平均成绩、每门课程的平均成绩及班级总平均分。*/

主要涉及知识点：二维数组的输入输出，及相应的处理

```

#include <stdio.h>
#include <stdlib.h>
int main()
{
    float a[4][5],sum1,sum2;
    int i,j;
    for(i=0;i<3;i++)
        for(j=0;j<4;j++)
            scanf("%f",&a[i][j]);
    for(i=0;i<3;i++)
    {
        sum1=0;
        for(j=0;j<4;j++)
            sum1+=a[i][j];
    }
}

```

```

        a[i][4]=sum1/4;
    }
    for(j=0;j<5;j++)
    {
        sum2=0;
        for(i=0;i<3;i++)
            sum2+=a[i][j];
        a[3][j]=sum2/3;
    }
    for(i=0;i<4;i++)
    {
        for(j=0;j<5;j++)
            printf("%6.2f",a[i][j]);
        printf("\n");
    }
    return 0;
}

```

7、/*输入一个字符串，判断其是否为回文。回文字符串是指从左到右读和从右到左读完全相同的字符串。*/

主要涉及知识点：字符数组的输入输出，及相应的处理

```

#include <stdio.h>
#include <string.h>
int main()
{
    char s[100];
    int i,j,n;
    printf("输入字符串: \n");
    gets(s);
    n=strlen(s);
    for(i=0,j=n-1;i<j;i++,j--)
        if(s[i]!=s[j]) break;
    if(i>=j) printf("是回文串\n");
    else printf("不是回文串\n");
    return 0;
}

```

8、/*下面程序的功能是从字符数组 a 中删除存放在其中的字符 c。*/

主要涉及知识点：字符数组的输入输出，及相应的处理

```

#include <stdio.h>
int main()
{
    char s[80],c;
    int j,k;
    printf("\nEnter a string: ");
    gets(s);

```

```

printf("\nEnter a character: ");
c=getchar( );
for(j=k=0;s[j]!='\0';j++)
    if(s[j]!=c)
        s[k++]=s[j];
s[k]='\0';
printf("\n%s",s);
return 0;
}

```

9、/*完善程序，实现将输入的字符串反序输出，如输入 windows 输出 swodniw。*/

主要涉及知识点：字符串的输入输出、字符串处理函数、指针的用法、文件相关操作

```

#include <stdio.h>
#include <string.h>
int main()
{
    char  c[200],c1;
    int i,j,k;
    printf("Enter a string: ");
    scanf("%s",c);
    k=strlen(c);
    for (i=0,j=k-1;i<k/2;i++,j--)
    {
        c1=c[i];
        c[i]=c[j];
        c[j]=c1;
    }
    printf("%s\n",c);
    return 0;
}

```

指针法、文件相关：

```

#include <stdio.h>
void invert(char *s)
{
    int i,j,k;
    char t;
    k=strlen(s);
    for(i=0,j=k-1;i<k/2;i++,j--)
    {
        t=*(s+i);
        *(s+i)=*(s+j);
        *(s+j)=t;
    }
}
int main()

```

```

{
    FILE *fp;
    char str[200],*p,i,j;
    if((fp=fopen("p9_2.out","w"))==NULL)
    {
        printf("cannot open the file\n");
        exit(0);
    }
    printf("input str:\n");
    gets(str);
    printf("\n%s",str);
    fprintf(fp,"%s",str);
    invert(str);
    printf("\n%s",str);
    fprintf(fp,"\n%s",str);
    fclose(fp);
    return 0;
}

```

10、编写函数 countpi，利用公式

$$\frac{\pi}{2} \approx 1 + \frac{1}{3} + \frac{1}{3} \times \frac{2}{5} + \frac{1}{3} \times \frac{2}{5} \times \frac{3}{7} + \frac{1}{3} \times \frac{2}{5} \times \frac{3}{7} \times \frac{4}{9} + \dots$$

计算 π 的近似值，当某一项的值小于 10^{-5} 时，认为达到精度要求，请完善函数。将结果显示在屏幕上并输出到文件 p7_3.out 中。

主要涉及知识点：函数、实数的比较、相关公式用程序实现的技巧、文件的操作

```

#include<stdio.h>
double countpi(double eps)                /*eps 为允许误差*/
{
    int m=1;
    double temp=1.0,s=0;
    while(temp>=eps)
    {
        s+=temp;
        temp=temp*m/(2*m+1);
        m++;
    }
    return(2*s);
}
int main()
{
    FILE *fp;
    double eps=1e-5,pi;
    if((fp=fopen("p7_3.out","w"))==NULL)
    {
        printf("cannot open the file\n");
    }
}

```

```

        exit(0);
    }
    pi= countpi(eps);
    printf("pi=%lf\n",pi);
    fprintf(fp,"pi=%lf\n",pi);
    fclose(fp);
    return 0;
}

```

11、/*冒泡排序，从小到大，排序后结果输出到屏幕及文件 myf2.out*/

主要涉及知识点：函数、排序算法、循环嵌套、文件相关操作

```

#include<stdio.h>
void fun(int a[],int n)
{
    int i,j,t;
    for(i=0;i<=n-1;i++)
        for(j=0;j<n-i;j++)
            if(a[j]>a[j+1])
            {
                t=a[j];
                a[j]=a[j+1];
                a[j+1]=t;
            }
}
int main()
{
    int a[10]={ 12,45,7,8,96,4,10,48,2,46},n=10,i;
    FILE *f;
    if((f=fopen("myf2.out","w"))==NULL)
        printf("open file myf2.out failed!\n");
    fun(a,10);
    for(i=0;i<10;i++)
    {
        printf("%4d",a[i]);
        fprintf(f,"%4d",a[i]);
    }
    printf("\n");
    fclose(f);
    return 0;
}

```

12、/*在一个字串 s1 中查找一子串 s2，若存在则返回子串在主串中的起始位置，不存在则返回-1。*/

主要涉及知识点：字符数组、查找算法、字符串处理函数

```

#include<stdio.h>
#include <string.h>

```

```

int search(char s1[],char s2[])
{
    int i=0,j,len=strlen(s2);
    while(s1[i])
    {
        for(j=0;j<len;j++)
            if(s1[i+j]!=s2[j]) break;
        if(j>=len) return i;
        else i++;
    }
    return -1;
}

```

```

int main()
{
    char s1[6]="thisis";
    char s2[5]="is";
    printf("%d\n",search(s1,s2));
    system("pause");
    return 0;
}

```

13、/*编写函数 replace(char *s,char c1,char c2)实现将 s 所指向的字符串中所有字符 c1 用 c2 替换，字符串、字符 c1 和 c2 均在主函数中输入，将原始字符串和替换后的字符串显示在屏幕上，并输出到文件 p10_2.out 中*/

主要涉及知识点：替换算法、函数、字符数组、文件操作

```
#include<stdio.h>
```

```
replace(char *s,char c1,char c2)
```

```

{
    while(*s!='\0')
    {
        if (*s==c1)
            *s=c2;
        s++;
    }
}

```

```
int main()
```

```

{
    FILE *fp;
    char str[100],a,b;
    if((fp=fopen("p10_2.out","w"))==NULL)
    {
        printf("cannot open the file\n");
        exit(0);
    }
}

```



```

printf("Enter a string:\n");
gets(str);
printf("Enter a&&b:\n");
scanf("%c %c",&a,&b);
printf("%s\n",str);
fprintf(fp,"%s\n",str);
replace(str,a,b);
printf("The new string is----%s\n",str);
fprintf(fp,"The new string is----%s\n",str);
fclose(fp);
return 0;
}

```

14、/*用指针变量输出结构体数组元素。*/

主要涉及知识点：指针与结构体、结构体数组的赋值、输出

```

#include <stdio.h>
struct student
{
    int num;
    char *name;
    char sex;
    int age;
}stu[5]={ { 1001,"lihua",'F',18},{ 1002,"liuxing",'M',19},{ 1003,"huangke",'F',19},{ 1004,"fengshou",
'F',19},{ 1005,"Wangming",'M',18} };
int main()
{
    int i;
    struct student *ps;
    printf("Num \tName\t\t\tSex\tAge\t\n");
    /*用指针变量输出结构体数组元素。*/
    for(ps=stu;ps<stu+5;ps++)
        printf("%d\t%-10s\t\t%c\t%d\t\n",ps->num,ps->name,ps->sex,ps->age);
    /*用数组下标法输出结构体数组元素学号和年龄。*/
    for(i=0;i<5;i++)
        printf("%d\t%d\t\n",stu[i].num,stu[i].age);
    return 0;
}

```

15、/*建立一个有三个结点的简单链表：*/

主要涉及知识点：结构体、链表相关操作

```

#include <stdio.h>
#define NULL 0
struct student
{
    int num;
    char *name;

```

```
    int age;
    struct student *next;
};
int main()
{
    struct student a,b,c,*head,*p;
    a.num=1001; a.name="lihua"; a.age=18; /* 对结点成员进行赋值 */
    b.num=1002; b.name="liuxing"; b.age=19;
    c.num=1003; c.name="huangke"; c.age=18;
    head=&a; /* 建立链表，a 为头结点 */
    a.next=&b;
    b.next=&c;
    c.next=NULL;
    p=head; /* 输出链表 */
    do{
        printf("%5d,%s,%3d\n",p->num,p->name,p->age);
        p=p->next;
    }while(p!=NULL);
    return 0;
}
```