

实验 1 参考程序

2. 实验内容

(4) 输入以下 C 语言程序并上机调试运行。程序的功能是从随意输入的两个数中找出较大的数并输出,如输入 8 10,观察运行结果。

```
#include<stdio.h>
int max(int a,int b);          /* 函数声明 */
int main()                    /* 主函数 */
{
    int x,y,z;                /* 变量说明 */
    printf("input two numbers:\n");
    scanf("%d%d",&x,&y);      /* 输入 x、y 值 */
    z=max(x,y);               /* 调用 max() 函数 */
    printf("max=%d\n",z);     /* 输出 */
    return 0;
}
int max(int a,int b)          /* 定义 max() 函数 */
{
    if(a>b)
        return a;
    else
        return b;            /* 把结果返回主调函数 */
}
```

答: 运行结果如下。

```
input two numbers:
8 10
max=10
```

实验 2 参考程序

2. 实验内容

(1) 对于以下程序:

```
#include<stdio.h>
int main()
{ short a,b;
  a=32767;
  b=a+1;
  printf("%d, %d\n", a,b);
  return 0;
}
```

① 请分析该程序的输出是什么?

答: 分析该程序的输出是 32767,32768。

② 输入并运行该程序,程序运行的结果是什么?

答: 该程序运行的结果是 32767,-32768。

③ 试对比分析结果和程序的运行结果一样吗? 如果不一样,为什么?

答: 分析的结果和程序运行的结果不一样,因为变量 a、b 定义为短整型,其能表示的最大数为 32767,加 1 后超出短整型能表示的数值范围。

(2) 对于以下程序:

```
#include<stdio.h>
int main()
{ float a,b;
  a=1234567.89e5;
  b=a+30;
  printf("%f\n",b);
  return 0;
}
```

① 请分析该程序的输出是什么?

答: 分析该程序的输出是 123456789030。

② 输入并运行该程序,程序运行的结果是什么?

答: 该程序运行的结果是 1234567890558.000000

③ 试分析你的分析结果和程序运行的结果一样吗? 如果不一样,为什么?

答: 分析的结果和程序运行的结果不一样,因为变量 a、b 定义为单精度浮点型数据,其有效数字为 6~7 位,超过的位数已不能准确地表示该数。

(3) 输入以下程序:

```
#include<stdio.h>
int main()
{ char c1,c2;
  c1='a';
  c2='b';
  printf("%c%c\n",c1,c2);
  return 0;
}
```

① 运行此程序。

答：此程序的运行结果为 a b。

② 在上面 printf 语句下面再增加一个 printf 语句：

```
printf("%d %d\n", c1, c2);
```

再运行,并分析结果。

答：此程序的运行结果为 a b

97 98。

③ 将第 3 行改为

```
int c1, c2;
```

再使之运行,并观察结果。

答：此程序的运行结果为 a b

97 98。

④ 再将第 4、5 行改为

```
c1=a;          /* 不用单撇号 */  
c2=b;
```

再使之运行,分析其运行结果。

答：程序出错。a、b 不用单撇号表示两个内存变量,而程序中并没有定义这两个变量,所以不能这样赋值。

⑤ 再将第 4、5 行改为

```
c1="a";        /* 用双撇号 */  
c2="b";
```

再使之运行,分析其运行结果。

答：程序出错。a、b 用双撇号不是表示字符常量,所以不能这样赋值。

⑥ 再将第 4、5 行改为

```
c1=300;        /* 用大于 255 的整数 */  
c2=400;
```

再使之运行,分析其运行结果。

答：程序出错。字符型数据和整型数据在一定条件下通用,这个条件是指在 ASCII 码的范围内,“c1=300;c2=400;”这样的赋值代码已经超出了这个范围。

(4) 求解如下表达式的值,试分析运行结果并编程验证,看自己的分析是否正确。

① $x+a\%3 * (\text{int})(x+y)\%2/4$, 其中 $x=2.5, a=7, y=4.7$ 。

分析运行结果为 2.75。

编程运行结果为 2.5。

如果结果不一致,试分析其原因。

答：在 C 语言中,两个整数相除,结果是一个整数,所以上面表达式中的 $1/4$,结果是 0,而不是 0.25。

② $(\text{float})(a+b)/2+(\text{int})x/(\text{int})y$, 其中 $a=2, b=3, x=3.5, y=2.5$ 。

分析运行结果为 3。

编程运行结果为 3.5。

如果结果不一致,试分析其原因。

答:在C语言中,两个整数相除,结果是一个整数,上面表达式中的 $(a+b)$ 结果虽然是整数,但前面有一个 (float) ,所以把整型转换成了浮点型,即 $5.0/2+3\%2=2.5+1=3.5$ 。

(5) 对于以下程序:

```
#include<stdio.h>
int main()
{ char c1='a',c2='b',c3='c',c4='\101',c5='\116';
  printf("a%c b%c\tc %c\tabc\n",c1,c2,c3);
  printf("\t\b%c%c\n",c4,c5);
  return 0;
}
```

① 分析该程序的输出是什么?

答:略(可能有多种)。

② 输入并运行该程序,程序的运行结果是什么?

答:程序运行结果是 aa bb c c abc\nAN。

③ 试分析你的分析结果和运行结果一样吗? 如果不一样,为什么?

答:“ $c4='\101',c5='\116'$;”语句中的 101 和 116 为八进制数,转换成十进制数分别为 65 和 78,所以 $\backslash101$ 表示字符 A, $\backslash116$ 表示字符 N。“ $\text{printf}("a\%c b\%c\tc \%c\tabc\n",c1,c2,c3);$ ”输出语句中的 3 个 $\%c$ 表示输出对应的 3 个字符: a,b,c,“ $\backslash t$ ”表示横向跳格的转义字符,“ $\backslash \backslash$ ”表示输出一个“ \backslash ”字符,其他按照原字符输出。“ $\text{printf}("\t\b\%c\%c\n",c4,c5);$ ”输出语句中“ $\backslash t$ ”表示横向跳格,“ $\backslash b$ ”表示退格。

(6) 输入并运行下面的程序:

```
#include<stdio.h>
int main()
{ int a,b;
  unsigned c,d;
  long e,f;
  a=100;
  b=-100;
  e=50000;
  f=32767;
  c=a;
  d=b;
  printf("%d,%d\n",a,b);
  printf("%u,%u\n",a,b);
  printf("%u,%u\n",c,d);
  c=a=e;
  d=b=f;
```

```

printf("%d,%d\n",a,b);
printf("%u,%u\n",c,d);
return 0;
}

```

请对照程序和运行结果分析:

① 将一个负整数赋给一个无符号的变量,会得到什么结果?

答:不能得到原来无符号变量的值。

② 将一个大于 32 767 的长整数赋给整型变量(假定所用的 C 语言系统分配给整型变量 2 字节),会得到什么结果?

答:将一个大于 32 767 的长整数赋值给整型变量,会造成溢出,数据丢失。

③ 将一个长整数赋给无符号变量,会得到什么结果(分别考虑长整数的值大于或等于 65 535 和小于 65 535 的情况)。

答:可能会溢出,也可能正常。

(7) 输入以下程序:

```

#include<stdio.h>
int main()
{ int i,j,m,n;
  i=8;
  j=10;
  m=++i;
  n=j++;
  printf("%d,%d,%d,%d\n",i,j,m,n);
  return 0;
}

```

运行程序,注意 i、j、m、n 各变量的值,分别做以下改动后再分析 i、j、m、n 各变量值的变化情况。

① 将第 6、7 行改为

```

m=i++;
n=++j;

```

再运行。

② 程序改为

```

#include<stdio.h>
int main()
{ int i,j;
  i=8;
  j=10;
  printf("%d,%d\n",i++,j++);
  return 0;
}

```

③ 在②的基础上,将 printf 语句改为

```
printf("%d,%d\n", ++i, ++j);
```

④ 再将 printf 语句改为

```
printf("%d,%d,%d,%d\n", i, j, i++, j++);
```

⑤ 程序改为

```
#include<stdio.h>
int main()
{ int i, j, m=0, n=0;
  i=8;
  j=10;
  m+=i++;n---j;
  printf("i=%d, j=%d, n=%d\n", i, j, m, n);
  return 0;
}
```

答: 以上程序结果的不同是由于自增或自减运算符引起的。自增或自减运算时,可以先将它们从源程序中取出来,如果++或--在后,则先把值放回去,然后再自增或自减;如果++或--在前,则先自增或自减,然后再把结果放进去。

实验3 参考程序

2. 实验内容

(1) 编写程序,把 560 分钟换算成用小时和分钟表示,然后输出。

答: 编写程序如下。

```
#include "stdio.h"
int main()
{ int a=560, b=60, c, d;
  c=a/b;
  d=a%b;
  printf("560 分钟=%d 小时%d 分钟", c, d);
}
```

运行结果:

```
560 分钟=9 小时 20 分钟
```

(2) 编写程序,读入 3 个双精度数,求出它们的平均值并保留此平均值小数点后一位数,对小数点后的第二位数进行四舍五入,最后输出结果。

答: 编写程序如下。

```
#include "stdio.h"
int main()
```

```

{ double a,b,c,ave;
  printf("Enter three numbers:");
  scanf("%lf%lf%lf",&a,&b,&c);
  ave=(a+b+c)/3;
  printf("(1)ave=%f\n",ave);
  ave=(int)(ave*10+0.5)/10.0;
  printf("(2)ave=%f\n",ave);
}

```

运行结果:

```

Enter three numbers: 5.7 3.9 8.6
(1) ave=6.066667
(2) ave=6.100000

```

实验 4 参考程序

2. 实验内容

(1) 编写一个程序,当给 x 输入小于 1 的数时,输出 $x-1$ 的值;当给 x 输入大于或等于 1 且小于 10 的数时,输出 $2(x-1)$ 的值;当给 x 输入大于或等于 10 的数时,输出 $3x-10$ 的值。用数学形式可以表示为:

$$y = \begin{cases} x-1 & x < 1 \\ 2(x-1) & 1 \leq x < 10 \\ 3x-10 & x \geq 10 \end{cases}$$

答:编写程序如下。

```

#include<stdio.h>
int main()
{
  float x,y;
  scanf("%f",&x);
  if(x<1)
    y=x-1;
  else
    if(x>1&&x<10)
      y=2*(x-1);
    else
      y=3*x-10;
  printf("%f\n",y);
}

```

运行结果:

```

100
290.000000

```

(2) 输入一个不多于 3 位数的正整数,编写程序实现下面的要求:

- ① 判断这个数是几位数。
- ② 分别输出这个数的每位数。
- ③ 按照输入的数的逆序输出另外一个数。如输入 123,输出 321。

答:编写程序如下。

```
#include<stdio.h>
int main()
{
    int num;
    int indiv, ten, hundred, place;
    printf("请输入一个整数(0~999): ");
    scanf("%d", &num);
    if(num>99)
        place=3;
    else
        if(num>9)
            place=2;
        else
            place=1;
    printf("位数: %d\n", place);
    printf("每位数字为: ");
    hundred= (int) num/100;
    ten= (int) (num-hundred * 100) /10;
    indiv= (int) (num-hundred * 100-ten * 10);
    switch(place)
    {
        case 3:printf("%d, %d, %d\n", hundred, ten, indiv);
                printf("逆序输出的数字为: ");
                printf("%d\n", indiv * 100+ten * 10+hundred);
                break;
        case 2:printf("%d, %d\n", ten, indiv);
                printf("逆序输出的数字为: ");
                printf("%d\n", indiv * 10+ten);
                break;
        case 1:printf("%d\n", indiv);
                printf("逆序输出的数字为: ");
                printf("%d\n", indiv);
                break;
    }
}
```

运行结果:

请输入一个整数(0~999): 123

位数: 3

每位数字为: 1, 2, 3

逆序输出的数字为：321

(3) 某企业发放的年终奖金根据企业的年度利润分段提成计算。年度利润小于或等于 10 万元部分,可以按照 10%提取年终奖金;年度利润大于 10 万元且小于或等于 20 万元部分,可以按照 7.5%提取年终奖金;年度利润大于 20 万元且小于或等于 40 万元部分,可以按照 5%提取年终奖金;年度利润大于 40 万元且小于或等于 60 万元部分,可以按照 3%提取年终奖金;年度利润大于 60 万元且小于或等于 100 万元部分,可以按照 1.5%提取年终奖金;超过 100 万元部分按照 1.5%提取年度奖金。分别用 if 语句和 switch 语句编写程序,输入年度利润总数,输出年终奖金总数。

答:

① 用 if 语句编写程序如下。

```
#include<stdio.h>
void main()
{
    long double i;
    double b,b1,b2,b4,b6,b10;
    b1=100000 * 0.1;
    b2=b1+100000 * 0.075;
    b4=b2+200000 * 0.05;
    b6=b4+200000 * 0.03;
    b10=b6+400000 * 0.015;
    printf("请输入利润 i: ");
    scanf("%lf",&i);
    if(i<=100000)
        b=i * 0.1;
    else
        if(i<=200000)
            b=b1+(i-100000) * 0.075;
        else
            if(i<=400000)
                b=b2+(i-200000) * 0.05;
            else
                if(i<=600000)
                    b=b4+(i-400000) * 0.03;
                else
                    if(i<=1000000)
                        b=b6+(i-600000) * 0.015;
                    else
                        b=b10+(i-1000000) * 0.1;
    printf("奖金是: %10.2f\n",b);
}
```

运行结果:

请输入利润 i: 200000

奖金是: 17500.00

② 用 switch 语句编写程序如下。

```
#include<stdio.h>
void main()
{
    long int i;
    double b,b1,b2,b4,b6,b10;
    int t;
    b1=100000 * 0.1;
    b2=b1+100000 * 0.075;
    b4=b2+200000 * 0.05;
    b6=b4+200000 * 0.03;
    b10=b6+400000 * 0.015;
    printf("请输入利润 i: ");
    scanf("%ld",&i);
    t=i/100000;
    if(t>10)
        t=10;
    switch(t)
    {
        case 0:b=i * 0.1; break;
        case 1:b=b1+ (i-100000) * 0.075; break;
        case 2:
        case 3:b=b2+ (i-200000) * 0.05; break;
        case 4:
        case 5:b=b4+ (i-400000) * 0.03; break;
        case 6:
        case 7:
        case 8:
        case 9:b=b6+ (i-600000) * 0.015; break;
        case 10:b=b10+ (i-1000000) * 0.01; break;
    }
    printf("奖金是: %10.2f\n",b);
}
```

运行结果:

请输入利润 i: 200000

奖金是: 17500.00

(4) 根据有关数据统计分析,某个人成年时的身高与其性别有关,并且还与其父亲的身高和母亲的身高有关,最后还与是否经常参加体育锻炼和是否有良好的卫生饮食习惯有关。

身高预测公式如下:

男性成年后时的身高=(父亲的身高+母亲的身高)×0.54(厘米)

女性成年后时的身高=(父亲的身高×0.923+母亲的身高)/2(厘米)