

20071116

一、 编写一个训练小学生练习九九表的程序。（练习使用随机函数）

说明：产生两个小于10的正整数，让学生回答乘积是多少，判断对错。

最小要求：如果回答正确，鼓励，回答不正确，则告知正确结果。

自行发挥：一道题允许学生多回答几次，然后再给出正确结果。出题的数目也可以用循环控制，如果能正确做出五道题，则结束训练，否则，继续出题。

//最小要求

```
void multiple()//九九乘法运算测试
{
    int a,b,answer;
    srand(time(0));
    a = rand() % 9 + 1; b = rand() % 9 + 1;
    cout << a << " * " << b << "=" << "?";
    cin >> answer;
    if (answer==a*b)
    { cout << "正确!"<<endl; }
    else
    { cout <<"错了"<< "正确结果是" <<a * b<<endl; }
}
```

//一次可多做几题的情况

```
void multiple()//九九乘法运算测试
{
    int a,b,answer;char ch;
    do
    { srand(time(0));
      a = rand() % 9 + 1; b = rand() % 9 + 1;
      cout << a << " * " << b << "=" << "?";
      cin >> answer;
```

```

if (answer==a * b)
{ cout << "正确!"<<endl; }
else
{ cout << "错了" << "正确结果是" <<a * b<< endl; }
cout << "continue ? (y or n) " << endl;
cin >> ch;
if (ch== 'n' ) break;
} while (1)
}

```

//记录正确率的情况

void multiple()//九九乘法运算测试

```

{
int a,b,answer, total=0, correct=0;char ch;
do
{ total++;
srand(time(0));
a = rand() % 9 + 1; b = rand() % 9 + 1;
cout << a << " * " << b << "=" << "?";
cin >> answer;
if (answer==a * b)
{ cout << "正确!"<<endl; correct ++; }
else
{ cout << "错了" << "正确结果是" <<a * b<< endl; }
cout << "continue ? (y or n) " << endl;
cin >> ch;
if (ch== 'n' ) break;
} while (1)
cout << "正确率是" <<correct*1.0/total * 100 "% " <<endl;
}

```

如果一道题可多次回答:

```
void multiple()//九九乘法运算测试
{
    int a,b,answer,total = 0,correct = 0,i;char ch;
    do
    { total++;
      srand(time(0));
      a = rand() % 9 + 1; b = rand() % 9 + 1;
      for ( i = 0; i < 5 i++)
      { cout << a << " * " << b << "=" << "?";
        cin >> answer;
        if (answer==a * b)
        { cout << "正确!"<<endl; correct ++; break; }
        else
        { cout<< "错了再回答一次";}
      }
      if (i>5)
      { cout << "正确结果是" <<a * b<< endl;}
      cout << "continue ? (y or n) " << endl;
      cin >> ch;
      if (ch== 'n' ) break;
    } while (1)
      cout << "正确率是" <<correct*1.0/total * 100 "% " <<endl;
}
```

二、使用 1 角，2 角，5 角硬币组成一元钱。要求输出：每种方法输出一行，每位数字占 5 位，要求用递归的方法完成。

```
-----
int c[4]={0,1,2,5};//第 1、2、3 种硬币的面值分别是 1 角、2 角、5 角
int p[4]={0};//每种面值硬币的个数初始定为 0 个
void f(int,int);//声明函数
void name2()
```

```

{
    cout<<"用以下枚数的 1 角, 2 角, 5 角硬币可以组成 1 元钱: "<<endl;
    f(3,10);//调用函数
    return;
}
void f(int k,int n)//计算由前 k 种硬币组成 n 角钱的所有方法
{
    if(k==0)
    {
        if(n==0)
            cout<<setw(5)<<p[1]<<setw(5)<<p[2]<<setw(5)<<p[3]<<endl;//结果输出, 每个数字占 5 位
            return;
        }
    for (int i=0;i<=int(n/c[k]);i++)//枚举第 k 种硬币的个数
    {
        p[k]=i;
        f(k-1,n-i*c[k]);//递归调用函数
    }
}

```

```

int coin[3],num=0;
void trycoin(int n)
{
    int i;
    if (n==3)
    {
        if ((coin[0]+2*coin[1]+5*coin[2])==10)
        {
            num++;
            cout << setw(4) << num;
            for (i=0; i<3; i++) cout << setw(4) << coin[i];
            cout << endl;
        }
    }
    else
    {
        for (i=0; i<=10; i++)
        {
            coin[v]=i;
            trycoin(v+1);
        }
    }
}

```

二、 用反复加法实现乘法, 写出递归函数, 例如: $A*B = A+A+A \dots +A$ (共 B 次)

```
int mystery(int a, int b)
{
    if (b==1)
        return a;
    else
        return (a + mystery(a, b-1));
}
```

四、ACM: 2944, 鼓励使用各种可以用的上的字符串函数, 见教材附录, 有相应的函数表。