四则运算1

2016-03-09 19:37

一：设计思路：

1.随机产生2个0到99之间的随机数，用作运算数；随机产生1个0到3之间的数，分别代表运算符+，-，\*,/。

2.if....else....语句控制输出的算式，for循环输出30道题。

3.当运算符为“/”号时，需要判断除数是否为0，如果为0，则循环次数加一。

4.为了保证随机种子的不同，需要用到srand(time(NULL))函数。

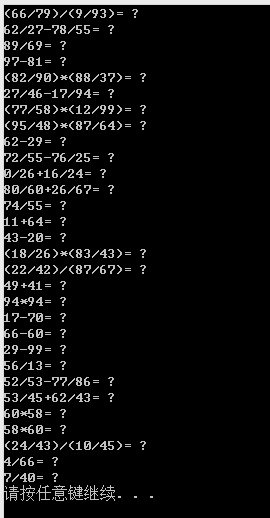
5.循环遍历已经出过的题，查看是否有重题。

二：源程序：

#include<iostream>  
using namespace std;  
#include<string>  
#include<time.h>

void main()  
{  
     srand(time(NULL));  
     string mark;  
     int key\_one;  
     int key\_two;  
     int dividend1,divider1,dividend2,divider2;  
     string result[30];  
  
     for(int i=0;i<30;i++)  
     {  
  
         key\_one=rand()%4;  
         key\_two=rand()%2;  
  
       if(key\_two==0)  
      {  
            dividend1=rand()%100;  
            divider1=rand()%100;  
         if(key\_one==0)  
        {  
            cout << dividend1 << "+" << divider1 << "= ?"<<endl;  
            result[i]=dividend1 + divider1;  
        }  
         else if(key\_one==1)  
        {  
            cout << dividend1 << "-" << divider1 << "= ?"<<endl;  
            result[i]=dividend1 - divider1;  
        }  
        else if(key\_one==2)  
       {  
            cout << dividend1 << "\*" << divider1 << "= ?"<<endl;  
            result[i]=dividend1 \* divider1;  
       }  
       else if(key\_one==3)  
      {  
         if(divider1==0)  
        {  
            i++;  
            break;  
        }  
         cout << dividend1 << "/" << divider1 << "= ?"<<endl;  
        result[i]="dividend/divider";  
     }  
      else  
         cout << "Please try again!" << endl;  
   }  
     if(key\_two==1)  
    {  
       dividend1=rand()%100;  
       divider1=rand()%100;  
       dividend2=rand()%100;  
       divider2=rand()%100;  
       if(key\_one==0)  
       {  
             cout << dividend1 << "/" <<divider1 << "+" <<dividend2<<"/" <<divider2<< "= ?"<<endl;  
        }  
       else if(key\_one==1)  
       {  
             cout << dividend1 << "/" <<divider1 << "-" <<dividend2<<"/" <<divider2<< "= ?"<<endl;  
       }  
      else if(key\_one==2)  
      {  
             cout <<"("<< dividend1 << "/" <<divider1 <<")"<< "\*" <<"("<<dividend2<<"/" <<divider2<<")"<< "= ?"<<endl;  
      }  
      else if(key\_one==3)  
     {  
            if(divider1==0||divider2==0)  
           {  
                i++;  
                break;  
           }  
           cout <<"("<< dividend1 << "/" <<divider1 <<")"<< "/" <<"("<<dividend2<<"/" <<divider2<<")"<< "= ?"<<endl;  
    }  
     else  
          cout << "Please try again!" << endl;  
    }  
  }  
}

三：结果截图：



四：上课未按时完成作业的原因：

没有认真思考问题，盲目的敲代码，思路不清晰，只是想一步写一步，一旦出错只能从头查找修改，浪费了大量的时间。

博客地址：http://www.cnblogs.com/java-meng/p/5259382.html