

# 计算概论期中题目备选

## 简答题

### 计算题

```
printf("%d", (int) (7.2/5)+3/2+010+0x1);
```

答案: 11 (考察类型转化, 八进制以及十六进制);

```
int x=0;  
x++?x=2:x==3;  
printf("%d",x);
```

答案: 1 (考察++号前置后置的区别, 条件运算符以及逻辑判断不改变变量数值);

```
char c='a';  
printf("%c",c+2);
```

答案: c (考察字符编码的连续性, 以及c语言中字符类型可以当成整数运算)

```
int flag=1;  
int t=0;  
switch(flag)  
{  
    default:  
        t++;  
    case -1:  
        t++;  
        break;  
    case 0:  
        t++;  
        break;  
}  
printf("%d",t);
```

答案: 2 (考察switch语句和break的用法)

## 代码理解

阅读下面代码, 写出程序的输出结果;

```
#include <stdio.h>  
#define LIMIT 6  
  
void fun(int n)
```

```

{
    int i=0;
    if (n<=LIMIT)
    {
        for(i=0;i<n;i++)
        {
            printf("*");
        }
        printf("\n");
    }
    if (n+1<=LIMIT)
    {
        fun(n+1);
    }
    if (n<LIMIT)
    {
        for(i=0;i<n;i++)
        {
            printf("*");
        }
        printf("\n");
    }
}

int main()
{
    fun(1);
    return 0;
}

```

答案：

```

*
**
***
****
*****
*****
*****
*****
****
***
**
*

```

(考察递归程序的理解)

## 编程题

### 求平方根

自己实现求平方根的函数 `double my_sqrt(double a)`，要求不使用标准库中的已有函数，例如 `sqrt`，`pow` 等。

(提示：求解方程  $x^2 - a = 0$ )

参考程序:

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

double my_sqrt(double a)
{
    // Negative input
    if(a<0)
    {
        printf("Error! Negative Input!");
        return -1;
    }
    // Zero input
    if(a<1e-10)
    {
        return 0;
    }
    // Otherwise
    double res=0;
    double x1=a, x2=a-1;
    double change=fabs(x1-x2);
    while(change>1e-5)
    {
        x2=x1;
        x1=x1-(x1*x1-a)/2.0/x1;
        change=fabs(x1-x2);
    }
    return x1;
}

int main()
{
    double a=12334;
    double x1=sqrt(a);
    double x2=my_sqrt(a);
    printf("sqrt(%f)=%e\n",a,x1);
    printf("my_sqrt(%f)=%e\n",a,x2);
    printf("Relative Error: %e\n",fabs(x1-x2)/x1);
    return 0;
}
```

## 丑数判断

实现函数 `int isUgly(int x)`，判断正整数 `x` 是不是丑数。丑数定义为除2,3,5外，没有其他素因子的正整数。

例如2,9,1024是丑数，而14不是。

参考代码

(循环或递归)

```

#include <stdio.h>

/*
// Recursion
int isUgly(int x)
{
    // Recursion exit
    if(x==1)
    {
        return 1;
    }
    // Recursion
    if(x%2==0)
    {
        return isUgly(x/2);
    }
    if(x%3==0)
    {
        return isUgly(x/3);
    }
    if(x%5==0)
    {
        return isUgly(x/5);
    }
    // Have other factors
    return 0;
}
*/

// Loop
int isUgly(int x)
{
    // Module all 2,3,5 factors
    while(x!=1)
    {
        if(x%2==0)
        {
            x=x/2;
            continue;
        }
        if(x%3==0)
        {
            x=x/3;
            continue;
        }
        if(x%5==0)
        {
            x=x/5;
            continue;
        }
        break;
    }
    // Check

```

```
    if(x==1)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}

int main()
{
    printf("Num \t isUgly\n");
    int x=6;
    printf("%d \t %d\n",x,isUgly(x));
    x=9;
    printf("%d \t %d\n",x,isUgly(x));
    x=14;
    printf("%d \t %d\n",x,isUgly(x));
    x=121;
    printf("%d \t %d\n",x,isUgly(x));
    x=1024;
    printf("%d \t %d\n",x,isUgly(x));
    return 0;
}
```