

Problem A. 中秋节快乐

难度	考点
1	字符串输出

直接复制粘贴hint就好，注意理解printf函数的用法。

Problem B. AC

难度	考点
1	字符串输出

简单的字符串输出。可以使用添加'\n'的方式，使得printf输出引号中的部分换行：

```
#include <stdio.h>
int main() {
    printf("\n
        ****                *****\n\
        *****            *****\n\
        **      **          **\n\
        **      **          **\n\
        *****            **\n\
        *****            **\n\
        **                **  **\n\
        **                **  **\n\
        **                **  **\n\
        **                **  *****\n\
        **                **  *****\n\");
    return 0;
}
```

上机中发现的问题

- 在部分ide的格式中，空格和*的宽度不等，AC字符画复制过去后可能发生错位，此时无需添加多余空格。

Problem C. Relu函数

难度	考点
1	数据输入输出、条件判断

本题考查如何进行输入和输出，以及如何使用 `if` 语句进行条件判断。

示例程序

```
#include <stdio.h>
int main()
{
    int x;
    scanf("%d", &x);

    if (x <= 0)
    {
        printf("0\n");
    }
    else
    {
        printf("%d\n", x);
    }

    return 0;
}
```

Problem D. 生日

难度	考点
1	数据输入输出、条件判断

本题考查如何进行输入和输出，以及如何使用 `if` 语句进行条件判断。

解法1：使用if-else语句

示例程序

```
#include <stdio.h>
int main() {
    int y,m,d;
    scanf("%d%d%d",&y,&m,&d);
    printf("My birthday is %d.",y);
    if(m<=9) printf("0"); //判断月份是否需要补0
    printf("%d.",m);
    if(d<=9) printf("0");//判断日期是否需要补0
    printf("%d\n",d);
    return 0;
}
```

解法二：格式符

示例程序

```
#include <stdio.h>
int main() {
    int y,m,d;
    scanf("%d%d%d",&y,&m,&d);
    printf("My birthday is %d.%02d.%02d",y,m,d);
    return 0;
}
```

Prblem E. 真·计算机

难度	考点
1	五种基本运算

题目分析

注意判断除数是否为0。

参考程序

```
#include<stdio.h>

int main() {
    long long a, b;

    scanf("%lld%lld", &a, &b);

    printf("%lld\n%lld\n%lld\n", a + b, a - b, a * b);
    if(0 == b) printf("[warning] division by zero [-wdiv-by-zero]");
    else printf("%lld\n%lld", a / b, a % b);

    return 0;
}
```

总结

把变量设为0，再以之为除数，这是很隐蔽的BUG。从现在起请有意识地学习调试方法和小技巧。

此外，上面写的是"if(0 == b)"，因为若你写成"if(b = 0)"不会报错，但写成"if(0 = b)"会。可以养成这个"常数 == 变量"的习惯，并思考一下为什么。

Problem F. 输出倒三角

难度	考点
1	字符的读入与输出

参考程序

```
#include <stdio.h>
int main() {
    char c; scanf("%c",&c);
    printf("%c%c%c%c%c\n",c,c,c,c,c);
    printf(" %c%c%c\n",c,c,c);
    printf("  %c\n",c);
}
```

Problem G. 三角形的周长

难度	考点
2	浮点数的读入与计算

参考程序

```
#include<stdio.h>
#include<math.h>
int main(){
    double x1,y1,x2,y2,x3,y3;
    scanf("%lf%lf",&x1,&y1);
    scanf("%lf%lf",&x2,&y2);
    scanf("%lf%lf",&x3,&y3);
    double d1=sqrt((x1-x2)*(x1-x2)+(y1-y2)*(y1-y2)); //计算边长
    double d2=sqrt((x1-x3)*(x1-x3)+(y1-y3)*(y1-y3));
    double d3=sqrt((x2-x3)*(x2-x3)+(y2-y3)*(y2-y3));
    printf("%.2f\n",d1+d2+d3);
}
```

Problem H. 打印进度条

难度	考点
2	输出字符图案

本题考查如何在屏幕上输出字符图案。

程序只能从上至下、从左至右依次输出字符，因此我们首先应将工作分为：

1. 输出竖线
2. 输出警号

3. 输出短横线
4. 输出竖线

由于进度条共有102列，除去两个竖线后共有100列，而警号共有 n 个，因此短横线应输出 $100 - n$ 个，使用循环语句即可。

示例程序

```
#include <stdio.h>

int main()
{
    int n;
    scanf("%d", &n);

    if (n < 1 || n > 100)
    {
        printf("wake up!");
        return 0;
        // 在main函数中使用return 0语句可立刻退出程序
    }

    putchar('|');
    for (int i = 0; i < n; i++)
    {
        putchar('#');
    }
    for (int i = 0; i < 100 - n; i++)
    {
        putchar('-');
    }
    putchar('|');

    return 0;
}
```

Problem I. 有向立方和

难度	考点
3	不确定多组数据的读入与计算

要点一：不确定的多组输入

```
while (scanf("%d", &n) != EOF) {
    // do something
}
```

初学者可以先记住这种输入方式

要点二：判断数字的奇偶性

```
if (n % 2 == 0) {  
    // do somethong  
} else {  
    // do something  
}
```

对2取模，如果模为0则为偶数，否则为奇数

要点三：数据范围

观察数据的最大范围，最多为1000组1000，此时结果数量级为 $1e12$ ，超过 `int` 的表示范围，因此应当使用

`long long`

示例程序

```
#include <stdio.h>  
  
int main() {  
    long long sum = 0;  
    int n;  
    while (scanf("%d", &n) != EOF) {  
        if (n % 2 == 0) {  
            sum += n * n * n;  
        } else {  
            sum -= n * n * n;  
        }  
    }  
    printf("%lld", sum);  
    return 0;  
}
```

Problem J. 零花钱

零花钱

难度	考点
2	循环，模拟（或数学）

题目分析

注意15块钱是个分界点，需要加入判断。

由题目条件，可以把它看成金字塔求和的问题。第一行一个数1，第二行两个数2.....所以可以先判断出k属于哪一行，之后求和。

示例程序

```
#include<stdio.h>

int main()
{
    int i, k, sum = 0;
    scanf("%d", &k);
    for (i = 1; (i * (i + 1) / 2) <= k && i < 15; i++); //k天所在的“行数”
    for (int j = 1; j <= i - 1; j++) { //把k所在行之前的所有行的零花钱加起来
        sum += j * j;
    }
    sum += (k - (i * (i - 1) / 2)) * i; //把k所在行的零花钱加起来
    printf("%d", sum);
    return 0;
}
```