

# ·用整型变量标记状态的方法实现替换goto语句

这是计卓1801陈柏余小萌新对书上一点疑惑的学习

## 原文

goto语句不是必需的语言成分。因为用goto语句实现的任何控制转移，都可以通过循环语句、if语句和其他转移语句的适当配合，以及*用整型变量标记状态的方法实现*。（课本P94页）

## 问题

- 对于“用整型变量标记状态的方法实现”不是很理解，询问老师后，得知是用诸如用-1,0,1来表示程序运行的状态来控制循环条件的判断
- 使用goto语句的原程序：课本P92页程序4.20

## 使用整型变量标记状态的方法实现替换goto语句

```
#include <stdio.h>
int main (void)
{
    double x,y;
    char op;
    int state = -1;                                /* 定义初始状态为-1，表示开始计算 */

    while (state!=-1){
        printf("input arithmetic expression:\n");
        scanf("%lf",&x);
        while((op = getchar()) != '\n'){
            state =0;                                /* 改状态值为0，输入y */
            while (state == 0){
                scanf("%lf",&y);
                switch(op){
                    case '+': x+=y;state =1; break;    /* 成功计算，改状态为1，退出第三层while，重新判断第二层while循环输入新的运算符 */
                    case '-': x-=y;state =1; break;    /* 成功计算，改状态为1，退出第三层while，重新判断第二层while循环输入新的运算符 */
                    case '*': x*=y;state =1; break;    /* 成功计算，改状态为1，退出第三层while，重新判断第二层while循环输入新的运算符 */
                    case '/':
                        if (y) {
                            x/=y;
                            state =1;                /* 成功计算，改状态为1，退出第三层while，重新判断第二层while循环输入新的运算符 */
                        }else {
                            printf("divisor is zero,input divisor again!\n");
                            state =0;                /* 除数为零，改状态值为0，重新输入y */
                        }
                        break;
                    default:
                        printf("illegai operator,input arithmetic expression again!\n");
                        state =-1;                    /* 输入非法运算符，该状态值为-1，重新开始计算 */
                }
            }
        }
    }
    printf("%lf\n",x);
    return 0;
}
```