

第五次实验报告

实验目的：

完成三道必做题、三道选做题：

必做题一：

编写程序，输入三个整数，判断能否构成三角形，若能，则输出三角形类型，若为等边三角形输出“Equilateral triangle”，若为等腰三角形输出“Isosceles triangle”，若为一般三角形输出“Triangle”，否则输出“Not triangle”。

(1) 输入：三个整数 xyz，用空格隔开。

(2) 输出：判断结果。

必做题二：

依次输入运算符@（+*/四种之一）和整数 a、b、c、d，计算(a/b)@(c/d)或计算(a@b@c@d)（本次实验选择输出 a@b@c@d）并输出结果的值。仅给出计算结果，不需要显示表达式。当输入的 b 或 d 出现 0 时，输出错误提示“input error”；当输入运算符为‘/’且输入的 c 为 0 时，同样提示“input error”。

(1) 输入：运算符@和整数 a、b、c、d，用空格隔开；

(2) 输出：计算结果或“input error”。

必做题三：

给出一个不多于 5 位的正整数。要求：(1) 求出它是几位数；(2) 分别打印出每一位数字；(3) 按照逆向打印各位数字；(4) 最少设计 10 个不同数据来测试运行结果。

选做题一：

将数字时间转化成英文表示，规则如下：

时间用时 h 和分 m 表示，如果 m 为 0，则将时读出来，然后加上 o' clock；如果 m 不为 0，则将时读出来，然后将分读出来。

时和分的读法使用的时英文数字的读法，对于大于 20 小于 60 的数字，首先读整十数，再加上个位数。

输入格式：输入以 Tab 分割两个非负整数 h 和 m 表示时和分。h 小于 24，m 小于 60。

输出格式：输出时间时刻的英文，单词之间以空格分割。

选做题二：

有四个圆塔，圆心分别为 (2, 2)、(-2, 2)、(-2, -2)、(2, -2)，圆半径均为 1，高度分别为 10m、9m、8m、7m，塔外无建筑物，今输入任意一点坐标，求该点建筑物高度（塔为圆柱体，塔以外输出 0）。

(1) 输入：点的横纵坐标 x、y，用空格隔开；

(2) 输出：点的高度。

选做题三：

输入一个整数 N，整数数值不超过 int 型的范围，统计 N 各位上出现数字 5、6、7 的次数。

(1) 输入：整数 N。

(2) 输出：N 各位上出现数字 5、6、7 的次数，用空格隔开。

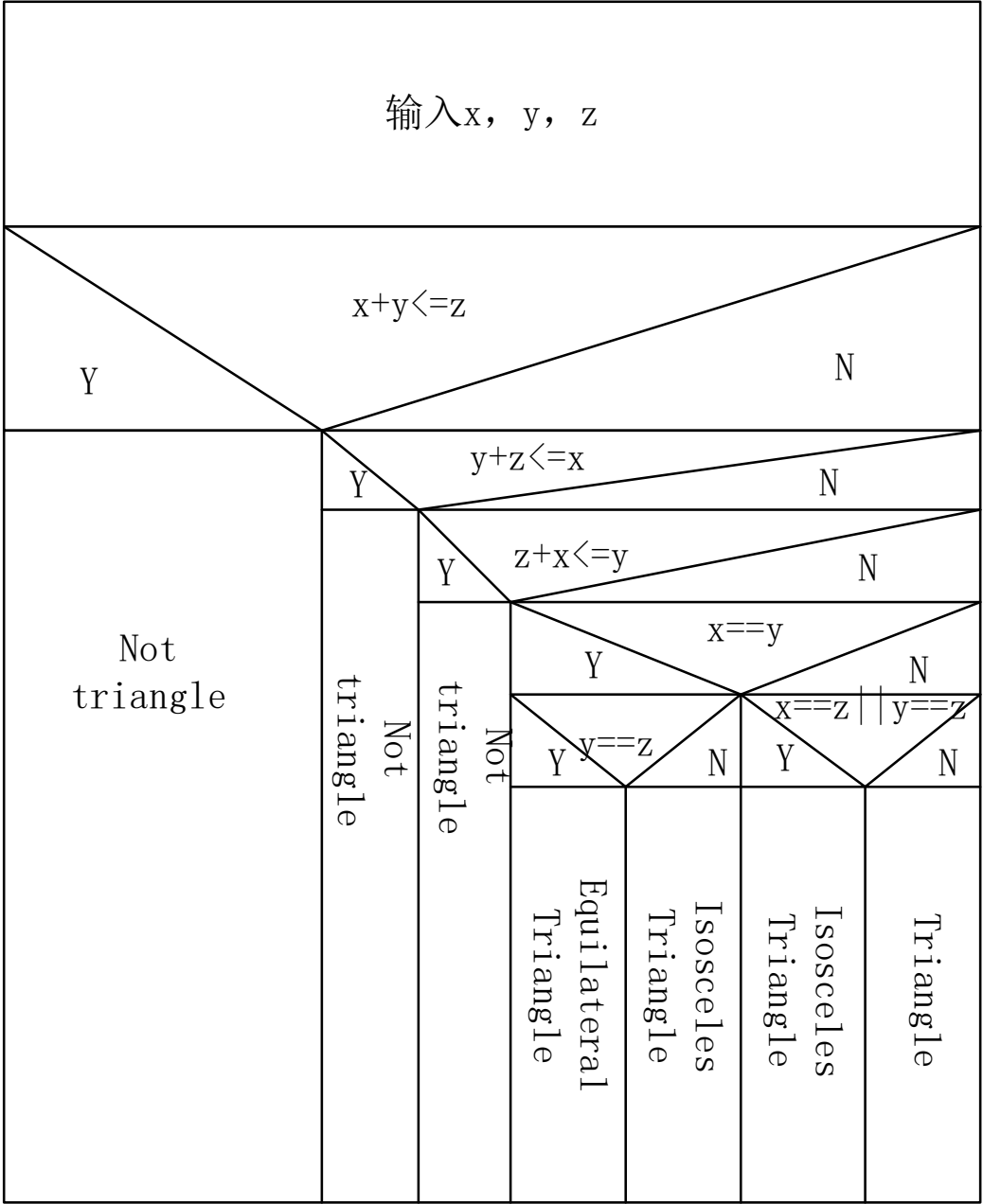
实验环境：

个人电脑（windows 11 操作系统），visual studio 2012 编译器。

实验内容：

必做题一：

算法的 NS 图：



程序代码：

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int x,y,z;
    unsigned short usTemp;
    scanf_s("%d %d %d",&x,&y,&z);
    if (x+y<=z) usTemp=0;
```

```

        else if (y+z<=x) usTemp=1;
        else if (z+x<=y) usTemp=2;
            else if (x==y) {(y==z)?(usTemp=3):(usTemp=4);}
                else {(x==z)|| (y==z))?(usTemp=5):(usTemp=6);}
switch(usTemp)
{
case 0:
case 1:
case 2:printf("Not triangle");break;
case 3:printf("Equilateral triangle");break;
case 4:
case 5:printf("Isosceles triangle");break;
case 6:printf("Triangle");
}
system("pause");
return 0;
}

```

运行结果:

```

1 4 9
Not triangle请按任意键继续. . . |

```

```

9 1 4
Not triangle请按任意键继续. . .

```

```

1 1000 4
Not triangle请按任意键继续. . .

```

```

9 9 9
Equilateral triangle请按任意键继续. . .

```

```

6 7 6
Isosceles triangle请按任意键继续. . .

```

```

5 5 9
Isosceles triangle请按任意键继续. . .

```

```

9 7 7
Isosceles triangle请按任意键继续. . .

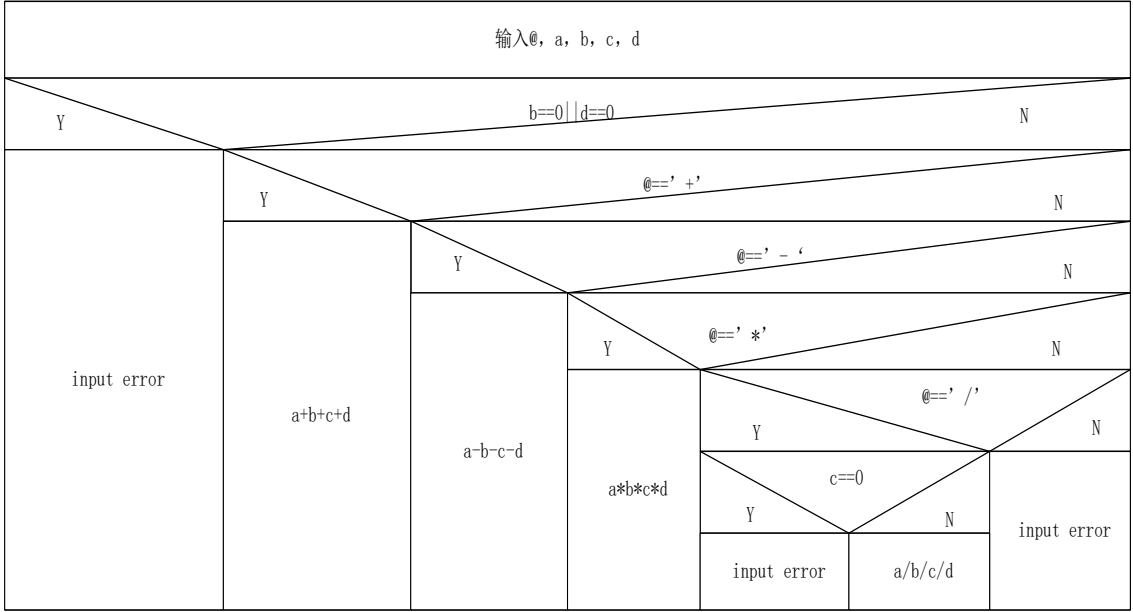
```

```

3 8 6
Triangle请按任意键继续. . .

```

必做题二：
算法的 NS 图：



程序代码：

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    double a, b, c, d;
    char cAt, cBuffer;
    unsigned short usTemp;
    scanf("%c%c %lf %lf %lf %lf", &cAt, &cBuffer, &a, &b, &c, &d); /*输入*/
    /*判断情况*/
    if (b==0||d==0) usTemp=0;
    else if (cAt=='+') usTemp=1;
    else if (cAt=='-') usTemp=2;
    else if (cAt=='*') usTemp=3;
    else if (cAt=='/') (c==0)?(usTemp=4):(usTemp=5);
    else usTemp=6;
    /*处理情况对应的输出*/
    switch(usTemp)
    {
        case 1:printf("%lf", a+b+c+d);break;
        case 2:printf("%lf", a-b-c-d);break;
        case 3:printf("%lf", a*b*c*d);break;
        case 5:printf("%lf", a/b/c/d);break;
        default:printf("input error");
    }
    system("pause");
    return 0;
}
```

运行结果:

```
/ 2 3 0 1.2
input error请按任意键继续. . .
```

```
* 2 0 1.2 2.9
input error请按任意键继续. . .
```

```
+ 1.2 9 1.2 3
14.400000请按任意键继续. . .
```

```
- -2 -7 1.2 -1.3
5.100000请按任意键继续. . .
```

```
* 2 3 0 1.7
0.000000请按任意键继续. . .
```

```
/ 7.1 1 0.71 0.5
20.000000请按任意键继续. . .
```

必做题三:

程序代码:

```
#include<stdio.h>
#include<stdlib.h>
int main() {
    int iInput=0;
    int iCopy=0;
    unsigned short usLength=0;
    unsigned short usTemp=0;
    unsigned short usTestRound=0;
    int iTenPower=1;
    for (usTestRound=0;usTestRound<12;++usTestRound)/*循环12次以测试12个不同的
数据是否能得到正确的结果*/
    {
        printf("这是第%u次测试。\\n请输入一个待测正整数，不超过5位:
",usTestRound+1);
        scanf_s("%d",&iInput);
        /*正确输入五位正整数的情况下执行程序主体*/
        if (iInput<1E5 && iInput>0) {
            /*判断输入整数的位数*/
            if (iInput<1E1) usLength=1;
            else if (iInput<1E2) usLength=2;
            else if (iInput<1E3) usLength=3;
            else if (iInput<1E4) usLength=4;
```

```

else usLength=5;
/*输出整数的位数*/
printf("该正整数的位数为%d\n", usLength);
iCopy=iInput;
/*顺序打印每一位数字*/
printf("顺序打印: ");
for (usTemp=0;usTemp<usLength;++usTemp) {
    iTenPower*=10;
}
for (usTemp=0;usTemp<usLength;++usTemp) {
    iTenPower/=10;
    printf("%d ", iCopy/iTenPower);
    iCopy%=iTenPower;
}
/*逆序打印每一位数字*/
printf("\n逆序打印: ");
for (usTemp=0;usTemp<usLength;++usTemp) {
    printf("%d ", iInput%10);
    iInput/=10;
}
}
else printf("输入不规范, 本次测试略过.\n");
printf("\n\n");
}
system("pause");
return 0;
}

```

运行结果:

这是第 1 次测试。

请输入一个待测正整数, 不超过 5 位: 12345

该正整数的位数为 5

顺序打印: 1 2 3 4 5

逆序打印: 5 4 3 2 1

这是第 2 次测试。

请输入一个待测正整数, 不超过 5 位: 1234

该正整数的位数为 4

顺序打印: 1 2 3 4

逆序打印: 4 3 2 1

这是第 3 次测试。

请输入一个待测正整数, 不超过 5 位: 123

该正整数的位数为 3

顺序打印：1 2 3

逆序打印：3 2 1

这是第 4 次测试。

请输入一个待测正整数，不超过 5 位：12

该正整数的位数为 2

顺序打印：1 2

逆序打印：2 1

这是第 5 次测试。

请输入一个待测正整数，不超过 5 位：1

该正整数的位数为 1

顺序打印：1

逆序打印：1

这是第 6 次测试。

请输入一个待测正整数，不超过 5 位：20000

该正整数的位数为 5

顺序打印：2 0 0 0 0

逆序打印：0 0 0 0 2

这是第 7 次测试。

请输入一个待测正整数，不超过 5 位：1003

该正整数的位数为 4

顺序打印：1 0 0 3

逆序打印：3 0 0 1

这是第 8 次测试。

请输入一个待测正整数，不超过 5 位：8238

该正整数的位数为 4

顺序打印：8 2 3 8

逆序打印：8 3 2 8

这是第 9 次测试。

请输入一个待测正整数，不超过 5 位：7330

该正整数的位数为 4

顺序打印：7 3 3 0

逆序打印：0 3 3 7

这是第 10 次测试。

请输入一个待测正整数，不超过 5 位：21

该正整数的位数为 2

顺序打印：2 1

逆序打印：1 2

这是第 11 次测试。

请输入一个待测正整数，不超过 5 位：0

输入不规范，本次测试略过。

这是第 12 次测试。

请输入一个待测正整数，不超过 5 位：1234567

输入不规范，本次测试略过。

选做题一：

程序代码：

```
#include<stdio.h>
#include<stdlib.h>
int main() {
    unsigned short usMinute=(unsigned short)0,usHour=(unsigned short)0; /*定义分
m、时h变量*/
    unsigned short funcEnglish(unsigned short); /*声明函数*/
    printf("请输入一个时间，先输入时，再输入分，用制表符隔开：");
    scanf("%hd\t%hd",&usHour,&usMinute); /*输入时间*/
    if ((usHour>=24) || (usMinute>=60)) printf("输入错误。"); /*检查输入是否有效*/
    else {
        /*输出小时*/
        if (usHour>20) { /*小时需用两个单词表示*/
            funcEnglish((unsigned short)20);
            printf(" ");
            funcEnglish((unsigned short)(usHour-20));
        }
        else funcEnglish(usHour); /*小时可用一个单词表示*/
        printf(" ");
        /*输出分钟*/
        if (usMinute==0) printf("o'clock"); /*整点*/
        else if ((usMinute%10==0) || (usMinute<20)) funcEnglish(usMinute); /*分
钟可以用一个单词表示*/
        else { /*分钟需要用两个单词表示*/
            funcEnglish((unsigned short)(usMinute/10*10));
            printf(" ");
            funcEnglish((unsigned short)(usMinute%10));
        }
    }
    system("pause");
    return 0;
}
```


/*输出数字对应英文单词的函数*/

```
unsigned short funcEnglish(unsigned short usNumber) {  
    switch(usNumber) {  
        case 0: printf("zero");break;  
        case 1: printf("one");break;  
        case 2: printf("two");break;  
        case 3: printf("three");break;  
        case 4: printf("four");break;  
        case 5: printf("five");break;  
        case 6: printf("six");break;  
        case 7: printf("seven");break;  
        case 8: printf("eight");break;  
        case 9: printf("nine");break;  
        case 10:printf("ten");break;  
        case 11:printf("eleven");break;  
        case 12:printf("twelve");break;  
        case 13:printf("thirteen");break;  
        case 14:printf("fourteen");break;  
        case 15:printf("fifteen");break;  
        case 16:printf("sixteen");break;  
        case 17:printf("seventeen");break;  
        case 18:printf("eighteen");break;  
        case 19:printf("nineteen");break;  
        case 20:printf("twenty");break;  
        case 30:printf("thirty");break;  
        case 40:printf("forty");break;  
        case 50:printf("fifty");break;}  
    return 0;  
}
```

运行结果:

```
请输入一个时间，先输入时，再输入分，用制表符隔开：0 0  
zero o'clock请按任意键继续. . . |
```

```
请输入一个时间，先输入时，再输入分，用制表符隔开：23 29  
twenty three twenty nine请按任意键继续. . . |
```

```
请输入一个时间，先输入时，再输入分，用制表符隔开：11 19  
eleven nineteen请按任意键继续. . . |
```

选做题二:

程序代码:

```
#include "stdio.h"
```

```

#include "stdlib.h"
#include "math.h"
double usDistance(double, double, double, double);
unsigned short usBuildingJudging(double, double);
int main() {
    double dX, dY; /*定义浮点坐标*/
    printf("请输入坐标，用空格隔开：");
    scanf("%lf %lf", &dX, &dY); /*输入*/
    printf("楼高： %d", usBuildingJudging(dX, dY)); /*输出*/
    system("pause");
    return 0;
}

/*定义计算距离的函数*/
double usDistance(double dCoordinateX, double dCoordinateY, double
dBuildingX, double dBuildingY) {
    return sqrt((dCoordinateX-dBuildingX)*(dCoordinateX-
dBuildingX)+(dCoordinateY-dBuildingY)*(dCoordinateY-dBuildingY));
}

/*定义判断在哪个建筑物上方的函数*/
unsigned short usBuildingJudging(double dCoordX, double dCoordY) {
    unsigned short usTemp=0;
    if (usDistance(dCoordX, dCoordY, 2.0, 2.0)<=1) usTemp=10;
    else if (usDistance(dCoordX, dCoordY, -2.0, 2.0)<=1) usTemp=9;
    else if (usDistance(dCoordX, dCoordY, -2.0, -2.0)<=1) usTemp=8;
    else if (usDistance(dCoordX, dCoordY, 2.0, -2.0)<=1) usTemp=7;
    return usTemp;
}

```

运行结果：

```

请输入坐标，用空格隔开：-1.5 -1.5
楼高：8请按任意键继续... |

```

```

请输入坐标，用空格隔开：2.1 -1.9880
楼高：7请按任意键继续... |

```

```

请输入坐标，用空格隔开：0.1 -1.2
楼高：0请按任意键继续... |

```

```

请输入坐标，用空格隔开：2 1
楼高：10请按任意键继续... |

```

请输入坐标，用空格隔开：-1 2
楼高：9请按任意键继续. . . |

选做题三：

程序代码：

```
#include "stdio.h"
#include "stdlib.h"
int main() {
    int iTestNumber; /*用来储存输入的数*/
    unsigned short usFive=0, usSix=0, usSeven=0; /*用来储存5、6、7的个数*/
    printf("请输入一个不要太大的正整数：");
    scanf("%d", &iTestNumber); /*输入*/
    while (iTestNumber!=0) {
        switch(iTestNumber%10) {
            case 5: usFive+=1; break;
            case 6: usSix+=1; break;
            case 7: usSeven+=1; break;
        }
        iTestNumber/=10;
    }
    printf("5的个数： %d\n6的个数： %d\n7的个数： %d", usFive, usSix, usSeven);
    system("pause");
    return 0;
}
```

运行结果：

请输入一个不要太大的正整数：777022565
5的个数：2
6的个数：1
7的个数：3请按任意键继续. . . |

实验体会：

- 1、基本学会了 NS 图的绘制，感到可视化算法设计对程序设计捋清思路的作用；
- 2、基本学会了 switch 语句的使用，发现比 if 确实结构清晰了不少；
- 3、初步锻炼了循环结构和函数的使用。