实验三：

一．

1. 根据输入的腰长size，输入如下等腰三角形。
2. 思路分析：根据腰长设置循环次数，找出输出空格的规律。第一行和最后一行单独输出。
3. 代码实现：

#include<stdio.h>

void trangle(){

int size,i,j;

scanf("%d",&size);

for( i = 0 ; i < size ; i++ ){

if( i == 0 ){

for( j = 0 ; j < size - 1 - i ; j++ ){

putchar(' ');

}

putchar('\*');

putchar('\n');

}

else if( i == size - 1 ){

for( j = 0 ; j < size \* 2 - 1 ; j++ ){

putchar('\*');

}

}

else{

for( j = 0 ; j < size - i - 1 ; j++ ){

putchar(' ');

}

putchar('\*');

for( j = 0 ; j < i \* 2 - 1 ; j++ ){

putchar(' ');

}

putchar('\*');

putchar('\n');

}

}

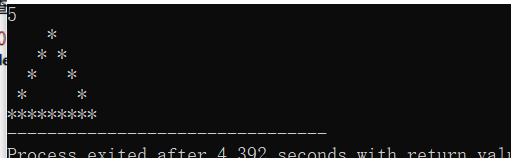
}

int main(){

trangle();

}

1. 运行截图



二：

1. 凸包问题：给定平面中的一个点集，它的凸包就是包含所有这些点的凸多边形。输出点集的凸包。
2. 思路分析：根据给出的数学公式，遍历所有的二元点的组合，判断其他点是否在选出两点连成直线的同侧，同侧输出这组点。暴力解法。
3. 代码实现：

#include<stdio.h>

#define NUMBER 7

void exp(){

int x[NUMBER] = {1,3,5,1,2,2,2};

int y[NUMBER] = {1,1,4,3,2,3,6};

int i,j,k;

int a,b,c;

int count1,count2;

for( i = 0 ; i < NUMBER - 1 ; i++ ){

for( j = i + 1 ; j < NUMBER ; j++ ){

a = y[j] - y[i];

b = x[i] - x[j];

c = x[i] \* y[j] - x[j] \* y[i];

count1 = 0;

count2 = 0;

for( k = 0 ; k < NUMBER ; k++ ){

if( a \* x[k] + b \* y[k] > c ){

count1++;

}

if( a \* x[k] + b \* y[k] < c ){

count2++;

}

}

if(count1 \* count2 == 0)

printf("%d--%d\n",i+1,j+1);

}

}

}

int main(){

exp();

return 0;

}

1. 运行截图

