# include<stdio.h>

# include<string.h>

# include<math.h>

//基础函数

void print1x9(int n)

{

int i, k = 0;

while(1)

{

for(i=0; i<10; ++i)

{

if(i == 0)

printf("\*");

else

printf("%d", i);

k++;

if(k==n)

{

printf("\n");

return;

}

}

}

printf("\n");

}

void printstrx(char \*ch)

{

printf("\*%s\*\n", ch);

}

void printx(int i)

{

//i += 2;

while(i--)

printf("\*");

printf("\n");

}

void samequestion1(char \*str)//第三题与第八题相似

{ //合并成同类型函数

printx(strlen(str) + 2);

printstrx(str);

printx(strlen(str) + 2);

}

void samequestion2(int \*i)

{

int j, k;

for(j=0; j<(6-\*i)/2; ++j)

printf("#");

for(k=0; k<\*i; ++k)

printf(" ");

for(j; j<6-\*i-1; ++j)

printf("#");

printf("\n");

}

//以下为实现函数

void print1()

{

int i, j, k;

for(i=0; i<2; ++i)

samequestion2(&i);

printf("# #\n");

//i--;

while(i--)

samequestion2(&i);

}

void print2()

{

int i, j;

printx(9);

for(i=1; i<10; ++i)

{

for(j=i; j<10; ++j)

printf("%d", j);

printf("\n");

}

printx(9);

}

void print3()

{

char ch[] = {"1 + '1' = '2'"};

samequestion1(ch);

}

void print4()

{

int i, j;

print1x9(20);

for(i=10; i>0; --i)

{

for(j=i; j<10; ++j)

printf(" ");

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

printf("@");

printf("\n");

}

print1x9(20);

}

void print5()

{

int a = 0, b = 1, c, i;

print1x9(25);

printf("%5d%5d", a, b);

for(i=1; i<=18; i++)

{

c = a + b;

printf("%5d", c);

a = b;

b = c;

if(i%5 == 3)

printf("\n");

}

print1x9(25);

}

void print6()

{

char str[] = {" "};

printx(5);

printstrx(str);

printstrx(str);

printstrx(str);

printx(5);

}

void print7()

{

int i, j;

printx(9);

for(i=1; i<10; ++i)

{

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

printf(" ");

for(j=i; j<10; ++j)

printf("%d", j);

printf("\n");

}

printx(9);

}

void print8()

{

char str[] = {"\"Hello!\""};

samequestion1(str);

}

void print9()

{

int i, j;

print1x9(20);

for(i=1; i<=10; ++i)

{

for(j=i; j<10; ++j)

printf(" ");

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

printf("@");

printf("\n");

}

print1x9(20);

}

void print10()

{

int i, j, k;

int num[16][16];

int n = 15;

print1x9(80);

for(i=0; i<(n+1); ++i)

{

/\*

下面两行代码

目的为输出带空格的

杨辉三角

\*/

//for(j=i; j<n; ++j)

//printf(" ");

num[i][0] = 1;

//num[i][i] = 1;

printf("%5d", num[i][0]);

for(j=1; j<=(i/2); ++j)

{

num[i][j] = num[i-1][j] + num[i-1][j-1];

printf("%5d", num[i][j]);

}

k = (i/2) + 1;

if(i != 0)

{

if(i%2 == 0)

j--;

while(j--)

{

num[i][k] = num[i][j];

printf("%5d", num[i][k]);

k++;

}

}

puts("\n");

}

print1x9(80);

}

void print11()

{

int i, j;

print1x9(80);

for(i=1; i<=20; ++i)

{

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

printf("%4d", j\*i);

printf("\n");

}

print1x9(80);

}

void print12()

{

int i, n = 0;

print1x9(80);

while(1)

{

for(i=0; i<20; ++i)

{

n++;

printf("%4d", n);

}

printf("\n");

if(n == 400)

break;

}

print1x9(80);

}

//主函数

int main(void)

{

int n;

while(~scanf("%d", &n))//循环输入

{

switch(n)

{

case 1:

print1();

printf("\n");

break;

case 2:

print2();

printf("\n");

break;

case 3:

print3();

printf("\n");

break;

case 4:

print4();

printf("\n");

break;

case 5:

print5();

printf("\n");

break;

case 6:

print6();

printf("\n");

break;

case 7:

print7();

printf("\n");

break;

case 8:

print8();

printf("\n");

break;

case 9:

print9();

printf("\n");

break;

case 10:

print10();

printf("\n");

break;

case 11:

print11();

printf("\n");

break;

case 12:

print12();

printf("\n");

break;

default:

printf("error!\n");//跳出switch

}

if(n<1 || n>12)

break;//结束程序

}

return 0;

}

/\*

作业17

提高复用性可更改switch

里面函数（增加形参）

2017.12.1

\*/