//Mimi

1. Diamonds (both full and empty) and an X of stars. The user will

supply the height. The empty diamond and the X are of height 7

and the full diamond is of height 9.

//full diamond pattern

#include<iostream>

using namespace std;

int main(){

int n=5;

int i, a;

//top

for(i=1 ; i<n; i++){ //change i<=n to i<n(to decrease one row), so then there's side peak

for(a=i; a<=n; a++){ //decreasing pattern(if i=2, a will print from 2to5(4 empty spaces))

cout << " ";

}

for(a=1; a<i; a++){ //increasing pattern(if i=2, a will print from 1to2(2stars))

cout << "\* "; //change from a<=i to a<i (to decrease one column),so then there's a peak

}

for(a=1; a<=i; a++){ //increasing pattern(if i=3, a will print from 1to3(3stars))

cout << "\* ";

}

cout << endl;

}

//bottom

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

for(a=1; a<=i; a++){ //increasing pattern(if i=2, a will print from 1to2(2empty spaces))

cout << " ";

}

for(a=i; a<n; a++){ //decreasing pattern(if i=2, a will print from 2to5(4stars))

cout << "\* "; //change from a<=i to a<i (to decrease one column),so then there's a peak

}

for(a=i; a<=n; a++){ //decreasing pattern(if i=3, a will print from 3to5(3stars))

cout << "\* ";

}

cout << endl;

}

}

//cross pattern

#include<iostream>

using namespace std;

int main(){

int c=7;

int i,j;

for(i=1; i<=c; i++){

for(j=1 ;j<=c; j++){

if(i==j || i+j == c+1){ //print \* if i==j(diagonal line from left to right)

cout << "\*"; //or print \* if i+j==8(diagonal line from rigth to left)

}

else

cout << " "; //else make it an empty spaces

}

cout << endl;

}

}

//draw table which indicates row and column to understand this

//empty diamond

#include<iostream>

using namespace std;

//\*\*look at the spacebar carefully(error might occur)

int main(){

int d=5;

int i,j;

for(i=1; i<d; i++){ //decreasing, increasing^2

for(j=i; j<=d; j++){

cout << " ";

}

for(j=1; j<i; j++){ //increasing

if(j==1){

cout << "\* ";

}else

cout << " ";

}

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

if(j==i){

cout << "\* ";

}else

cout << " ";

}

cout << endl;

}

for(i=1; i<=d; i++){ //increasing, decreasing^2

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

cout << " ";

}

for(j=i; j<d; j++){

if(j==i){

cout << "\* ";

}else

cout << " ";

}

for(j=i; j<=d; j++){

if(j==d){

cout << "\* ";

}else

cout << " "; //there need to be two spacebar

}

cout << endl;

}

}

2. A multiplication table where the user specifies the size. A table of size 6 is shown.

//multiplication table

#include<iostream>

using namespace std;

int main(){

int x; //size

int i,j; //i=row, j=column

cout << "Enter the value for multiplication table : ";

cin >> x;

for(i=1; i<=x; i++){

for(j=1; j<=x; j++){

cout << i\*j << "\t";

}

cout << endl;

}

}

3. Pascal’s Triangle. The user will specify the height. The triangle shown below is of height 5.

//Pascal triangle

#include<iostream>

using namespace std;

int main()

{

int rows, num=1, i, j; // i=rows, j=column, rows=counter of rows, num=output of numbers we want(need to be1 because it's constant value)

cout<<"Enter the number of rows you want to be in Pascal's triangle: ";

cin>>rows;

cout<<endl;

for(i=0; i<rows; i++) //i need to be equal to 0 in this case, or it'll output weird structure and wrong numbers

{

for(j=i;j<=rows;j++){ //decreasing triangle(output empty spaces)

cout<< i;

}

for(j=0;j<=i;j++){ //j has to be <= i ,or else the right side won't print number 1

if(j==0){ //you can write if(column=0) or if(row==0 || column==0)

num = 1; //output number 1, if num=2 it'll output number2

}

else{

num = num\*(i-j+1)/j; //important code that will control the output number

}

cout << num << " ";

}

cout << endl;

}

return 0;

}

4. Estimate the value of pi using the infinite sequence. The user

will specify the number of terms.

//estimate the value of pi

#include<iostream>

#include<iomanip> //for setprecision(x)

using namespace std;

int main(){

double denom, num=0; //denom=odd number, num=value of pi

int term, i, x; //term=for user input, i=counter, x=number of decimal

cout << "Enter how many terms you want to estimate the value of pi : ";

cin >> term;

cout << "How many decimal you want for pi: ";

cin >> x;

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

denom=2\*i+1; //denominator need to be odd number

if(i%2 != 0)

num -= 1/denom; //if i is an odd term, subtract 1/odd number

else

num += 1/denom; //else if i is an even term, add 1/odd number

}

num \*= 4; //times 4 later after calculating

cout << fixed << setprecision(x) << "\nThe value of pi : " << num;

cout << "\nPress enter to continue...";

cin.get();

}