Lab 11/29

Please do not use array or global variable in this lab.
Please use recursive function to solve problem, and print out the answer in the main function.

Lab Part

1. Write the following functions:

   (a). dec2bin(n): convert decimal number into binary using recursion.

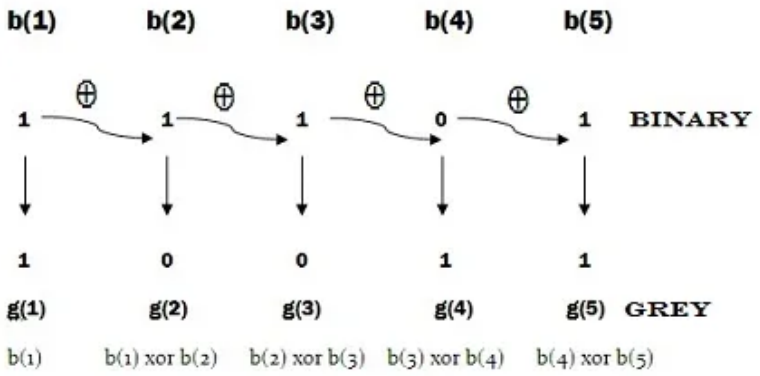
   (b). bin2gray(n): convert binary number into gray code using recursion.

   (c). gray2bin(n): convert gray code into binary number using recursion.

   Call these functions and print the result in the main function as shown in the example. Let the user continuously input n until Ctrl+Z.

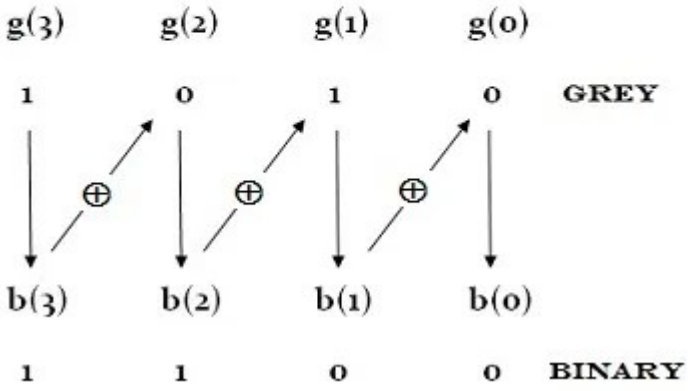
Binary to Grey Code Conversion

Convert the binary 11101_2 to its equivalent Grey code



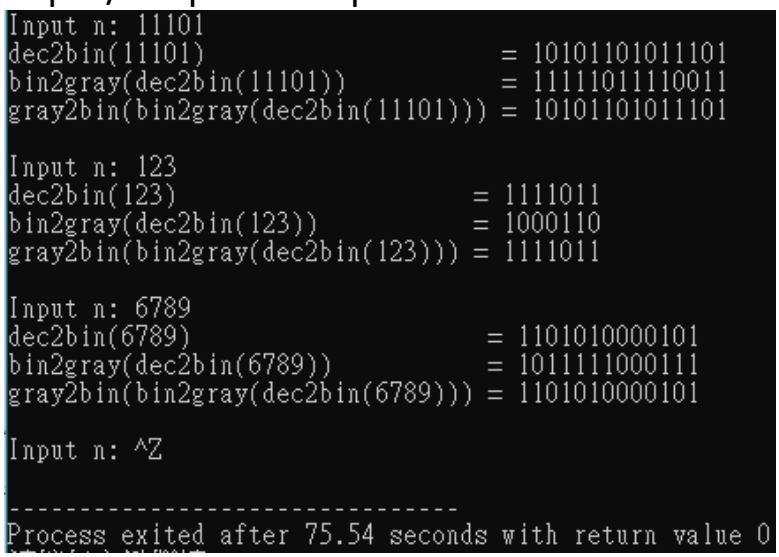
Grey Code to Binary Conversion

Convert the Grey code 1010 to its equivalent Binary

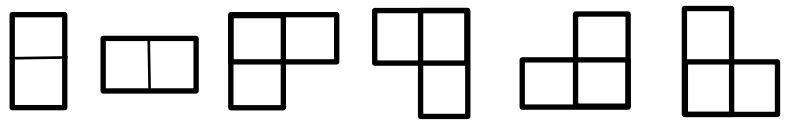


Binary to Gray Code Converter (ncalculators.com)

Input/Output Example:



2. Suppose that we have a 2 × n rectangular board divided into 2n squares. Please write a recursive function that computes the number of ways to cover this board exactly by the following dominoes. Let the user input the number n. Please print out the answer in the main function. Stop the program when inputting CTRL+D.

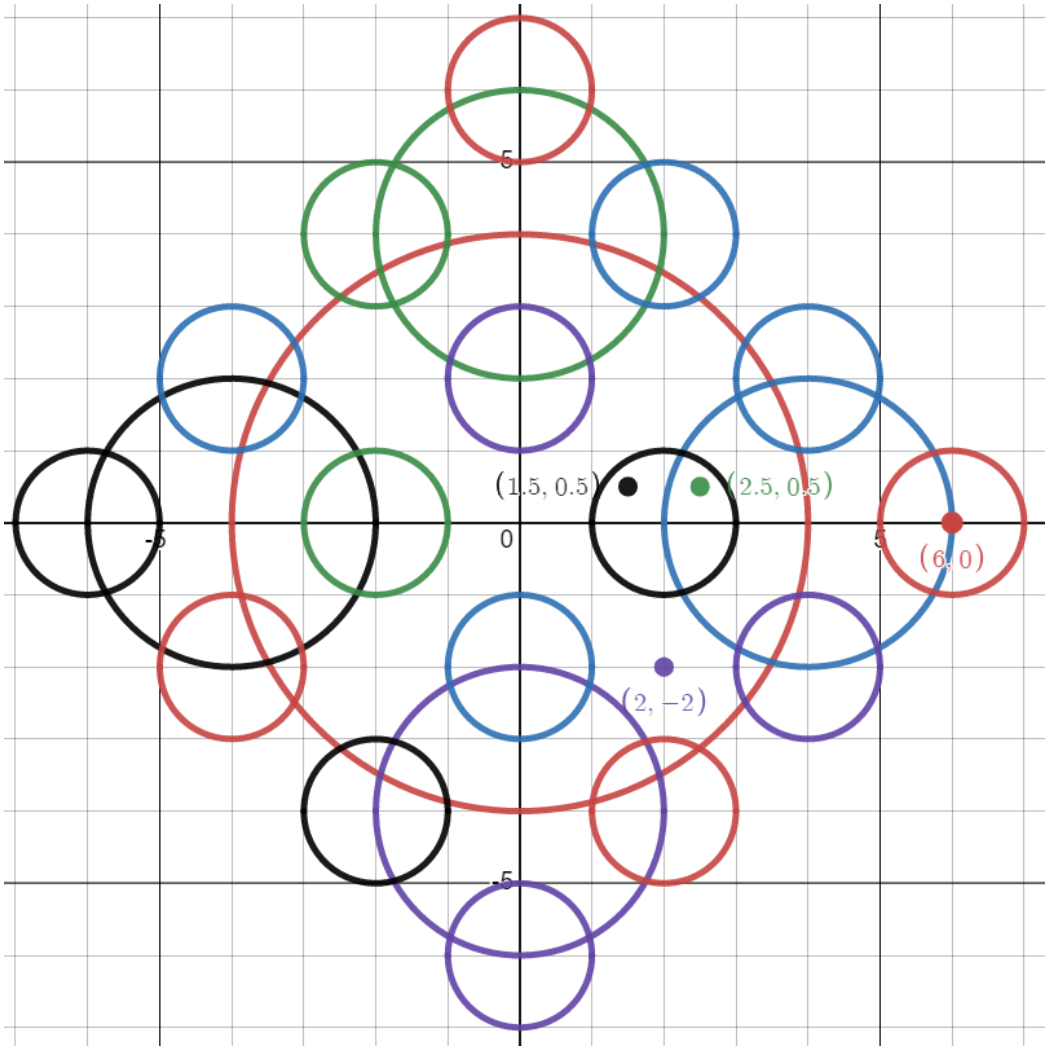


Input/Output Example:

```
n = 3
There are 5 ways

n = 5
There are 24 ways

n = 7
There are 117 ways

n = 0
There are 1 ways

n = 1
There are 1 ways

n = ^D

--------------------------------
Process exited after 7.935 seconds with return value 0
```

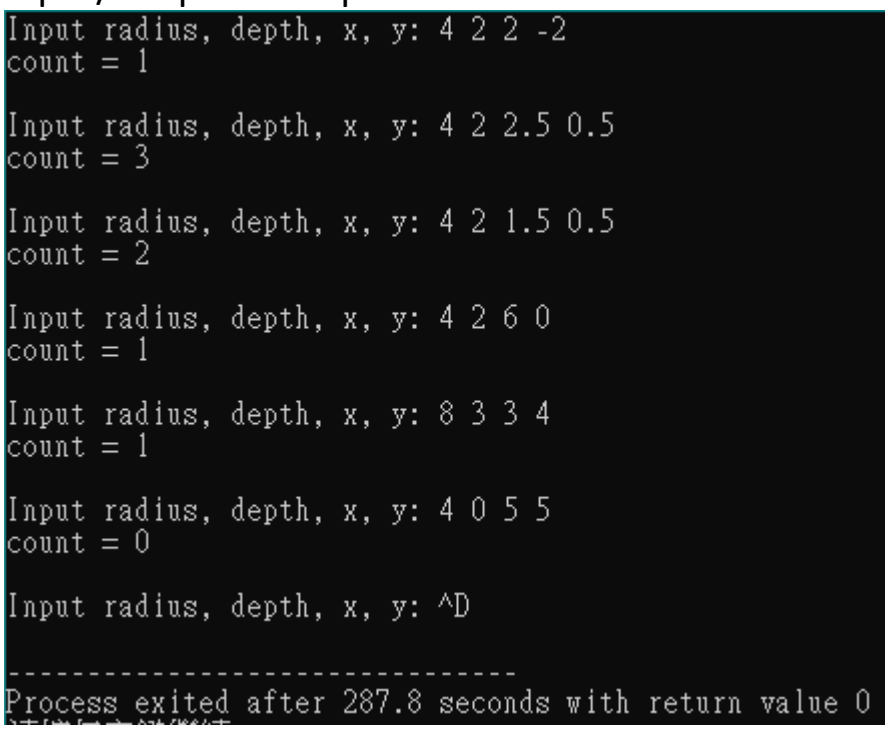
3. We can draw a circle with a center point and a radius. Now define a pattern of circles as follows:
   (a). Given the radius, we can draw the largest circle which is centered in origin (0,0).
   (b). There are 4 circles with half of the radius on 4 positions of each circle as shown below.
   (c). Repeat the process of (b) for a given depth.

   Example for radius = 4, depth = 2



   Please write a recursive function to calculate the number of circles surrounding the point. ($\boldsymbol{distance\_to\_center} < \boldsymbol{r}$)
   Let the user input the largest circle's radius, depth, and point (x, y), then output the answer in the main function.
   Let the user continuously input until Ctrl+D.

Input/Output Example



Homework Part

4. Write a program to approximate the value of sin(x) using the formula:

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \ldots + \frac{(-1)^{n+1} x^{2n-1}}{(2n-1)!}$$ (x 值由 keyboard key in).

The program stops when $\left| \frac{(-1)^{n+1} x^{2n-1}}{(2n-1)!} \right| < 10^{-12}$.

Once the term is less than 10^{-12}, this term "should not" be added into the sum. **Input x in the main function** and pass this data to the following functions by value:

Case I:

Write one value return function and use **one while loop** to calculate the sin value in the function body and then display the result to the **14th** decimal place in the main.

Case II:

Write one value return recursive function to calculate the sin value and display the result to the **14th** decimal place in the main**.**

Case III:

Use the same input to call the **sin** function in the **math.h** and display the result to the 14th decimal place in the main.

The input will stop when you enter Ctrl D.

Modification:

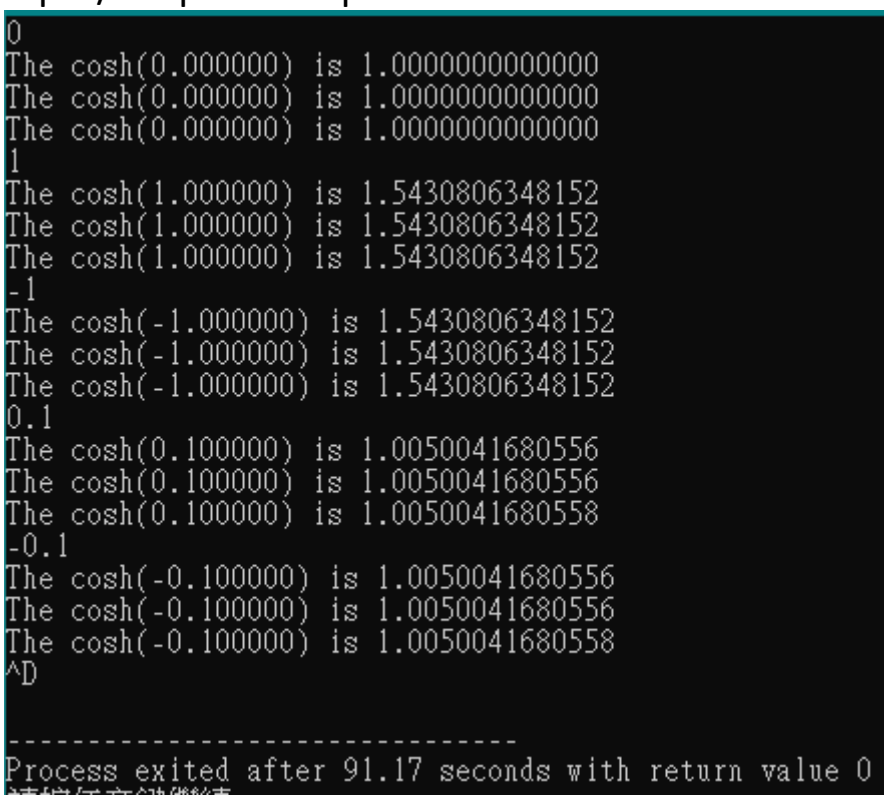
(a). ~~sin(x)~~ => $\cosh(x) = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \cdots + \frac{(1)^n x^{2n}}{(2n)!}$

(b). ~~$\left| \frac{(-1)^{n+1} x^{2n-1}}{(2n-1)!} \right| < 10^{-12}$~~ => 10^{-11}

(c). ~~14th decimal place~~ => 13th decimal place

(d). **sin** function in the **math.h** => **cosh** function

Input/Output Example:



5. Write a program that inputs two numbers: x and y. (data are all integers) in the main program and passes these two numbers (pass by value) to the recursive function: power that returns the x^y.

If $y \geq 0,\ power(x,y) = \begin{cases} 1 & if\ y = 0 \\ x & if\ y = 1 \\ x * power(x, y-1) & if\ y > 1 \end{cases}$

If $y < 0,\ power(x,y) = \frac{1}{power(x,-y)}$

Please print the result in the main program and show the answer to the 8th decimal place. The program should be able to execute repeatedly until user entering CTRL+D

Modification:

If $y \geq 0,\ power(x,y) = \begin{cases} 1 & if\ y = 0 \\ x & if\ y = 1 \\ 0.5 * x * power(x, y-1) & if\ y = 2 \\ x * power(x, y-1) & if\ y > 2 \end{cases}$

If $y < 0,\ power(x,y) = \frac{1}{power(x,-y)}$

Input/Output Example:

```
x,y = 2 0
power(x,y)= 1.00000000
x,y = 2 1
power(x,y)= 2.00000000
x,y = 2 -1
power(x,y)= 0.50000000
x,y = 2 2
power(x,y)= 2.00000000
x,y = 2 3
power(x,y)= 4.00000000
x,y = 3 4
power(x,y)= 40.50000000
x,y = 3 -4
power(x,y)= 0.02469136
x,y = ^D

--------------------------------
Process exited after 27.86 seconds with return value 0
```

6. A robot can take steps of 1 meter, 2 meters and 3 meters. Write a recursive function to evaluate the number of ways the robot can walk n meters. Let the user input the number **n**. Please print out the answers in the main function. Stop the program when inputting **CTRL+Z**

Modification:

A robot can take steps of 1 meter, 2 meters and 5 meters

Input/Output Example: