

SDS 322/329:
Introduction to Scientific Programming

Midterm (100 points)

DUE: Thursday, October 20th

Directions:

Create a local directory in your account called:

Midterm_***youreid***

Each question will be a separate program in your Midterm directory. The name of each program will be:

Midterm_Q***questionnumber***_youreid.c or .cc

Then, using the tar command, tarball all your questions together into a .tar file called

Midterm_***youreid***.tar

and copy the tar file into the submission folder.

/home/charlie/submissions/Midterm

You only have write access to the submission folder, only the instructors and the TA have read access.

NOTE:

Creating a tar file:

tar -cvf Midterm_***youreid***.tar ~/Midterm_***youreid***/

Random number generator demo code:

/home/charlie/Demo/Code/RandomNumberGen/rand.c

Q1. Control Structures (5 points each)

Suppose I have a 10x10 double dimension array, A.

Write code in **C** that does the following:

- Fill the array with random numbers between 0 and 10
- Sums up each column and puts the sums in a new array
- Sums up each row and puts the sums in a new array
- Calculates the trace (sum of the diagonal elements)
- Transposes the matrix (swaps rows and columns)
- Finds the sum of all the elements, finds the average value. Goes through the matrix, for each value > the average, replace the value with a 1; for each value < the average, replace the value with a 0

Q2. Arrays (5 points each)

Given the following declarations:

C:

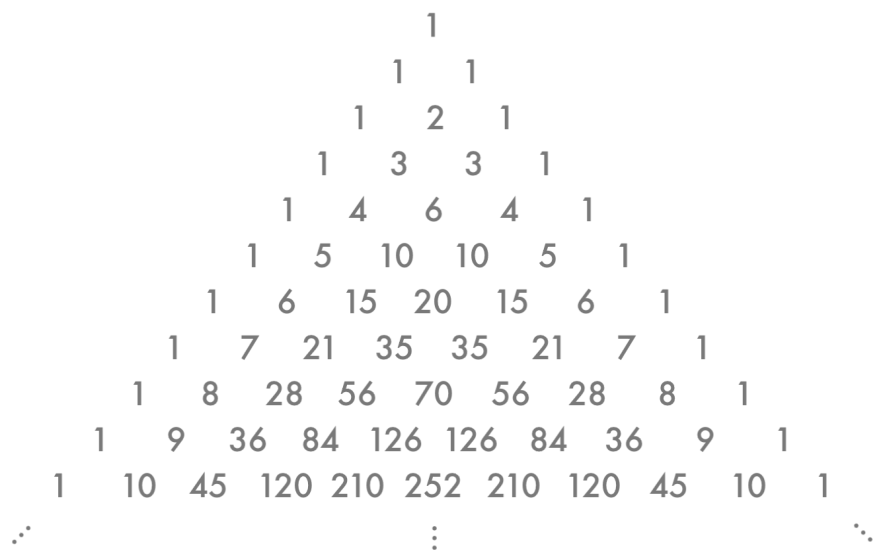
```
float x;  
float a[10], b[10];  
float c[10][10], d[10][10];
```

Initialize the arrays to random numbers, then write **C** code that does the following:

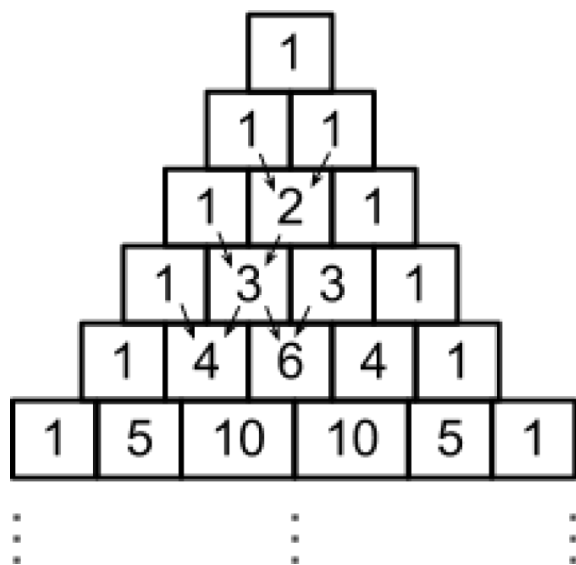
- sets **a** equal to **b**
- sets all elements of **a** equal to **x**
- sets **a** equal the first row of **c**
- sets the first 3 elements of **a** equal to the last 3 elements of **c**
- sets **c** equal to **d**

Q3. A Program. (45 points)

This is the first 10 rows of Pascal's Triangle:



Each number is obtained as the sum of the two neighboring numbers in the preceding row.



Pascal's triangle is very cool, and has a lot of hidden patterns

- the Fibonacci Sequence is hidden in the sum of the soft diagonals
- each row provides the coefficients of the expanded binomial raised to a power equal to the row number:

$$\begin{aligned}(a + b)^0 &= 1 \\(a + b)^1 &= a + b \\(a + b)^2 &= a^2 + 2ab + b^2 \\(a + b)^3 &= a^3 + 3a^2b + 3ab^2 + b^3 \\(a + b)^4 &= a^4 + 4a^3b + 6a^2b^2 + 4ab^3 + b^4 \\(a + b)^5 &= a^5 + 5a^4b + 10a^3b^2 + 10a^2b^3 + 5ab^4 + b^5\end{aligned}$$

It is also a cool fractal design.

Part 1 (30 points)

Using C, write a program that calculates the first 50 rows of Pascal's Triangle.

If the value is even, print a space, if the value is odd, an X. (Mod 2)

Print out the triangle of spaces and X's

Part 2 (15 points)

Then, ask the user to enter an integer, N.

Take the N modulus of the elements from Pascal's triangle, if the modulus is 0, print a space, else print an X

Print out the triangle of spaces and X's

Extra Credit. (15 points)

We have 3 objects, a point, a line, and a square:

Point object:

an x, y coordinate

Line object:

made up of two points

a function returns the length

Square objects:

made of 4 lines

a function that returns the area

a function that returns the length of the perimeter

Write a C++ program that allows the user to enter 2 points (a1, b1) and (a2, b2) to build a line and returns 2 adjacent square objects that share that line as a side, print out the coordinates of the 6 lines

