|  |
| --- |
| **Course: Programming Fundamental – ENSF 337**  Lab #: 3  Instructor: Khedr  Student Name: Aleksander Berezowski  Lab Section: B04  Date submitted: October 14th |

**Exercise A**

Diagram

Description automatically generated

**Exercise B**

Diagram

Description automatically generated

**Exercise C**

Diagram, schematic

Description automatically generated

**Exercise D**

Text

Description automatically generated

*void* pascal\_triangle(*int n*) {  
 *//DESCRIPTION: Prints a pascal's triangle of n rows   
 //PROMISES: Prints a correctly assembled pascal's triangle with no return  
 //REQUIRES: n must belong to all whole numbers, fit within int type, n >= 1  
   
 //Variable definitions  
 int* i; *//y-coord  
 int* j; *//x-coord   
 int* currentArray[*n*]; *//n length array used to print values  
 int* previousArray[*n*]; *//n length array used tostore previous values   
   
 //Clears both arrays incase of memory garbage  
 for*(i=0; i<*n*; i++){  
 currentArray[i] = 0;  
 previousArray[i] = 0;  
 }  
   
 *//Loop through each line of the triangle (n lines)  
 for*(i=0; i<*n*; i++){  
   
 *//Copies currentArray to previousArray  
 for*(j=0; j<*n*; j++){  
 previousArray[j] = currentArray[j];  
 }  
   
 *//Assembles currentArray for line i  
 for*(j=0; j<*n*; j++){  
 *if*(j==0){  
 *//If on first index, set it equal to one* currentArray[j] = 1;  
 } *else* {  
 *//If on any other index, calculate the value* currentArray[j] = previousArray[j-1] + previousArray[j];  
 }  
 }  
   
 *//Display line i by printing currentArray  
 for*(j=0; j<*n*; j++){  
 *//Only print non-0 numbers (makes it look cleaner)  
 if*(currentArray[j] != 0){  
 printf("%d\t", currentArray[j]);  
 }  
 }  
 printf("\n");  
  
 }  
}

**Exercise E**

Text

Description automatically generated

**Exercise F**

Text

Description automatically generated