Long Exam;

Pseudocode && flowchart;

MEMBERS []= {Jetrho ruiz vargas, maria taylor pena, vince regala, isaac manalo, karl marx roxas};

2020

Pseudocode in Pre-game

**B. Pre-game**

sum = 0

counter = 0

while counter < 1000

if counter % 3 == 0 || counter % 5 == 0

if counter % 6 == 0

end if

else

print counter

sum = sum + counter;

end else

end if

counter ++

end while

print "sum: " + sum

END

150 002

**C. Pre-game**

Input

M = 10

N = 11

1.) //insert x for the integer

2.) //input the integer numbers where M & N as a starting value

3.) //program that asks input integers

4.) //let's print the Fibonacci series

5.) //for loop

6.) //print integers 1 to 10

Output

Fibonacci sequence are series of:

10 11 21 32 53 85 138 223 361 584

Flowchart in Minor League

A.

START

END

print stock

j++

i ++

TRUE

TRUE

FALSE

FALSE

Stock += (array[j] \* array[i]) + array[j]

j = 0; j < n

i = 0; i < n

int [] array = {1, 2, 3};

int n = array.length;

int i;

int k;

int j;

int x;

int stock = 0;

B.

static String isPalindromic(String A)

(function) START

String enter\_string = A;

String stock\_1 = “”;

i- -

FALSE

int i = enter\_string.length() -1; i >= 0;

TRUEW

stock\_1 += enter\_string.charAt(i)

return stock\_1

END FOR

START (MAIN FUNCTION)

If(isPalindromic(“mmaad”).equals(“daamm”))

TRUE

ELSE

Print(“mmaad: “)

Print(“false”)

Print(“mmaad: “)

Print(“true”)

If(isPalindromic(“window”).equals(“window”))

TRUE

Print(“window: “)

Print(“true”)

ELSE

Print(“window: “)

Print(“false”)

END

C.

START (MAIN FUNCTION)

static void whatArray(int[][] A)

(function) START

n = 0

m = 0

END FOR

print “square”

print “skinny”

print “fat”

n < karl\_A.length

END

TRUE

If m < n

If m > n

TRUE

If m == n

print m + “ m”

print n + “ n”

END FOR

m < karl\_A[m].length

n++

FALSE

FALSE

TRUE

TRUE

FALSE

TRUE

FALSE

int[][] A = {{1, 3, 21, 9}, {2, 8, 7}, {12, 12, 12, 1}};

whatArray(A)

n++

END

FALSE

D.

print (i + " , " + j + " , " + k + "")

TRUE

TRUE

TRUE

FALSE

FALSE

FALSE

END

X[]= {10, 20, 10, 40, 50, 60, 70}

int sum\_1 = 40

int sum\_2 = 80

int arr\_size = X.length

threeIndices(X, arr\_size, sum\_1)

threeIndices(X, arr\_size, sum\_2)

START (MAIN FUNCTION)

j ++

END

return false

j ++

i ++

END FOR

END FOR

END FOR

return true

boolean threeIndices(x[], arr\_size, sum)(FUNCTION) START

int i=0;

i < arr\_size -2

int j= i + 1;

j < arr\_size -1

int k= j + 1;

k < arr\_size

If(X[i] + X[j] + X[k] == sum)

E.

Int[] addArray(nums[])

START

int i = nums.length -1; i>-1

END

END

print("A = " + Arrays.toString(nums\_1))

print("A'= " + Arrays.toString(addArray(nums\_1))

print()

print("A = " + Arrays.toString(nums\_2))

print("A'= " + Arrays.toString(addArray(nums\_2))

[]nums\_1 = {5, 6, 7}

[]nums\_2 = {9, 9, 9}

(MAIN FUNCTION)

START

int [] result = new int [nums.length +1]

result[0] = 1

return result

return nums

TRUE

TRUE

TRUE

FALSE

FALSE

--i

++j

nums[j] = 0

int j = i+1;

j<nums.length

nums[i] += 1

If(nums[i] !=9)

Pseudocode in Major League

A.

A. Major League

Function newArray[][](int [][] A, int [][] B)

row1 =2;

row2 =2;

col1 =2;

col2 =2;

stock[][] = new int[row][col2]

for(int i = 0; i < row1; i++){

for(int j = 0; j < col2; j++){

for(int k = 0; k < col1; k++){

stock[i][j] += (A[i][k] \* B[k][j]) + B[k][j];

end for

end for

end for

print(“array C: “)

for(int[] r : stock){

for(int c : r){

System.out.print(c);

System.out.print(" ");

end for

System.out.println("");

end for

return stock;

main function

A[][] = {{1, 2, 3}, {4, 5, 6}};

B[][]= {{11, 12}, {14, 15}, {17, 18}}

(Call function) newArray(A, B);

B.

B. Major League

Function reduce (int [][]A, int [][]B)

if(A[0][0] == A[1][0] && A[0][1] == A[1][1])

if (B[0][0] == B[1][0] && B[1][0] == B[2][0] && B[2][0] == B[3][0]

&& B[0][1] == B[1][1] && B[1][1] == B[2][1] && B[2][1] == B[3][1]

&& B[0][2] == B[1][2] && B[1][2] == B[2][2] && B[2][2] == B[3][2]

&& B[0][3] == B[1][3] && B[1][3] == B[2][3] && B[2][3] == B[3][3]

Print(“Hooray”)

End if

End if

Else

End else

Main function

int [][] A = {{1, 2}, {1, 2}}

int [][] B = {{1, 1, 2, 2},

{1, 1, 2, 2},

{1, 1, 2, 2},

{1, 1, 2, 2}}

Call function reduce (A, B)