****

**03 – Punto de Control C++**

Semana 03

**Nombre del estudiante:**

Daniel Isaac Juárez Funes

**Número de cuenta:**

12141153

**Sede de estudio:**

UNITEC TGU

**Docente:**

Ing. Martin Nelbren Cuellar

**Sección:**

CCC208 – Sec. 493

**Fecha de entrega:**

Lunes 08 de agosto del 2022

Ejercicio #04

#include <iostream>

using namespace std;

int main() {

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

int n[3] = { 10, 30, 50 };

int\* p = m[0];

for (int i = -3; i < 6; i++)

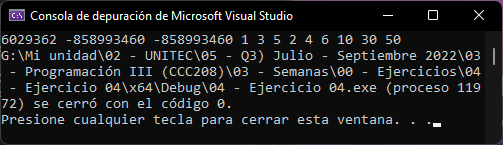
cout << p[i] << " ";

p[-1] = 777;

for (int i = 0; i < 3; i++)

cout << n[i] << " ";

}



Utilizamos un apuntador para referirnos a la matriz m. Los primeros tres números se generan de manera inesperada ya que el ciclo for que usamos para imprimir la matriz, comienza desde una posición inexistente en la matriz, una vez llega a la posición 0, comienza a imprimir los números definidos dentro de la matriz.

La segunda parte de la salida imprime de manera común y normal el arreglo n, empezando en su posición 0 hasta la posición 3.

Ejercicio #05

#include <iostream>

#include <iomanip>

using namespace std;

int main(int argc, char\* argv[]) {

int Filas, Columnas;

Filas = atoi(argv[1]);

Columnas = atoi(argv[2]);

long\*\* Matriz = new long\* [Filas];

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

Matriz[i] = new long[Columnas];

}

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

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

Matriz[i][j] = (i + 1) \* (j + 1);

if (((i + 1) \* (j + 1)) > 9) {

cout << setw(3) << "a[" << i << "]["

<< j << "]=" << setw(3) << (i + 1) \* (j + 1);

} else {

cout << setw(3) << "a[" << i << "]["

<< j << "]=" << setw(3) << (i + 1) \* (j + 1);

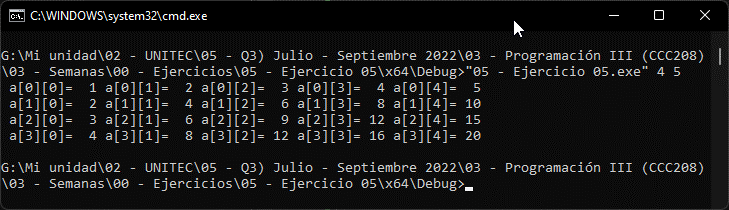
}

}

cout << endl;

}

}



Preguntas de Repaso

1. ¿Cuáles son los segmentos de memoria?
   * Segmento de Heap (HS), de Pila (SS), de Datos (DS) y de Código (CS).
2. El segmento donde están los objetos dinámicos:
   * Segmento de Heap (HS)
3. Desarrolle un programa que solicite 3 cadenas, las almacene en un arreglo de cadenas con máxima longitud de 80, y se pueda hacer búsqueda en el arreglo de cadenas.

#include <iostream>

#include <cstdlib>

using namespace std;

int main() {

char ArregloCadenas[3][80], lookup[80];

bool found = false;

//Llenado

for (int i = 0; i < 3; i++) {

cout << "Ingrese una Cadena: ";

cin.getline(ArregloCadenas[i], 80, '\n');

}

//Solicitar

cout << "\nCadena a Buscar: ";

cin.getline(lookup,80, '\n');

//Busqueda

for (int i = 0; i < 3; i++) {

if (strstr(lookup, ArregloCadenas[i]))

found = true;

}

if (found)

cout << endl << "La Cadena que Buscaba SI Existe!" << endl;

else

cout << endl << "Esa Cadena NO Existe" << endl;

}

