



UAEM

Universidad Autónoma
del Estado de México



Universidad Autónoma del Estado de México

Centro Universitario UAEM Atlacomulco

“Grafo dirigido e introducción a HTML 5”

Presenta:

Elias Edgardo Segundo Antonio

Revisa:

Ingeniero José Luis García Morales

Unidad de Aprendizaje:

Programación avanzada

Fecha de entrega:

4 de marzo de 2017



Inclusión de menú y función de seteo en el programa de grafos

Al programa creado en clase añadir un menú con las siguientes opciones

1. Pedir nuevos valores: Se solicitará una nueva matriz de adyacencia de orden 6
2. Test de adyacencia: Se pedirá origen y destino y se responder si existe conexión
3. Imprimir grafo
4. Salir

Implementar las secciones faltantes para lograrlo.

Código en C++- gd.cpp

```
#include <iomanip>
#include <iostream>
#include <cstdlib>
using namespace std;
const int NODOS = 6;
void printGraph(int [NODOS][NODOS]);
int isConnected(int [NODOS][NODOS], int, int);
void newGraph(int [NODOS][NODOS]);

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

    int G[NODOS][NODOS] = {
        {0,0,1,0,0,0},
        {1,0,0,1,0,1},
        {0,1,0,0,0,0},
        {0,0,0,0,0,0},
        {0,1,0,1,0,0},
        {0,0,0,0,0,0}};
    int option,o,d;

    while(option!=4)
    {
        cout<<"Programa para la prueba de GD, estas son las opciones:\n";
        cout<<"1.-Pedir nuevos valores (se pedira una nueva matriz)\n";
        cout<<"2.-Verificar adyacencia de nodos (se solicitara origen y destino)\n";
        cout<<"3.-Imprimir Grafo\n";
        cout<<"4.-Salir\n";
        cout<<"Inserte el numero de la opcion deseada: ";
        cin>>option;
        switch(option)
        {
            case 1:
```



```

    newGraph(G);
    break;
case 2:
    cout<<"\nProporcione el origen (0-5) : ";
    cin>>o;
    cout<<"\nAhora proporcione el destino (0-5) : ";
    cin>>d;
    if(isConected(G,o,d))
    {
        cout<<"\n\nEn efecto, hay conexion\n\n";
    }
    else cout<<"\n\nNo existe conexion entre los nodos\n\n";
    break;
case 3:
    cout<<"\n\nEste es el grafo: \n\n";
    printGraph(G);
    break;
default:
    if(option!=4)
    {
        cout<<"\n\nPor favor, inserte una opcion valida\n\n";
    }
}

}
return 0;

}

void printGraph(int matrix[NODOS][NODOS]) {
    for (int i = 0; i < NODOS; i++) {
        for (int j = 0; j < NODOS; j++) {
            cout << matrix[i][j] << "\t";
        }
        cout << "\n";
    }
}

int isConected(int matrix[NODOS][NODOS], int o, int d){
    if(o <= NODOS || o >= NODOS || d <= NODOS || d >= NODOS )
        return matrix[o][d];
}

void newGraph(int G[NODOS][NODOS])
{
    int conection;
    for(int i=0;i<NODOS;i++)
    {

```



```

for (int j=0;j<NODOS;j++)
{
    cout<<"\n\nHay conexión de "<<i+1<<" a "<<j+1<<"?(0,1): ";
    cin>>conection;
    G[i][j]=conection;
}
}
}

```

Capturas

```

C:\Users\Elink\Documents\Materias\Programacion avanzada\Practica5
λ gd
Programa para la prueba de GD, estas son las opciones:
1.-Pedir nuevos valores (se pedira una nueva matriz)
2.-Verificar adyacencia de nodos (se solicitara origen y destino)
3.-Imprimir Grafo
4.-Salir
Inserte el numero de la opcion deseada: |

```

```

Inserte el numero de la opcion deseada: 3

Este es el grafo:
0      0      1      0      0      0
1      0      0      1      0      1
0      1      0      0      0      0
0      0      0      0      0      0
0      1      0      1      0      0
0      0      0      0      0      0

Programa para la prueba de GD, estas son las opciones:
1.-Pedir nuevos valores (se pedira una nueva matriz)
2.-Verificar adyacencia de nodos (se solicitara origen y destino)
3.-Imprimir Grafo
4.-Salir
Inserte el numero de la opcion deseada: |

```



```
Programa para la prueba de GD, estas son las opciones:
1.-Pedir nuevos valores (se pedira una nueva matriz)
2.-Verificar adyacencia de nodos (se solicitara origen y destino)
3.-Imprimir Grafo
4.-Salir
Inserte el numero de la opcion deseada: 2
Proporcione el origen (0-5) : 1
Ahora proporcione el destino (0-5) : 2
No existe conexion entre los nodos
```

```
Programa para la prueba de GD, estas son las opciones:
1.-Pedir nuevos valores (se pedira una nueva matriz)
2.-Verificar adyacencia de nodos (se solicitara origen y destino)
3.-Imprimir Grafo
4.-Salir
Inserte el numero de la opcion deseada: 1
Hay conexion de 1 a 1?(0,1): 1
Hay conexion de 1 a 2?(0,1): 1
Hay conexion de 1 a 3?(0,1): 1
Hay conexion de 1 a 4?(0,1): 0
Hay conexion de 1 a 5?(0,1): 0
Hay conexion de 1 a 6?(0,1): .....|
```



```
Hay conexion de 5 a 5?(0,1): 0
Hay conexion de 5 a 6?(0,1): 0
Hay conexion de 6 a 1?(0,1): 0
Hay conexion de 6 a 2?(0,1): 0
Hay conexion de 6 a 3?(0,1): 0
Hay conexion de 6 a 4?(0,1): 0
Hay conexion de 6 a 5?(0,1): 0
Hay conexion de 6 a 6?(0,1): 0
Programa para la prueba de GD, estas son las opciones:
1.-Pedir nuevos valores (se pedira una nueva matriz)
2.-Verificar adyacencia de nodos (se solicitara origen y destino)
3.-Imprimir Grafo
4.-Salir
Inserte el numero de la opcion deseada: 4

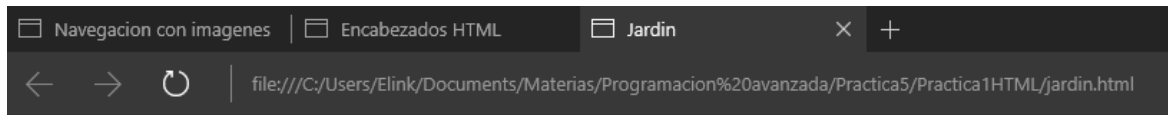
C:\Users\Elink\Documents\Materias\Programacion avanzada\Practica5
λ |
```

Introducción a HTML 5

Se realizaron los ejercicios recibidos por correo, para reforzar la práctica se trató de solo mirar las salidas en HTML y recrear el código con datos diferentes, después se revisó su parecido y se pulió el resultado. Las capturas se presentan a continuación, en algunos casos se añadieron estilos diferentes la página, la carpeta de los archivos se envió adjunta.



Capturas



Jardin bello

Apoyando la belleza del entorno desde 2001

Bienvenidos al JARDIN BELLO esperamos que pueda encontrar sus productos

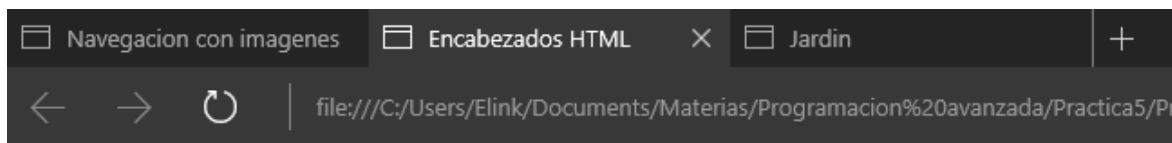
Vengan a nuestra tienda

Nuestra direccion es:

Calle de las flores num 12 San Lorenzo Tlacotepec Atlacomulco, Mex

Puede visitarnos desde los *Lunes a sabado desde las 7 a 8¹*

Nota 1 Cerramos a las 6 el 1° de enero



Encabezado

Encabezado

Encabezado

Encabezado

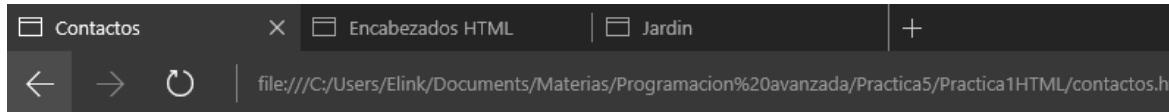
Encabezado

Encabezado

Practica 4



Dar clic en alguna imagen



[Envia un correo a Web & Development](#)

Toda la información en este sitio contiene derechos de © **Web & Development, Inc 2017**

~~Usted puede descargar 3.14×10^{-3} caracteres por valor de información de este sitio. El primer valor de la serie es X_1~~

Nota: $< \frac{1}{4}$ de la información presentada aquí se actualiza diariamente



Listas en HTML

Encabezados HTML

Jardín

+

←

→

↻

file:///C:/Users/Elink/Documents/Materias/Programacion%20Avanzada/Practica5/Practica%20HTML/listas.html

Tecnologias para desarrollo web

Tecnologias basicas

1. HTML 5
2. Javascript
3. CSS3
4. Bases de datos SQL y NoSQL

Tecnologias y tematicas complementarias basicas

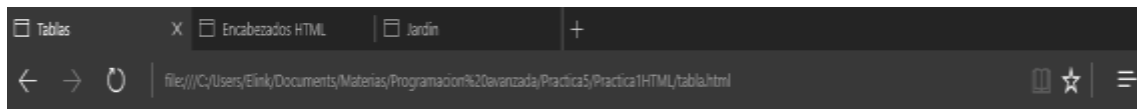
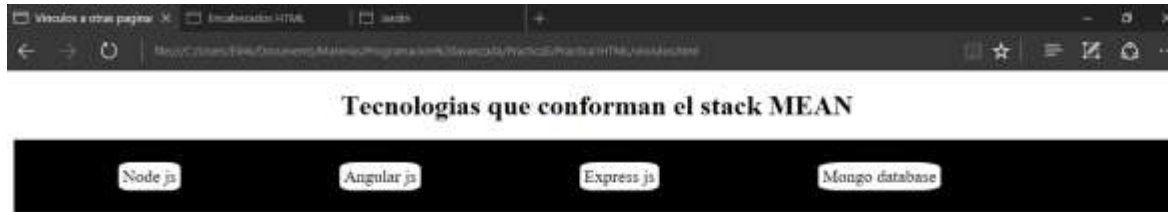
- Arquitectura de aplicaciones
- Sistemas de versiones GIT
- Programacion orientada a objetos
- Servidores

Frameworks de apoyo

1. Express js
 - Javascript
2. Django
 - Python
3. Laravel
 - PHP
 - Blade
4. Ruby on rails
 - Ruby
5. Angular js
 - Javascript
 - Node js

Stacks famosos

Practica 4



Lenguajes del curso de programacion, lenguaje, paradigma y duracion

Lenguaje	Paradigma	Tiempo del curso
Python	Multiparadigma	1 mes
Haskell	Funcional	2 meses
Tiempo total		3 meses



Formulario de retroalim

×

Encabezados HTML

|

Jardin

←

→

↻

|

file:///C:/Users/Elink/Documents/Materias/Programacion

Formulario de retroalimentacion

Ayudenos llenando este formulario

Nombre:

Comentarios:

E-mail:

Cosas que le gustaron

Contenido ☐ Descripciones ☐ Material de apoyo ☐ Tematica del sitio ☐

Como conocio el sitio

Busqueda en internet ☒ Amigos ☐ Correo ☐ Direccion de la pagina ☐

Califique nuestro sitio

1

Enviar

Borrar