

//Nombre del curso



Adrian Coello, Ariana Cordero, Valeria Martinez, Diego Orellana, Sebastian Nuñez

Ingeniería de Sistemas Computacionales, Universidad Privada Boliviana

Facultad de Ingeniería y Arquitectura

Algorítmica I

Ing. Paul Wilker Landaeta Flores

13 de mayo de 2023

1. Introducción

El proyecto tiene como propósito facilitar la experiencia de los estudiantes de las promociones que visitan la Universidad Privada Bolivariana (UPB) al permitirles recorrer y explorar las diversas exposiciones con los jefes de carrera de cada una, así como alcanzar la mesa de exposición de la carrera que tienen en mente. Para lograr esto, se empleará el algoritmo de Dijkstra, el cual optimizará las rutas tomando en cuenta tanto la distancia como el tiempo de exposición de los docentes de cada carrera, asegurando así una visita eficiente y completa para que pueda llegar a la carrera que el estudiante tenga en mente. Además, se detallarán los requisitos de software necesarios y se proporcionará una guía de instalación para ejecutar el proyecto, el cual fue desarrollado en lenguaje C++ , utilizando el entorno de Visual Studio Code y implementando las herramientas de Git y Github.

2. Problemática

El proyecto está diseñado para un problema común en el Departamento de Admisiones (DAAE), con el objetivo de mejorar la experiencia de los estudiantes de las promociones que visitan la universidad ya que la mayoría de las veces los estudiantes van por las carreras más cercanas y no llegan a la mesa de la carrera que desean por motivos de tiempo límite. Al utilizar el algoritmo de Dijkstra, la aplicación permite a los estudiantes recorrer y explorar las exposiciones universitarias de manera eficiente, incluyendo la mesa de exposición de la carrera que tienen en mente considerando el tiempo de exposición de los jefes de cada carrera, asegurando una visita completa y enriquecedora para que logren tener la mejor decisión. Esta solución optimiza la planificación de la visita, proporcionando una experiencia personalizada y mejorada para los estudiantes.

3. Requisitos de software

El proyecto se realizó en lenguaje C++ y se desarrolló en el editor Visual Studio Code, por lo que se recomienda utilizar el mismo por la comodidad en su ejecución.

Para realizar la correcta instalación de C++ recomendamos seguir el siguiente tutorial:

 [C/C++ en Visual Studio Code | C++ Extension](#)

Previamente asegurarse de tener todas las extensiones necesarias para que el programa compile de manera correcta para su uso.

También será necesario el tener conocimiento básico en GitHub y contar con una cuenta para poder clonar el repositorio. Para guiarse en esta parte proponemos ver el siguiente tutorial para no tener problemas al clonarlo:

 [Clonar un repositorio Git de GitHub](#)

4. Instrucciones para ejecutar el programa

Paso 1: Instalar Git

Primero, necesitas tener Git instalado en tu máquina. Si no lo tienes, puedes descargarlo desde la página oficial de Git. Sigue las instrucciones de instalación proporcionadas en la página.

[Download Git](#)

Paso 2: Clonar el Repositorio

Windows 11:

Una vez que Git está instalado, puedes clonar el repositorio en tu máquina local:

- Crea una carpeta en tu computadora.
- Dentro de la carpeta haga click derecho en la carpeta vacía.
- Busque la opción mostrar más opciones.
- Busque la sección que diga "open git Bash here".
- Una vez dentro coloca el siguiente comando.

```
git clone https://github.com/AdrianCoello/ProyectoFinal.git
```

macOS:

1. Crea una nueva carpeta en la ubicación de tu elección en tu máquina local.
2. Abre la Terminal (puedes encontrarla en Aplicaciones -> Utilidades).
3. Navega a la carpeta que acabas de crear utilizando el comando cd. Por ejemplo, si creaste una carpeta llamada ProyectoFinal en tu Escritorio, escribe `cd ~/Desktop/ProyectoFinal`.
4. Ahora, abre Git Bash en esta carpeta. Puedes hacerlo haciendo clic derecho en la carpeta y seleccionando Git Bash Here.
5. Finalmente, clona el repositorio con el siguiente comando:

```
git clone https://github.com/AdrianCoello/ProyectoFinal.git
```

Linux:

1. Crea una nueva carpeta en la ubicación de tu elección en tu máquina local.
2. Abre la Terminal.
3. Navega a la carpeta que acabas de crear utilizando el comando cd. Por ejemplo, si creaste una carpeta llamada ProyectoFinal en tu Directorio Personal, escribe `cd ~/ProyectoFinal`.
4. Ahora, abre Git Bash en esta carpeta. Puedes hacerlo haciendo clic derecho en la carpeta y seleccionando Open in Terminal.
5. Finalmente, clona el repositorio con el siguiente comando:

git clone <https://github.com/AdrianCoello/ProyectoFinal.git>

Paso 3: Instalar el Compilador

Para ejecutar el programa, necesitarás tener instalado el compilador correspondiente. Encontrarás las instrucciones detalladas para instalar el compilador en la carpeta Instalación del Compilador.

Paso 4: Explorar el Repositorio

Ahora puedes explorar las carpetas y archivos en el repositorio. Cada carpeta tiene un propósito específico y contiene un archivo README que explica su contenido.

Paso 5: Compilar y Ejecutar el Programa

Finalmente, puedes compilar y ejecutar el código del proyecto. Consulta el README en la carpeta Proyecto para más detalles. Para este apartado se tendrá el archivo input donde entraran estos datos:

0 1 9

0 2 14

0 3 10

0 4 15

1 5 13

1 6 11

1 7 8

1 13 10

2 8 12

2 9 8

4 10 15

4 14 10

4 16 14

6 11 8

6 12 11

9 13 15

14 15 9

14 16 11

14 17 8

15 5 10

16 15 11

16 17 14

Y donde en otro archivo output saldrá el camino mas corto o conveniente para que el estudiante no pierda su tiempo.

5. Explicación del algoritmo

El algoritmo de Dijkstra es una herramienta esencial donde este algoritmo busca el camino más corto dependiendo del grafo ponderado de las rutas (en términos de distancia y tiempo), en nuestro contexto las promociones con el algoritmo de Dijkstra podrán encontrar la ruta más corta o el camino más corto entre los nodos de un grafo en nuestro caso las mesas de exposición y también a todos los otros nodos del grafo, generando un árbol del camino más corto.

6. Ejemplos

Ejemplo 1: Estudiante quiere visitar lugares específicos.

Estudiante: María, una estudiante de secundaria interesada en carreras de ingeniería y negocios.

Objetivo: Visitar los stands de los programas de Ingeniería Informática, Ingeniería Mecánica y Administración de Empresas.

Ubicación actual: La entrada principal UPB.

Tiempo disponible: 1 hora

Ruta optimizada:

1. Entrada principal (0 minutos)
2. Stand de Ingeniería Informática (10 minutos de exposición)
3. Stand de Ingeniería Mecánica (15 minutos de exposición)
4. Stand de Administración de Empresas (20 minutos de exposición)

Tiempo total: 45 minutos

Beneficios:

- * María puede visitar de manera eficiente los stands de los programas que más le interesan.
- * Tiene suficiente tiempo para interactuar con el profesorado y conocer cada programa.
- * Evita perder tiempo en stands que no son relevantes para sus intereses.

Ejemplo 2:

Un grupo de estudiantes desea visitar las exposiciones de todas las carreras. El algoritmo de Dijkstra calcularía la ruta óptima para visitar todas las mesas de exposición en el menor tiempo posible. La ruta podría ser:

1. Ingeniería Civil (12 minutos de exposición)
2. Ingeniería Industrial (15 minutos de exposición)
3. Ingeniería Electrónica (18 minutos de exposición)
4. Arquitectura (20 minutos de exposición)

5. Derecho (15 minutos de exposición)
6. Administración de Empresas (20 minutos de exposición)
7. Contaduría Pública (18 minutos de exposición)
8. Psicología (15 minutos de exposición)
9. Comunicación Social (12 minutos de exposición)

7. Conclusión

A través del algoritmo de Dijkstra, el proyecto “Optimización de visitas guiadas a ferias universitarias” ofrece una solución efectiva para los estudiantes que visitan la Universidad Privada Boliviana (UPB). Al optimizar los recorridos y tener en cuenta el tiempo de contacto de los docentes, se garantiza a los estudiantes la oportunidad de visitar las mesas expositivas de las ocupaciones que les interesan y obtener información completa y relevante para tomar una decisión informada sobre su futuro académico.

Beneficios del proyecto:

- * Mejora la experiencia de los estudiantes: los estudiantes pueden navegar eficientemente por las exposiciones y obtener información completa sobre las ocupaciones que les interesan.
- * Optimizar el tiempo de los profesores: los profesores pueden dedicar más tiempo a los estudiantes que están genuinamente interesados en sus carreras.
- * Reduce descontrol las mesas de exposición: Al distribuir uniformemente a los estudiantes, se evita el montón en las mesas de exposición.
- * Potencia la imagen de la universidad: La universidad se presenta como una institución innovadora y comprometida con mejorar la experiencia de sus estudiantes. de.