

Evidencia estructura de datos

Fernanda Cantú - A01782232 - Andrés Tarazona A01023332 - Alina Rosas - A01252720

Introducción

Las estructuras de datos se pueden comprender como una manera de organizar la información dentro de una computadora, describe el formato en el que los valores son almacenados, modificados y la relación entre estos, para así poder realizar operaciones de una forma eficiente.

Las estructuras de datos son importantes ya que es más sencillo manipular datos que están organizados, entonces esto simplifica la tarea de un programador. Además, permite tener herramientas para solucionar ciertos problemas y trabajar de manera más óptima y eficiente.

Existen diferentes tipos de estructuras de datos como pilas, colas y listas ligadas.

Metodología

Las listas circulares tienen como ventaja que puedes navegar al derecho y al revés. También evitan excepciones en operaciones que realizan ya que cada nodo siempre tiene uno anterior y uno siguiente, por lo tanto no existen casos especiales a menos que el programador lo implemente.

Por otro lado, los queues con prioridad son muy relevantes en sistemas que manipulan múltiples programas y sus ejecuciones, por ejemplo una computadora. También son importantes en los sistemas de navegación como el internet ya que priorizan datos para que la información importante llegue más rápido al lector.

Para la implementación de estos programas, se utilizó el lenguaje C y se trabajó en un repositorio de GitHub como se muestra a continuación:



Implementación - Lista Circular

```
Insertando el primer elemento en la lista circular: 1
Insertando un elemento en la lista circular: 2
Insertando un elemento en la lista circular: 4
Insertando un elemento en la lista circular: 3
Insertando un elemento en la lista circular: 4
Insertando un elemento en la lista circular: 5
```

```
Los elementos en la lista son los siguientes:
1
2
4
3
4
5
```

```
Insertando un elemento en la lista circular: 6
Insertando un elemento en la lista circular: 6
Insertando un elemento en la lista circular: 6
```

```
Buscando un elemento en la lista circular: 4
El elemento 4 ha sido encontrado en la lista circular en la posición 3
```

La suma de los números pares es: 28

Existen 3 elementos impares en la lista circular

```
Contando los elementos de la lista circular
La lista circular tiene 9 elementos
Eliminar:
Eliminando un elemento de la lista circular: 3
Eliminando un elemento de la lista circular: 1
```

```
Los elementos en la lista son los siguientes:
2
4
4
4
5
6
6
6
```

Eliminar repetidos:

```
Se elimina el nodo del medio de la lista: 6
Se elimina el nodo del medio de la lista: 6
Se elimina el nodo del final de la lista: 6
```

```
Los elementos en la lista son los siguientes:
2
4
4
5
```

```
Eliminando todos los elementos de la lista circular
La lista está vacía
```

Implementación - Priority Queue

```
Metiendo el número 3 con peso 3
Metiendo el número 5 con peso 4
Metiendo el número 1 con peso 3
Metiendo el número 4 con peso 5
Metiendo el número 1 con peso 4
Metiendo el número 9 con peso 6
Metiendo el número 3 con peso 7
Metiendo el número 4 con peso 3
Metiendo el número 1 con peso 3
Metiendo el número 6 con peso 3
Metiendo el número 1 con peso 6
Metiendo el número 6 con peso 6
```

La lista contiene los siguientes elementos:

```
3 con peso 7
6 con peso 6
1 con peso 6
9 con peso 6
4 con peso 5
1 con peso 4
5 con peso 4
3 con peso 3
6 con peso 3
1 con peso 3
3 con peso 3
2 con peso 2
5 con peso 1
4 con peso 3
```

Sacando el número 3 con peso 7

Push consecutivos:

La lista contiene los siguientes elementos:

```
6 con peso 6
1 con peso 6
9 con peso 6
4 con peso 5
1 con peso 4
5 con peso 4
3 con peso 3
6 con peso 3
1 con peso 3
4 con peso 3
1 con peso 3
3 con peso 3
2 con peso 2
5 con peso 1
5 con peso 1
```

Buscar nodos:

```
El número 5 está en la cola con peso 4, en la posición 5
El número 5 está en la cola con peso 1, en la posición 13
El número 5 está en la cola con peso 1, en la posición 14
El número 5 está en la cola con peso 1, en la posición 15
El nodo 5 no está en la cola
```

Contar nodos:

La cola tiene 17 nodos

Promedio por pesos:

El promedio de los nodos con peso 3 es 3

Sumas de un peso específico:

La suma de los nodos con peso 3 es 15

Eliminar:

La cola está vacía

Resultados

Nuestra propuesta de solución consiste en utilizar funciones para manejar estructuras de datos y poder ejecutar de forma correcta tanto la lista circular como la cola con prioridades.

Conclusiones

Como equipo, hemos tenido una experiencia enriquecedora conociendo y aprendiendo a trabajar con estructuras de datos. Hemos descubierto la importancia de las estructuras de datos para crear códigos más eficientes y para resolver problemas más complejos.

REFERENCIAS

- Qué son las estructuras de datos y por qué son útiles. (2022, 3 noviembre). OpenWebinars.net. <https://openwebinars.net/blog/que-son-las-estructuras-de-datos-y-por-que-son-tan-utiles/>
- Listas Circulares. (s. f.). http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro9/listas_circulares.html
- Priority Queues | Brilliant Math & Science Wiki. (s. f.). <https://brilliant.org/wiki/priority-queues/>