

A. STL Queue

A. Una estructura de datos que opera bajo el principio de First In First Out (FIFO), es decir que los datos salen en el orden en el que entran. Los datos se insertan en la parte de atrás de la estructura y se borran (salen) por la parte delantera.

B. Operaciones del STL Queue

A. `empty()` — Regresa si la Queue esta vacía.

B. `size()` — Regresa el tamaño de la Queue.

C. `swap()` — Intercambia los contenidos de ambas Queues, tienen que ser del mismo tipo pero los tamaños pueden ser diferentes.

D. `emplace()` — Inserta un elemento a la estructura, en la parte de atrás.

E. `front()` — regresa una referencia al primer elemento de la Queue.

F. `back()` — regresa una referencia al ultimo elemento de la Queue.

G. `push(x)` - agrega el elemento "x" al final del la Queue.

H. `pop()` - borra el primer elemento de la Queue.

C. use case scenario

B. STL Stack

A. Una estructura de datos que opera bajo el principio de Last In First Out (LIFO), comparado a la Queue, este es diferente porque lo ultimo que entra es lo primero que sale.

B. Operaciones del STL Queue

A. `empty()` — Regresa si el Stack esta vacía.

B. `size()` — Regresa el tamaño del Stack.

C. `top()` — Regresa una referencia al elemento que esta "arriba" del Stack.

D. `push(x)` — Agrega el elemento "x" en la parte superior de la Stack.

E. `pop()` — Borre el elemento de la parte superior del Stack.

C. STL Priority Queue

A. Es un tipo de Queue en el que el elemento con mayor prioridad va adelante de la Queue. Cada elemento tiene una prioridad asociada.

B. Operaciones del STL Priority Queue

A. `empty()` — Regresa un valor si la Priority Queue esta vacía.

B. `size()` — Regresa el tamaño de la Priority Queue.

C. `top()` — Accesar el elemento más grande de la Priority Queue.

D. `push()` — Agregar un elemento nuevo a la Priority Queue

E. `pop()` — Quitar el elemento más antiguo de la Priority Queue.

F. `value_type()` — Regresa el tipo de valor de la Priority Queue.

C. La prioridad se puede cambiar al crear una función que cambie la prioridad, cambiando la comparación de los elementos.