Lab-2

DS5300: Data Structures for Data Science

August 13, 2024

The following text is from the book "Data Structures and Algorithms in Python, Roberto Tamassia, Michael H. Goldwasser , Michael T. Goodrich".

1 Positional List

- p.element(): Return the element stored at position p.

  In the context of the positional list ADT, positions serve as parameters to some methods and as return values from other methods. In describing the behaviors of a positional list, we being by presenting the accessor methods supported by a list L:

- L.first(): Return the position of the first element of L,or None if L isempty.
- L.last(): Return the position of the last element of L,or None if L is empty.
- L.before(p): Return the position of L immediately before position p, or None if p is the first position.
- L.after(p): Return the position of L immediately after position p, or None if p is the last position.
- L.is empty( ): Return True if list L does not contain any elements.
- len(L): Return the number of elements in the list.
- iter(L): Return a forward iterator for the elements of the list. See Section 1.8 for discussion of iterators in Python.

  The positional list ADT also includes the following **update** methods:

- L.add_first(e): Insert a new element e at the front of L, returning the position of the new element.
- L.add_last(e): Insert a new element e at the back of L, returning the position of the new element.
- L.add_before(p, e): Insert a new element e just before position p in L, returning the position of the new element.

- L.add_after(p, e): Insert a new element e just after position p in L, returning the position of the new element.
- L.replace(p, e): Replace the element at position p with element e, returning the element formerly at position p.
- L.delete(p): Remove and return the element at position p in L, invalidating the position.

2 Queue

- Q.enqueue(e): Add element e to the back of queue Q.
- Q.dequeue(): Remove and return the first element from queue Q; an error occurs if the queue is empty.

  The queue ADT also includes the following supporting methods (with first being analogous to the stack's top method):

- Q.first(): Returna reference to the element at the front of queue Q, without removing it; an error occurs if the queue is empty.
- Q.is_empty( ):Return True if queue Q does not contain any elements.
- len(Q): Return the number of elements in queue Q; in Python, we implement this with the special method __len__.

3 Stack

- S.push(e): Add element e to the top of stack S.
- S.pop(): Remove and return the top element from the stack S; an error occurs if the stack is empty.

  Additionally, let us define the following accessor methods for convenience:

- S.top( ): Return a reference to the top element of stack S, without removing it; an error occurs if the stack is empty.
- S.is empty( ): Return True if stack S does not contain any elements.
- len(S): Return the number of elements in stack S; in Python, we implement this with the special method __len__.

4 References

- **Source Code:** https://github.com/mjwestcott/Goodrich