

BIG O TIMES OF SORTED-LIST FUNCTIONS:

SLCreateIterator Running Time:

- Best Case/Worst Case: $O(1)$
- Nothing is iterated, simply creates a list.

SLDestroy Running Time

- Best Case: $O(1)$
- Worst Case: $O(n)$

SLInsert Running Time:

- Best Case: $O(1)$
- Worst Case: $O(n)$

SLRemove Running Time:

- Best Case: $O(1)$
- Worst Case: $O(n)$

SLDestroyIterator Running Time:

- Best/Worst Case: $O(1)$

SLNextItem Running Time:

- Best/Worst Case: $O(1)$

SLGetItem Running Time:

- Best/Worst Case: $O(1)$

HOW TO RUN

You must supply your own main.c file.

Our main.c file has the ability to test integers and doubles.

While not implementing every single data type, the fact that our main function works with integers and doubles shows that it does work well..

To test other data types, you must have your own comparator functions.

Take a look at this as an example of a comparator function that can be passed into the create list.

This one can test integers.

The destruct method can be passed in as the destructFunctionT

Will only display correctly on linux. For some reason it works on Linux (XOR) Mac AND Windows

```
int compareIntegerType(void *p1, void *p2)
{
    int firstInt = *(int *)p1;
```

```

    int secondInt = *(int *)p2;

    if(firstInt < secondInt)
    {
        return -1;
    }
    else if(firstInt > secondInt)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}

void destruct(void *a)
{
    free(a);
}

```

NOTE

Removing an object that the iterator is pointing too will not change it. After removing an object that the iterator is pointing to, please make sure to call Get Next Item in order to properly update it.