Computer Information Systems 23 - Homework 8

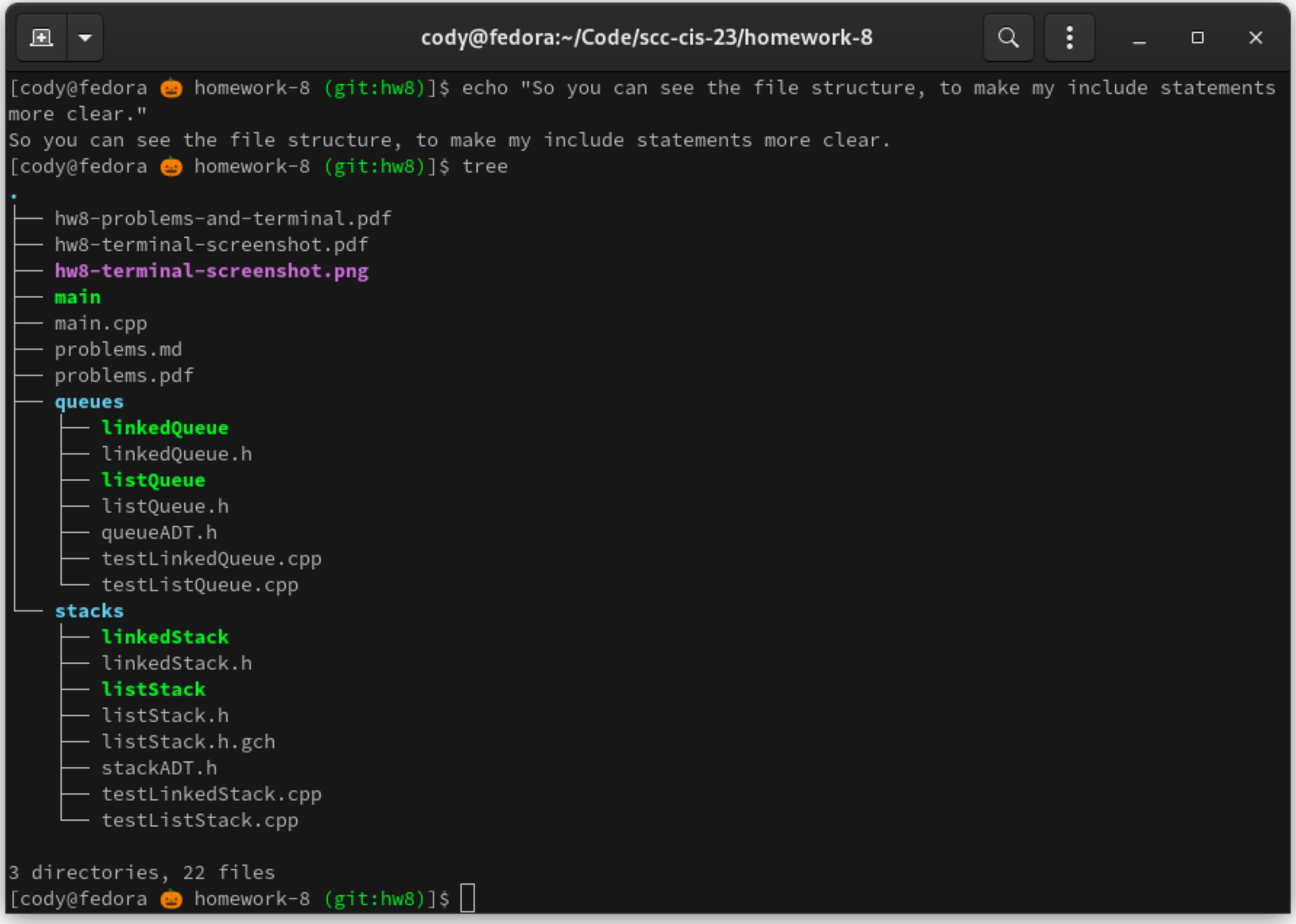
Cody Vig

Problem 1

(20 points) Suppose that stack is a object of type `linkedStackType<int>`. What is the difference between the statements `stack.top();` and `stack.pop();`?

Solution

`stack.top()` would return the last element in the stack without editing the stack itself. `stack.pop()` removes the last element from the stack all together.


cody@fedora:~/Code/scc-cis-23/homework-8
[cody@fedora 🎃 homework-8 (git:hw8)]$ echo "So you can see the file structure, to make my include statements more clear."
So you can see the file structure, to make my include statements more clear.
[cody@fedora 🎃 homework-8 (git:hw8)]$ tree
.
├── hw8-problems-and-terminal.pdf
├── hw8-terminal-screenshot.pdf
├── hw8-terminal-screenshot.png
├── main
├── main.cpp
├── problems.md
├── problems.pdf
├── queues
│   ├── linkedQueue
│   ├── linkedQueue.h
│   ├── listQueue
│   ├── listQueue.h
│   ├── queueADT.h
│   ├── testLinkedQueue.cpp
│   └── testListQueue.cpp
└── stacks
    ├── linkedStack
    ├── linkedStack.h
    ├── listStack
    ├── listStack.h
    ├── listStack.h.gch
    ├── stackADT.h
    ├── testLinkedStack.cpp
    └── testListStack.cpp
3 directories, 22 files
[cody@fedora 🎃 homework-8 (git:hw8)]$


cody@fedora:~/Code/scc-cis-23/homework-8
[cody@fedora 🎃 homework-8 (git:hw8)]$ g++ main.cpp -Wall -pedantic -std=c++11 -o main
[cody@fedora 🎃 homework-8 (git:hw8)]$ ls
main main.cpp problems.md problems.pdf queues stacks
[cody@fedora 🎃 homework-8 (git:hw8)]$ ./main 2
The old stack is: [9 5 1 4 1 3]
The new stack is: [3 1 4 1 5 9]
[cody@fedora 🎃 homework-8 (git:hw8)]$ ./main 3
Enter in a sentence:
Was it a car or a cat I saw
The input sentence is a palindrome!
[cody@fedora 🎃 homework-8 (git:hw8)]$ ./main 3
Enter in a sentence:
This is NOT a palindrome
The input sentence is not a palindrome.
[cody@fedora 🎃 homework-8 (git:hw8)]$