What is a runtime stack

What happens when the return statement is executed?

Run time stack

* Locals, parameters, return value

The heap

RECURSIVE MERGE SORT

BINARY SEARCH

QUICK SORT

Blocks

Q: Will you please explain to me the procedure of the recursion.py I’ve created?

Q: I know that self was referencing itself in the HW 2, and that makes the lists for the HashMap

be referenced by every methods. But, I would like to dig deeper on that.

Q: The recursion\_122 is just not working. Something about string indices and such.

Also, can we please go over the logical process of that file.

Q: What is a new activation record?

* Is Stack module one of the example of a run time stack?

Q: Let’s look at the Stack.py, that ANSWERS THE ABOVE QUESTION.

SHOULD DO:

* Let’s revisit the previous homeworks.

Linked list

Q: In the linked list def \_\_str\_\_(self), the return statement looks creative, I would like to analyze that.

Q: Along with the fact that if you look at the self.back.next = new, in my belief, self.back alone is a good enough statement to complete push\_back

Q: elif block for pop\_back: How can I think like a programmer?

In the HashMap

Q: Can we please analyze \_flattened?

Q: Can we please analyze if init?

Q:s[1], that’s really bizarre for me, can we please go over that?