| **SI SESSION PLAN** | | SI Leader: | Audrey Fuller | Session Date: | 2/27/23 | | --- | --- | --- | --- | | Week #: | 7 | Session Letter: | A | | Course & Section: | GCIS-123.4 | Course Instructor: | Professor Audi | | Planning Date: | 2/27/23 | Planning Time: | 9:55 | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**Beginning reminders:**

☑ Is the room set up in a way conducive to collaborative learning?

☑ Is the agenda posted to the board for participants to see?

☑ Do you have your attendance sheet up to record your attendance?

☑ Do you have any other documents/resources up and ready to go for your session?

If you are all set with the reminders, then go have fun and good luck!

**Is there a study strategy you want to focus on? (If so, what is it? Otherwise, leave blank.)**

**Main concepts student should feel more comfortable with:**

Arrays

* Fixed length, mutable data structure
* Contiguous block of memory
* Indexed 0 to length-1
* Is initialized with either 0 or NULL
* Can be accessed by calling index, array\_name[index] = value
* Not sequences, can’t be used with for loops
* import arrays (to mimic arrays)
* array\_name = arrays.Array(length, [optional prototype])
* len(array\_name)

Random

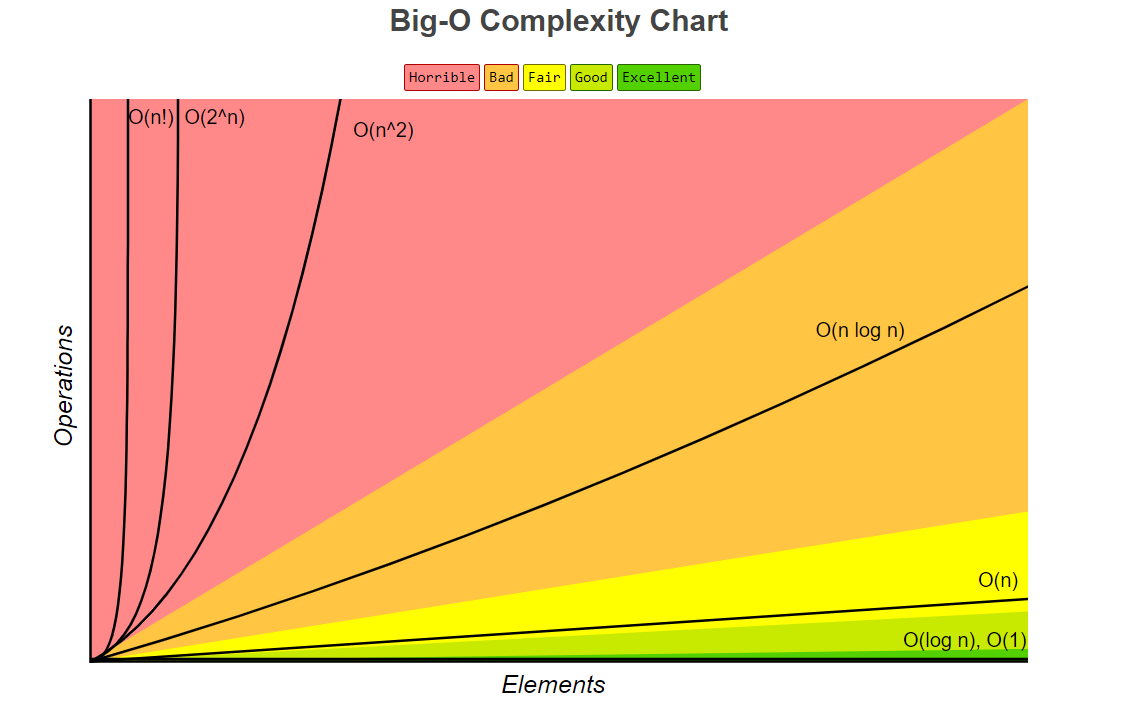
* import random
* randrange(start, stop, step=1) - returns a random integer from the specified range. The maximum possible value is stop-1.
* randint(a, b) - returns a random integer between a and b (inclusive).
* random() - returns a random floating point value between 0 and 1 (inclusive).
* random.seed(seed)

Linear Searching Algorithm

* Beginning at index 0 and compare the value at each index to the target. If one of the values matches the target, return the index. If you make it all the way through the array and never find the target, return None.
* O(n)

Timing & Time Complexity

* import time
* (begin = time.perf\_counter()) - (end = time.perf\_counter()) = elapsed time
* Can be used to predict time complexity (number of operations)
  + Time complexity is the *average* time it takes to complete an operation (scalars not important)



| **Activity\*** | **Process to use** | **Time** | **After Session Thoughts** |
| --- | --- | --- | --- |
| **Opener:**  **Kayak Question** | Ask each student that if they were to have a kayak, what color would they want ideally. Have them explain why and elaborate upon their response | 5 | | ☹ | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ☺ | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ | ☑ | ☐ | ☐ |   Quick, easy and fun activity. |
| **Vocab Review** | Complete this jamboard to go over vocab <https://jamboard.google.com/d/1jh-jq1USJYA0o0nT2FGs765gXBTPjP4kyPAWwpkhv5Y/edit?usp=sharing> | 25 | | ☹ | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ☺ | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | ☐ | ☐ | ☐ | ☑ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |   Took a lot longer than anticipated. Was also meant to be an interactive jamboard where people would brain dump all of the vocab they remembered on the board and go through defining them. But unfortunately the one student present was not a very active participant, and every time I would try to reframe the question to probe them for what they knew on the topic, I would get nowhere. |
| **Arrays, Random & Linear Search Coding Activity** | Complete coding activities as a group, including making an even array, finding the first duplicate, replacing character, creating random arrays and performing and timing a linear search. <https://github.com/alf9310/GCIS-123-4-SI-Sessions/blob/185a16a77dc3fff9a78566d6062483de8036f08c/week07/arrays&search.py> | 25 | | ☹ | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ☺ | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | ☐ | ☐ | ☑ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |   Code has an error, storing duplicates in a string did not work for multi-digit ints. Thankfully figured this out before the end of the session, but was still a bit of a hold-up. Additionally, there was the same problems as above where the student was not very cooperative in helping structure the code line-by-line. |
| **Closer: Other resources Review** | * Tutoring Center (GOL-2410) * WiC tutoring * Online review videos (ex. <https://www.youtube.com/watch?v=ixdr6V2vRC4>) | 5 | | ☹ | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ☺ | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ | ☑ | ☐ | ☐ |   Good information that I think the student appreciated hearing about. |

*\*See the* [*Activity Database*](https://docs.google.com/spreadsheets/d/1Oc6uAX2Uaq2Ym6M1FQjivRI_ryA_T9k1AcEKi__3Ml4/edit?usp=sharing) *and* [*SI Share*](https://drive.google.com/drive/folders/1WKkkRXpRW6_OVdc4eFVgAkDRt7y8E_VT?usp=sharing) *for ideas.*

**Ending reminders:**

☑ Did you mark down attendance on your attendance sheet?

☑ Did you remind everyone of the next session and any upcoming tests or quizzes or due dates?

☑ Did you fill in the after session thoughts?

**Optional Notes and Comments:**

**Bi-Weekly Question:** What resources are available for students in your course outside of SI? Have you encouraged students to utilize resources that may be beneficial for them?

Some available resources would be the Golisano tutoring center, Women in Computing Tutoring (not just for gender minorities), as well as online resource videos. I haven't mentioned these resources unless a student specifically asks me for homework help, but at the end of this week's session as a closer I’m planning on going over it.

**WELCOME TO SI SESSION WEEK 7!**

**Agenda:**

**Kayak Question**

**Vocab Review**

**Arrays, Random & Linear Search Coding Activity**

**Other Resources Review**