Stacks: Takeaways ⤴

Syntax

- Stack implementation:

```python
class Stack(LinkedList):
    def push(self, data):
        self.append(data)
    def peek(self):
        return self.tail.data
    def pop(self):
        ret = self.tail.data
        if self.length == 1:
            self.tail = self.head = None
        else:
            self.tail = self.tail.prev
            self.tail.next = None
        self.length -= 1
        return ret
```

- LCFS algorithm:

```python
cur_time = 0
num_processes_done = 0
wait_stack = Stack()
cur_pid = None
while num_processes_done < processes.shape[0]:
    if cur_pid is not None:
        if processes.loc[cur_pid, "Start"] + processes.loc[cur_pid, "Duration"] == cur_
            processes.loc[cur_pid, "End"] = cur_time
            cur_pid = None
            num_processes_done += 1
    ready_processes = processes[processes["Arrival"] == cur_time]
    for pid, _ in ready_processes.iterrows():
```

```
        wait_stack.push(pid)
    if cur_pid is None and len(wait_stack) > 0:
        cur_pid = wait_stack.pop()
        processes.loc[cur_pid, "Start"] = cur_time
    cur_time += 1
```

Concepts

- Stacks are a last-in, first-out (LIFO) data structure. This means that they work like a stack of plates in a restaurant. Dirty plates are put on top, and the washer takes the top plate to wash. So the last dirty plate to arrive is the first to get cleaned.

- We can implement stacks by extending the linked list data structure. When we extend a class in Python, all the attributes and methods are automatically available to the new class.

- Inheritance (class extension) is a very powerful programming tool because it allows transfer functionality from one class to another. It promotes code re-usability.

- The LCFS scheduling algorithm is an algorithm for scheduling usage of a single shared resource. Usage is granted in a last-come, first-served fashion. It can have advantages in situations where the age of processes is related to their urgency.

- The wait time of a process is the time between when the process arrives and the time when it starts being executed. The wait times in the LCFS algorithm are longer than with the FCFS algorithm. It can even be infinite if the set of processes is not finite.

- The turnaround time of a process is the time between when the process arrives and the time when it terminates.

Resources

- Stack
- Inheritance
- LIFO