OPERATING SYSTEMS

SWE - 208

Oğuzhan Yıldız	B211202029
Nurullah Kapancı	B211202068
Halit Ertaş	B2112020
Yunus Emre Yiğit	B2112020

halidBinVelid/SWE208OperatingSystems(github.com)

1. Introduction

This project involves developing a CPU processor scheduler that assigns processes to two CPUs based on certain criteria. The scheduler will check resource availability before assignment, prioritize processes and use different scheduling algorithms. The main goal is to efficiently manage processes in a simulated environment with limited resources.

2. System Design

- ## Overview

  The scheduler program reads from an input file containing a list of processes with certain attributes. Processes are assigned to two CPUs according to their priority and resource requirements. The assignment is guided by rules that ensure optimal utilization of CPU and RAM.

- ## Resources

  - *CPU-1:* Dedicated to the highest priority processes (priority 0).
  - *CPU-2:* Manages user processes with priority 1 (high), 2 (medium) and 3 (low).

- ## Scheduling Algorithms

  - *CPU-1:*
    - First-Come, First-Served (FCFS)
  - *CPU-2:*
    - *High priority user processes (priority 1):* Shortest Job First (SJF)
    - *Medium priority user processes (priority 2):* Round Robin (RR) with a quantum of 8
    - *Low priority user processes (priority 3):* Round Robin (RR) with a quantum of 16

3. Process Flow

- **Resource Check**

  Before assigning processes to CPUs, the scheduler checks if sufficient resources (RAM and CPU) are available. RAM is limited to 2048 MB, with 512 MB reserved for priority 0 processes.

- **Process Assignment**

  - *Input Reading:* Extract process details by parsing the input file.
  - *Resource Allocation:* Check and allocate the required resources for each process.
  - *Queuing:* Placing processes in appropriate queues according to their priorities and scheduling algorithm.
  - *Execution:* Assigning processes to CPUs and managing them according to defined scheduling algorithms.

4. Function Descriptions

- **Main Function**

  - *Function Name:* main
  - *Purpose:* Initializes the program, reads the input file, and starts the scheduling process.
  - *Key Operations:*
    - Reading input file
    - Initializing resource structures
    - Starting the scheduling loop

- **Resource Check Function**
  - *Function Name:* check_resources
  - *Purpose:* Verifies if sufficient resources are available for a given process.
  - *Key Operations:*
    - Checking RAM and CPU availability
    - Reserving resources for priority 0 processes

- **Scheduling Functions**
  - FCFS for CPU-1
    - *Function Name:* schedule_fcfs
    - *Purpose:* Schedules processes for CPU-1 using the FCFS algorithm.
  - SJF for High Priority Queue
    - *Function Name:* schedule_sjf
    - *Purpose:* Schedules high priority user processes using the SJF algorithm.
  - RR for Medium Priority Queue
    - *Function Name:* schedule_rr_medium
    - *Purpose:* Schedules medium priority user processes using RR with a quantum of 8.
  - RR for Low Priority Queue
    - *Function Name:* schedule_rr_low
    - *Purpose:* Schedules low priority user processes using RR with a quantum of 16.

5. Output

The output of the scheduler is written to output.txt, detailing the sequence of operations for each process. The log includes:

- Queueing of processes
- Assignment to CPUs
- Completion and termination of processes
- **Sample Output:**

----CPU-1 FIFO Algorithm----
Process P2 is queued to be assigned to CPU-1.
Process P2 is assigned to CPU-1.
Process P2 is completed and terminated.
Process P4 is queued to be assigned to CPU-1.
Process P4 is assigned to CPU-1.
Process P4 is completed and terminated.
Process P7 is queued to be assigned to CPU-1.
Process P7 is assigned to CPU-1.
Process P7 is completed and terminated.
Process P8 is queued to be assigned to CPU-1.
Process P8 is assigned to CPU-1.
Process P8 is completed and terminated.
Process P9 is queued to be assigned to CPU-1.
Process P9 is assigned to CPU-1.
Process P9 is completed and terminated.
Process P11 is queued to be assigned to CPU-1.
Process P11 is assigned to CPU-1.
Process P11 is completed and terminated.
Process P17 is queued to be assigned to CPU-1.
Process P17 is assigned to CPU-1.
Process P17 is completed and terminated.
Process P18 is queued to be assigned to CPU-1.

Process P18 is assigned to CPU-1.

Process P18 is completed and terminated.

Process P20 is queued to be assigned to CPU-1.

Process P20 is assigned to CPU-1.

Process P20 is completed and terminated.

----CPU-2 Shortest Job First (SJF) Algorithm----

time: 0, processing P1 (Remaining Burst Time: 2)

time: 1, processing P1 (Remaining Burst Time: 1)

time: 2, process P1 completed

time: 6, processing P14 (Remaining Burst Time: 2)

time: 7, processing P14 (Remaining Burst Time: 1)

time: 8, process P14 completed

time: 8, processing P15 (Remaining Burst Time: 4)

time: 9, processing P15 (Remaining Burst Time: 3)

time: 10, processing P15 (Remaining Burst Time: 2)

time: 11, processing P15 (Remaining Burst Time: 1)

time: 12, process P15 completed

time: 24, processing P25 (Remaining Burst Time: 2)

time: 25, processing P25 (Remaining Burst Time: 1)

time: 26, process P25 completed

----CPU-2 Round Robin Algorithm (queue 2, Quantum time: 8)----

time 1: process P5 is assigned to CPU-2.

time 3: process P5 is completed.

time 3: process P6 is assigned to CPU-2.

time 6: process P6 is completed.

time 6: process P10 is assigned to CPU-2.

time 14: process P10 quantum time is up, remaining burst time: 32

time 14: process P19 is assigned to CPU-2.

time 22: process P19 quantum time is up, remaining burst time: 2

time 22: process P10 is assigned to CPU-2.

time 30: process P10 quantum time is up, remaining burst time: 24

time 30: process P23 is assigned to CPU-2.

time 38: process P23 quantum time is up, remaining burst time: 25

time 38: process P19 is assigned to CPU-2.

time 40: process P19 is completed.

time 40: process P10 is assigned to CPU-2.

time 48: process P10 quantum time is up, remaining burst time: 16

time 48: process P23 is assigned to CPU-2.

time 56: process P23 quantum time is up, remaining burst time: 17

time 56: process P10 is assigned to CPU-2.

time 64: process P10 quantum time is up, remaining burst time: 8

time 64: process P23 is assigned to CPU-2.

time 72: process P23 quantum time is up, remaining burst time: 9

time 72: process P10 is assigned to CPU-2.

time 80: process P10 is completed.

time 80: process P23 is assigned to CPU-2.

time 88: process P23 quantum time is up, remaining burst time: 1

time 88: process P23 is assigned to CPU-2.

time 89: process P23 is completed.

All processes are done. whole time: 89

----CPU-2 Round Robin Algorithm (queue 3, Quantum time: 16)----

time 1: process P3 is assigned to CPU-2.

time 3: process P3 is completed.

time 5: process P12 is assigned to CPU-2.

time 7: process P12 is completed.

time 7: process P13 is assigned to CPU-2.

time 9: process P13 is completed.

time 9: process P16 is assigned to CPU-2.

time 13: process P16 is completed.

time 16: process P21 is assigned to CPU-2.

time 19: process P21 is completed.

time 19: process P22 is assigned to CPU-2.

time 21: process P22 is completed.

time 23: process P24 is assigned to CPU-2.

time 25: process P24 is completed.

All processes are done. whole time: 25

6. Conclusion

The CPU processor scheduler efficiently assigns processes to CPUs according to priority and resource requirements. The use of different scheduling algorithms ensures optimal CPU utilization and process management. The project illustrates key concepts in operating system design such as process scheduling, resource allocation and queue management.