COMP2432 Operating Systems

Assignment 3 (Tutorial 8)

Author
Wang Yuqi
[redacted]

Lecturer
Dr. MOHAMMED Aquil Mirza

BsC in Computer Science
Academic year: 2024/2025 Sem 2

Solutions

Solution to Question 1

Solution to Question 1(a) to (c)

FIFO (Frame = 3)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 2 2 2 6 6 6 0 0 0 5 5 5 2 2 2 3
  4 4 4 4 3 3 3 2 2 2 1 1 1 6 6 6 4 4 4
    1 1 1 1 5 5 5 4 4 4 3 3 3 0 0 0 1 1
M M M H M M M M M M M M M M M M M M M M -- Page Faults = 19
```

OPTI (Frame = 3)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 0 3 5 6 6 6 6 6 6 6 6 6 2 4 1 3
  4 4 4 4 4 4 4 4 4 0 0 0 0 0 0 0 0 0 0
    1 1 2 2 2 2 2 2 2 1 3 5 5 5 5 5 5 5
M M M H M M M M H H M M M M H H M M M M -- Page Faults = 15
```

LRU (Frame = 3)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 2 2 2 6 6 6 0 0 0 5 5 5 2 2 2 3
  4 4 4 4 4 5 5 5 4 4 4 3 3 3 0 0 0 1 1
    1 1 1 3 3 3 2 2 2 1 1 1 6 6 6 4 4 4
M M M H M M M M M M M M M M M M M M M M -- Page Faults = 19
```

FIFO (Frame = 4)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 0 3 3 3 3 3 0 0 0 0 6 6 6 6 1 1
  4 4 4 4 4 5 5 5 5 5 1 1 1 1 0 0 0 0 3
    1 1 1 1 1 6 6 6 6 6 3 3 3 3 2 2 2 2
        2 2 2 2 2 4 4 4 4 4 5 5 5 5 4 4 4
M M M H M M M M H M M M M M M M M M M M -- Page Faults = 18
```

OPTI (Frame = 4)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 4 1 3
  4 4 4 4 4 4 4 4 4 4 1 3 5 5 5 5 5 5 5
    1 1 1 3 5 6 6 6 6 6 6 6 6 6 6 6 6 6
        2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
M M M H M M M M H H H M M M H H H M M M -- Page Faults = 13
```

LRU (Frame = 4)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 0 3 3 3 3 4 4 4 4 5 5 5 5 4 4 4
  4 4 4 4 4 4 6 6 6 6 1 1 1 1 0 0 0 0 3
    1 1 1 1 5 5 5 5 0 0 0 0 6 6 6 6 1 1
        2 2 2 2 2 2 2 2 3 3 3 3 2 2 2 2
M M M H M M M M H M M M M M M M M M M M -- Page Faults = 18
```

(See Next Page)

Solution to Question 1(d)

FIFO (Frame = 3) after insert

New Reference String: [0, 4, 3, 1, 4, 2, 3, 5, 6, 2, 4, 0, 1, 3, 5, 6, 0, 2, 4, 1, 3], Page Faults: 18

```
0 4 3 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
------------------------------------------
0 0 0 1 1 1 1 1 6 6 6 6 1 1 1 6 6 6 4 4 4
  4 4 4 4 2 2 2 2 2 2 4 4 4 3 3 3 0 0 0 1 1
    3 3 3 3 3 5 5 5 5 0 0 0 5 5 5 2 2 2 3
M M M M H M H M M H M M M M M M M M M M M -- Page Faults = 18
```

LRU (Frame = 3) after insert

New Reference String: [0, 4, 1, 4, 2, 3, 2, 5, 6, 2, 4, 0, 1, 3, 5, 6, 0, 2, 4, 1, 3], Page Faults: 18

```
0 4 1 4 2 3 2 5 6 2 4 0 1 3 5 6 0 2 4 1 3
------------------------------------------
0 0 0 0 2 2 2 2 2 2 2 2 1 1 1 6 6 6 4 4 4
  4 4 4 4 4 4 5 5 5 4 4 4 3 3 3 0 0 0 1 1
    1 1 1 3 3 3 6 6 6 0 0 0 5 5 5 2 2 2 3
M M M H M M H M M H M M M M M M M M M M M -- Page Faults = 18
```

Answer:

- It is possible for **both LRU and FIFO**.

(See Next Page)

Solution to Question 2

CHP (Frame = 3, C = 4)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 2 3 3 3 3 4 4 4 4 5 6 6 6 6 1 1
  4 4 4 4 4 4 6 2 2 2 2 3 3 3 3 2 4 4 4
    1 1 1 1 5 5 5 5 0 1 1 1 1 0 0 0 0 3
M M M H M M M M M M M M M M M M M M M M -- Page Faults = 19
```

Predicted Cycles:

```
0 0 0 0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2
0 4 4 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4
0 0 1 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1
0 4 0 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
```

CHP (Frame = 3, C = 6)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 0 0 5 6 2 2 2 2 2 2 6 6 6 6 6 6
  4 4 4 4 4 4 4 4 4 0 0 0 0 0 0 2 4 1 1
    1 1 2 3 3 3 3 3 3 1 3 5 5 5 5 5 5 3
M M M H M M M M M H M M M M M H M M M M -- Page Faults = 17
```

Predicted Cycles:

```
0 0 0 0 0 0 4 1 4 2 3 5 6 2 4 0 1 3 5 6
0 4 4 4 4 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0
0 0 1 1 1 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2
0 4 0 4 4 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4
0 0 4 0 2 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1
0 4 1 4 0 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
```

CHP (Frame = 4, C = 4)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 0 3 3 3 3 4 4 4 4 5 5 5 5 4 4 4
  4 4 4 4 4 4 6 6 6 6 1 1 1 1 0 0 0 0 3
    1 1 1 1 5 5 5 5 0 0 0 0 6 6 6 6 1 1
        2 2 2 2 2 2 2 2 3 3 3 3 2 2 2 2
M M M H M M M M H M M M M M M M M M M M -- Page Faults = 18
```

Predicted Cycles:

```
0 0 0 0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2
0 4 4 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4
0 0 1 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1
0 4 0 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
```

CHP (Frame = 4, C = 6)

```
0 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
----------------------------------------
0 0 0 0 0 0 5 6 6 6 6 6 6 5 5 5 5 5 5 3
  4 4 4 4 4 4 4 4 4 0 0 0 0 0 0 2 4 4 4
    1 1 1 1 1 1 2 2 2 2 2 2 6 6 6 6 6 6
        2 3 3 3 3 3 3 1 3 3 3 3 3 3 1 1
M M M H M M M M M H M M M M M H M M M M -- Page Faults = 17
```

Predicted Cycles:

```
0 0 0 0 0 0 4 1 4 2 3 5 6 2 4 0 1 3 5 6
0 4 4 4 4 4 1 4 2 3 5 6 2 4 0 1 3 5 6 0
0 0 1 1 1 1 4 2 3 5 6 2 4 0 1 3 5 6 0 2
0 4 0 4 4 4 2 3 5 6 2 4 0 1 3 5 6 0 2 4
0 0 4 0 2 2 3 5 6 2 4 0 1 3 5 6 0 2 4 1
0 4 1 4 0 3 5 6 2 4 0 1 3 5 6 0 2 4 1 3
```

(See Next Page)

Solution to Question 3

Solution to 3(a)

No, there does not exist any safe sequences. After the allocation for request P_0, the remaining resources are $(1111) - (1011) = (0100)$. Afterwards, no process can be satisfied.

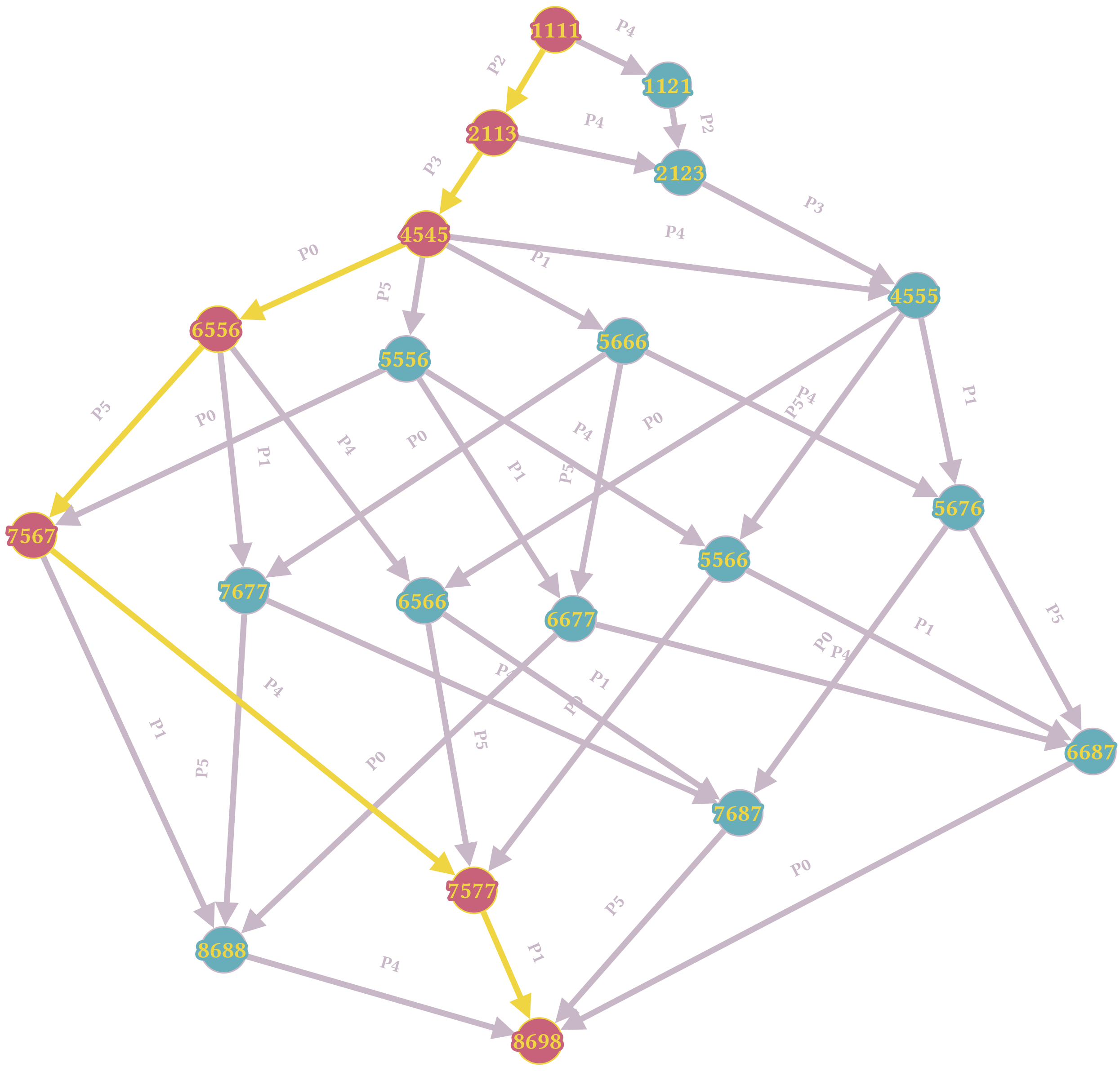

Figure 1: The Wait-For Graph for Question 3

Solution to 3(b)

If x (the last two bits) is $(00)_2$; then the processes are: P_0, P_4
This is **possible**:

- The increased (+1) needs of P_0 is (3534)
- The increased (+1) needs of P_4 is (2112)
- Both are satisfied by the path $P_2 \to P_3 \to P_0 \to P_5 \to P_4 \to P_1$
- **I.e., the highlighted path in Figure 1**

If x (the last two bits) is $(01)_2$; then the processes are P_1, P_5
This is **possible**:

- The increased (+1) needs of P_1 is (3413)
- The increased (+1) needs of P_5 is (2422)
- Both also satisfied by the path $P_2 \to P_3 \to P_0 \to P_5 \to P_4 \to P_1$
- **I.e., the highlighted path in Figure 1**

If x (the last two bits) is $(10)_2$; then the processes are P_2
This is **impossible**:

- The increased (+1) needs of P_2 is (2121)
- None of the P_2 edges in the graph satisfies "start node < 2121"
  - both (1121) and (1111) less than (2121)

If x (the last two bits) is $(11)_2$; then the processes are P_3 This is **impossible**:

- The increased (+1) needs of P_3 is (3113)
- None of the P_3 edges in the graph satisfies "start node < 3113"
  - both (2123) and (2113) less than (3113)

Note: we increase the values of the *need* by 1 for all types instead of a single type as specified in the question, because the specific type under-reported is corrupted. So the operating system has to consider the worst case where any type of resource can be under-reported. Enumerating each type of resource is equivalent to increasing the need of all types of resources by 1 and checking if the system is still in a safe state.

Answer:

- The processes are P_0, P_1, P_4, P_5
- Thus, X (the process number) can be 0, 1, 4, or 5

Solution to 3(c)

Consider $Y = A$. This is **impossible**:

- If P_2 need increase from $(1010) \to (2010)$

- Then, none of the edges in the graph satisfies "start node < 2010"
  - both (1111) and (1121) less than (2010)

Consider $Y = B$. This is **possible**:

- Because no matter which process is under-reported
- Path $P_2 \to P_3 \to P_0 \to P_5 \to P_4 \to P_1$ is always a safe sequence
- **I.e., the highlighted path in Figure 1**

Consider $Y = C$. This is **possible**:

- Because no matter which process is under-reported
- Path $P_2 \to P_3 \to P_0 \to P_5 \to P_4 \to P_1$ is always a safe sequence
- **I.e., the highlighted path in Figure 1**

Consider $Y = D$. This is **possible**:

- Because no matter which process is under-reported
- Path $P_2 \to P_3 \to P_0 \to P_5 \to P_4 \to P_1$ is always a safe sequence
- **I.e., the highlighted path in Figure 1**

(See Next Page)

Solution to Question 4

Solution to Question 4(a)-(c)

Figure 2: Memory allocation after each insertion based on *First-Fit* algorithm

Figure 3: Memory allocation after each insertion based on *Best-Fit* algorithm

Figure 4: Memory allocation after each insertion based on *Worst-Fit* algorithm

Solution to Question 4(d)

Allocation of P2	**Logical Address**	**Physical address for P1**	**Physical address for P2**
FF	(0, 44)	3055	1400
BF	(1, 231)	2132	3377
WF	(2, 82)	5760	6303
FF	(3, 199)	2631	1713
BF	(4, 56)	4490	2191
WF	(5, 304)	1315	5525
FF	(6, 135)	Invalid	2567