

OS Term Project #1 Report

- Process of Scheduling Queues -

컴퓨터공학과 2016110307 권동영

Scheduling Queues의 과정을 나타내는 프로그램은 C++로 작성하였으며, 총 3개의 Class로 구성되어 있다. 각각의 Class의 자세한 정보와 프로그램 동작 과정을 밑에 자세하게 설명하였다.

1. PCB Class

특정 Process를 관리할 정보를 포함한 OS Kernel의 자료구조이다. 원래는 많은 정보를 포함하고 있으나 이 프로그램은 3가지의 정보만 존재한다.

A. Member Variable

- **int m_pid**

- Process의 고유 번호이다. (Primary Key)
- PCB Instance를 만들 때마다 1씩 증가 된 숫자를 부여받는다.

- **int m_state**

- 아래의 같이 숫자를 통해 Process의 상태를 표현했다.
- 1 : New , Process가 처음으로 생성되는 상태
- 2 : Running , Process가 실행 중인 상태
- 3 : Waiting , Process가 실행 도중에 이벤트를 만나 대기하는 상태
- 4 : Ready , Process가 CPU의 자원을 받기 위해서 준비가 된 상태
- 5 : Terminated , Process가 끝날 때 보여주는 상태

- **PCB* m_pointer**

- 다음 실행 될 Process의 주소값이다.

B. Functions

- 멤버 변수를 Get, Set, Print 해주는 다수의 함수를 포함하고 있다.

2. Queue_Linkded_List Class

PCB를 손쉽게 관리하기 위해 Queue를 Linked-List로 표현한 자료구조이다. Job Queue, Ready Queue, Running Queue, Waiting Queue 총 4개의 Queue를 만들었다. Class의 자세한 정보는 운영체제의 수업과는 맞지 않으므로 보고서에는 생략하였고, Code의 주석으로 표시하였다.

3. Application Class

Scheduling Queues의 과정을 직접 구현해보도록 해주는 Class이다. 사용자가 Console에서 명령어를 입력하면 새로운 Process 생성, 모든 Process 상태 출력, Process를 다른 Queue로 이동, Process 종료를 구현하도록 되어있다.

A. Member Variable

- **int m_nCurCommand**
 - 사용자로부터 명령어를 입력받아 그에 맞는 함수를 구현해준다.
- **static int m_pid**
 - PID를 Process의 고유의 번호로 만들기 위한 전역변수
 - PCB Instance를 만들 때마다 1씩 증가하도록 구현하였다.
 - Ex. PCB1 → PCB2 → PCB3 → ...
- **QueueLinkedList<PCB> m_jobQue**
 - 모든 Process를 저장하고 있는 Queue.
 - Process를 종료할 때 까지 저장하고 있다.
- **QueueLinkedList<PCB> m_readyQue**
 - 실행 가능한 Process들이 CPU에 자원을 할당받기 위해 대기하는 Queue.
- **QueueLinkedList<PCB> m_waitingQue**
 - I/O 장치에서 Process를 처리하는 동안 대기하는 Queue.
- **QueueLinkedList<PCB> m_runningQue**
 - 실행 중인 Process를 저장하고 있는 Queue.
 - 하나의 Process만 저장하도록 구현했다.

B. Functions

- **AddProcess()**
 - 새로운 Process를 Job Queue에 추가하도록 구현했다.
- **DisplayStatus()**
 - 모든 Process의 현재 상태를 출력하도록 구현했다.
- **MoveJobToReady()**
 - Job Queue의 Process를 Ready Queue에 복사하도록 구현했다.
 - Job Queue에 존재하는 Process만 복사 가능하다.
 - 복사할 때 Job Queue를 제외한 다른 Queue에 존재하면 복사 불가능하도록 구현했다.
- **MoveReadyToRun()**
 - Ready Queue의 Process를 Running Queue로 이동하도록 구현했다.
 - Ready Queue에 존재하는 Process만 이동 가능하다.
 - 이동할 때, Running Queue에 이미 Process가 존재하면 이동 불가능하다.
- **MoveRunToWait()**
 - Running Queue의 Process를 Waiting Queue로 이동하도록 구현했다.
 - Running Queue에 존재하는 Process만 이동 가능하다.
 - 만약 이동하고 나서, Ready Queue에 Process가 존재하면 MoveReadyToRun()를 실행한다.
- **MoveWaitToReady()**
 - Waiting Queue의 Process를 Ready Queue로 이동하도록 구현했다.
 - Waiting Queue에 존재하는 Process만 이동 가능하다.

- **MoveRunToExit()**

- Running Queue의 Process를 종료시킨다.
- Running Queue에 존재하는 Process만 종료 가능하다.
- 종료된 Process는 Job Queue에서 제외되도록 구현했다.

- **MakeEmpty()**

- 현재 존재하는 모든 Process를 없애도록 구현했다.

4. 프로그램 동작 과정

```
→ OS_Project1 git:(master) x ./a.out

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

Choose a Command--> █
```

```
---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

Choose a Command--> 1

<==All Process State==>
--Current Job Que list--
PCB1(1),
--Current Ready Que list--

--Current Running Que list--

--Current Waiting Que list--
```

< Console 초기 화면 >

```
---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

Choose a Command--> 1

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1), PCB4(1),
--Current Ready Que list--

--Current Running Que list--

--Current Waiting Que list--
```

< Process 4개 생성 >

< 새로운 Process 생성 >

< PCB1(1)가 의미하는 것은 PID는 1이고 상태는 1(new)이다 >

```
---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

Choose a Command--> 3

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1), PCB4(1),
--Current Ready Que list--

--Current Running Que list--

--Current Waiting Que list--

Enter PID : █
```

< PID를 입력하면 ReadyQue로 복사 >

```

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1), PCB4(1),
--Current Ready Que list--

--Current Running Que list--

--Current Waiting Que list--

Enter PID : 1

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1), PCB4(1),
--Current Ready Que list--
PCB1(4),
--Current Running Que list--

--Current Waiting Que list--

```

< PCB1 ReadyQueue로 복사 >
 < PCB1 상태 변경 1(New) → 4(Ready) >

```

Choose a Command--> 3

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1), PCB4(1),
--Current Ready Que list--
PCB1(4), PCB4(4), PCB3(4), PCB2(4),
--Current Running Que list--

--Current Waiting Que list--

Enter PID : 5
No Exist PCB in Job Que

```

< JobQueue에 없는 PID입력으로 인한 오류 >

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

Choose a Command--> 3

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1), PCB4(1),
--Current Ready Que list--

--Current Running Que list--
PCB1(2),
--Current Waiting Que list--

Enter PID : 1
Exist Same PCB in another Que

```

< JobQueue를 제외한 나머지 Queue에 존재로 인한 오류 >

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

Choose a Command--> 4

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1), PCB4(1),
--Current Ready Que list--
PCB4(4), PCB3(4), PCB2(4),
--Current Running Que list--
PCB1(2),
--Current Waiting Que list--

```

< Ready Queue → Running Queue >
 < PCB1 상태 변경 4(Ready) → 2(Running) >

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

```

Choose a Command--> 4

Exist PCB in Running Que

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

```

Choose a Command--> 5

```

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1), PCB4(1),
--Current Ready Que list--
PCB3(4), PCB2(4),
--Current Running Que list--
PCB4(2),
--Current Waiting Que list--
PCB1(3),

```

< JobQue를 제외한 나머지 Queue에 존재로 인한 오류 >

< Running Queue → Waiting Queue >
 < PCB1 상태 변경 2(Running) → 3(Waiting) >
 < Ready Queue의 PCB4가 Running Queue로 이동 >
 < PCB4 상태 변경 4(Ready) → 2(Running) >

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

```

Choose a Command--> 5

No Exist PCB in Running Que

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

```

Choose a Command--> 6

```

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1), PCB4(1),
--Current Ready Que list--
PCB3(4), PCB2(4), PCB1(4),
--Current Running Que list--
PCB4(2),
--Current Waiting Que list--

```

< Running Queue의 Process의 부재로 인한 오류 >

< Waiting Queue → Ready Queue >
 < PCB1 상태 변경 3(Waiting) → 4(Ready) >

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

```

Choose a Command--> 7

```

<==All Process State==>
--Current Job Que list--
PCB1(1), PCB2(1), PCB3(1),
--Current Ready Que list--
PCB3(4), PCB2(4), PCB1(4),
--Current Running Que list--

--Current Waiting Que list--

```

< Running Queue의 Process 종료 >
< Job Queue에 PCB4 사라짐 >

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

```

Choose a Command--> 7

No Exist PCB in Running Que

< Running Queue의 Process의 부재로 인한 오류 >

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

```

Choose a Command--> 8

All Queues are now empty.

```

---ID -- Command -----
1 : Add new Process
2 : Display All Process Status
3 : Copy JobQue to ReadyQue
4 : Move ReadyQue to RunningQue
5 : Move RunningQue to WaitingQue
6 : Move WaitingQue to ReadyQue
7 : Move RunningQue to Termination
8 : Make empty all Process
0 : Quit

```

Choose a Command--> 2

```

<==All Process State==>
--Current Job Que list--

--Current Ready Que list--

--Current Running Que list--

--Current Waiting Que list--

```

< 모든 Process 종료 >