운영체제[2021-2학기] Project-01 [MFQ Scheduling Simulation]

2015310464 원종원

1. 가정

가정1. 동일한 RQ로 추가되는 프로세스가 두개 이상일 경우 PID가 작은 순으로 먼저 진입한다.(AT를 기준으로 더 먼저 들어온 프로세스를 먼저 진입시킨다면 TT를 줄이는데 도움이 될것이라고 생각했으나, 실제 운영체제에서는 AT만으로 우선순위를 정하지는 않으므로 차라리 random하게 PID를 기준으로 삼는것도 별반 다르지 않을 것이라 생각하였습니다.)

가정2. 프로세스의 CPU BT 과 I/O BT는 번갈아가며 실행되므로 input에 나와있는 프로세스의 CPU BT과 I/O BT에 0이 입력될 수 없다.

가정3. # of Ps는 최대 1만개로 가정.

가정4. 각각의 RQ는 선형큐로 되어있으며 사이즈는 1만으로 가정.

2. 설계 및 구현에 사용한 도구

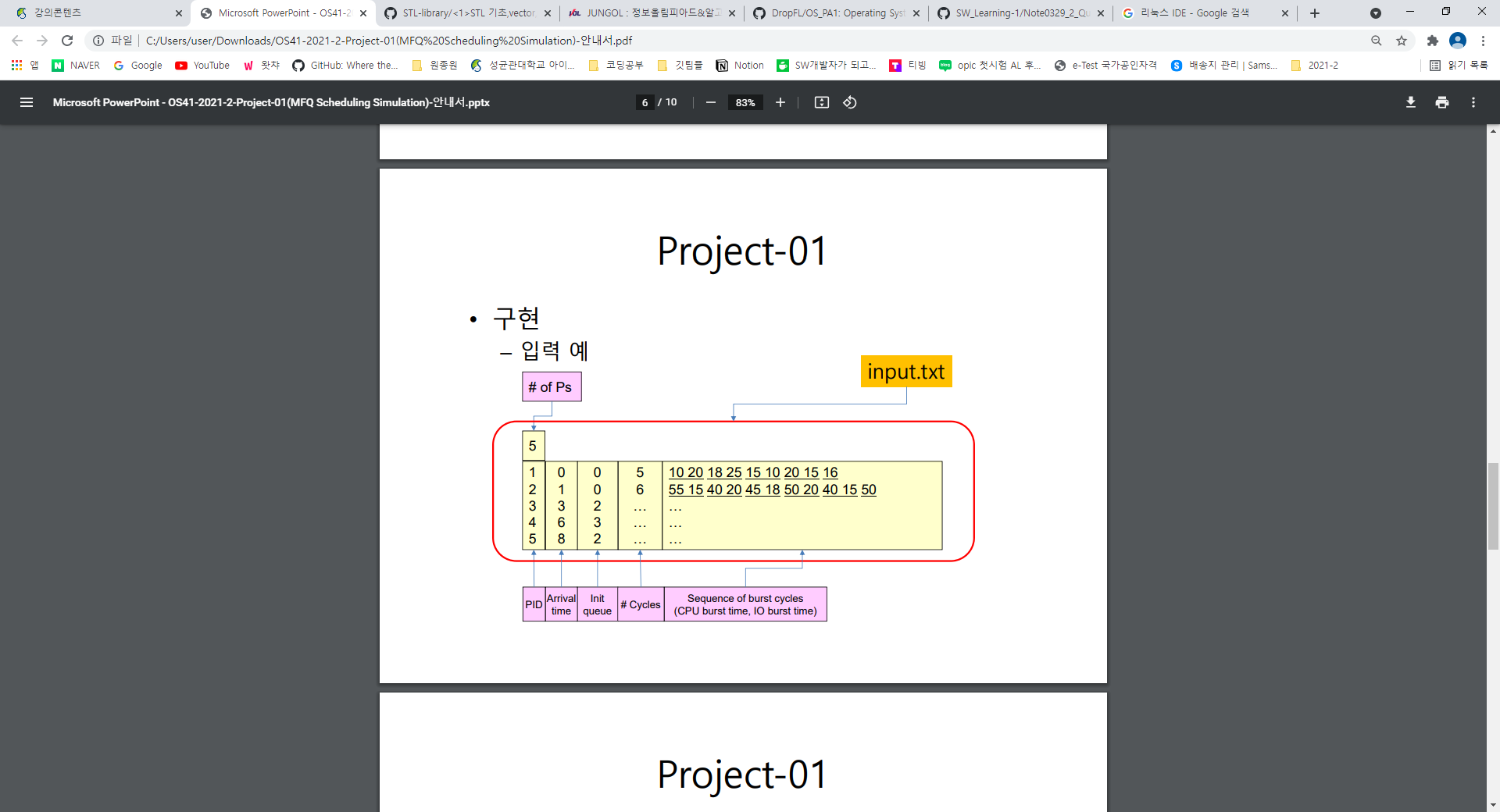
-사용언어 : C

-IDE : vscode

-MFQ 설계 배경지식 : 운영체제 3~4주차 수업 참고

-특이사항 : Sequence of burst cycles 또한 큐 자료구조를 활용하였습니다.

3. input에 대한 설명



과제 설명을 참고하여 동일한 방식으로 input을 구성하였습니다.

4. output에 대한 설명

기존에 교안에서 보았던 Gantt Chart와 달리 종적으로 볼 수 있도록 구성하였습니다.

첫번째 column은 time으로, cpu time quantum의 기준이 되도록 정수로 정의하였습니다.

다음 time이 등장하기 전까지 일어난 모든 행위는 모두 같은 time quantum안에서 일어난 것으로 간주합니다.

두번째 column은 현재 cpu에서 실행중인 프로세스의 pid를 나타냅니다. 반드시 하나의 quantum안에 하나의 프로세스만 등장하게 됩니다.

세번째 column은 RQ에 rear에 input으로 들어가는 (추가되는) 프로세스의 pid를 나타냅니다. 이때, 그 프로세스가 RQ에 처음 진입하는지, 혹은 time quantum이 끝나고 다시 RQ에 추가되는지, 혹은 I/O 작업을 마치고 다시 RQ에 추가되는지의 구분은 5번째 column ‘추가사항’을 통해 확인할 수 있습니다.

네번째 column은 세번째 column에 나타난 프로세스가 어떤 RQ에 진입하는지를 나타냅니다.

0인경우 RQ0에 진입하는 것을 의미합니다.(1,2,3 -> RQ1,RQ2,RQ3)

밑 부분에는 각 프로세스별 TT,WT과 전체 프로세스의 평균 TT,WT를 알아보기 쉽게 구성하였습니다. 이때 평균 TT와 WT는 소수점 1번째 자리까지 표시하였습니다

5. 실행 결과를 보는 방법

첨부한 input.txt 파일을 명령인수로 넣어 MFQ.c 파일을 실행하면 됩니다.

6. 실행 결과

<input-1>

3

1 2 1 3 1 2 1 5 1

2 1 1 3 3 1 4 2 4

3 1 1 2 1 1 2

<output-1>

Time 실행P 추가된P 진입RQ 추가사항

---------------------------------------------------------

<1>

2 1 새로운 P

3 1 새로운 P

2

---------------------------------------------------------

<2>

1 1 새로운 P

2

2 2 Preemption(남은 CPU BT: 1)

---------------------------------------------------------

<3>

3

3 프로세스 I/O진입

---------------------------------------------------------

<4>

3 0 IO burst 만료

1

1 프로세스 I/O진입

---------------------------------------------------------

<5>

3

3 1 Preemption(남은 CPU BT: 1)

---------------------------------------------------------

<6>

1 0 IO burst 만료

3

3 프로세스 completed

---------------------------------------------------------

<7>

1

1 프로세스 I/O진입

---------------------------------------------------------

<8>

2

2 프로세스 I/O진입

---------------------------------------------------------

<9>

IDLE

2 1 IO burst 만료

---------------------------------------------------------

<10>

2

---------------------------------------------------------

<11>

2

2 2 Preemption(남은 CPU BT: 2)

---------------------------------------------------------

<12>

1 0 IO burst 만료

2

---------------------------------------------------------

<13>

2

2 프로세스 I/O진입

---------------------------------------------------------

<14>

1

1 프로세스 completed

---------------------------------------------------------

<15>

IDLE

2 1 IO burst 만료

---------------------------------------------------------

<16>

2

---------------------------------------------------------

<17>

2

2 2 Preemption(남은 CPU BT: 2)

---------------------------------------------------------

<18>

2

---------------------------------------------------------

<19>

2

2 프로세스 completed

---------------------------------------------------------

<20>

실행 종료

PID TT WT

1 12 2

2 18 0

3 5 2

AVG.TT : 11.7

AVG.WT : 1.3

<input-2>

2

1 2 3 3 1 2 1 5 1

2 0 3 3 3 1 4 2 4

<output-2>

Time 실행P 추가된P 진입RQ 추가사항

---------------------------------------------------------

<0>

2 3 새로운 P

2

---------------------------------------------------------

<1>

2

---------------------------------------------------------

<2>

1 3 새로운 P

2

2 프로세스 I/O진입

---------------------------------------------------------

<3>

2 3 IO burst 만료

1

1 프로세스 I/O진입

---------------------------------------------------------

<4>

2

---------------------------------------------------------

<5>

1 3 IO burst 만료

2

---------------------------------------------------------

<6>

2

---------------------------------------------------------

<7>

2

2 프로세스 I/O진입

---------------------------------------------------------

<8>

1

1 프로세스 I/O진입

---------------------------------------------------------

<9>

IDLE

2 3 IO burst 만료

---------------------------------------------------------

<10>

2

---------------------------------------------------------

<11>

2

---------------------------------------------------------

<12>

2

---------------------------------------------------------

<13>

1 3 IO burst 만료

2

2 프로세스 completed

---------------------------------------------------------

<14>

1

1 프로세스 completed

---------------------------------------------------------

<15>

실행 종료

PID TT WT

1 12 1

2 13 0

AVG.TT : 12.5

AVG.WT : 0.5

<input-3>

3

1 0 1 3 1 2 1 1 2

2 2 3 2 1 1 1

3 4 2 2 2 2 2

<output-3>

Time 실행P 추가된P 진입RQ 추가사항

---------------------------------------------------------

<0>

1 1 새로운 P

1

1 프로세스 I/O진입

---------------------------------------------------------

<1>

IDLE

---------------------------------------------------------

<2>

2 3 새로운 P

1 0 IO burst 만료

2

2 프로세스 I/O진입

---------------------------------------------------------

<3>

2 3 IO burst 만료

1

1 프로세스 I/O진입

---------------------------------------------------------

<4>

3 2 새로운 P

1 0 IO burst 만료

3

---------------------------------------------------------

<5>

3

3 프로세스 I/O진입

---------------------------------------------------------

<6>

1

1 1 Preemption(남은 CPU BT: 1)

---------------------------------------------------------

<7>

3 1 IO burst 만료

1

1 프로세스 completed

---------------------------------------------------------

<8>

3

---------------------------------------------------------

<9>

3

3 프로세스 completed

---------------------------------------------------------

<10>

2

2 프로세스 completed

---------------------------------------------------------

<11>

실행 종료

PID TT WT

1 7 0

2 8 0

3 5 0

AVG.TT : 6.7

AVG.WT : 0.0

7. 느낀점

Input과 output을 직접 설계하고, 일부분에 한해 직접 필요한 가정을 세우면서 구현하다 보니 프로세스 관리 및 스케줄링에 대해 높은 이해도를 필요로 하는 과제였다고 생각합니다. 과제 설명을 제대로 읽지 않아 c++로 작성했던 코드를 c로 수정하며 디버깅하는 과정은 쉽지 않았으나, 많은 OS에서 널리 사용되는 MFQ를 직접 구현해 볼수 있다는 점에서 뜻깊고 재미있었습니다.