

운영체제_[2022-2학기] Project-01

[MFQ Scheduling Simulation]

2022.09.24

엄영익

목차

- 프로젝트 소개
 - 목표
 - 세부 내용
 - 구현
 - 제출물 및 제출기한

Project-01

- 목표
 - MFQ 스케줄링 기법 구현
(개인별 프로젝트)
- 세부 목표
 - 3개의 RQ를 갖는 MFQ 스케줄링 기법 구현
 - Q0: Time quantum 2인 RR 스케줄링 기법 사용
 - Q1: Time quantum 4인 RR 스케줄링 기법 사용
 - Q2: SRTN 스케줄링 기법 사용

Project-01

- 세부 내용

- 각 프로세스는 최초에 각자에게 지정된 queue로 진입하며, 최초 진입 queue는 입력에 의해 결정됨
- Q_i 에서 스케줄 받아 실행하고 해당 queue의 time quantum을 소모할 경우 Q_{i+1} 로 진입 ($i = 0, 1$)
- Q_i 에서 스케줄 받아 실행한 프로세스가 IO burst를 마치고 wakeup 되는 경우 Q_{i-1} 로 진입 ($i = 1, 2$)
- Q_2 의 SRTN 스케줄링은 Q_2 에 있는 프로세스들만을 대상으로 함
- 우선순위는 $Q_0 > Q_1 > Q_2$ 의 순이며, 스케줄링은 항상 높은 우선순위의 queue에서부터 이루어짐

Project-01

- 구현

- 입력 (입력 파일명 input.txt로 작성)

- 프로세스 개수 및

- 각 프로세스 별 도착 시간(AT), 최초 진입 queue, # Cycles, 수행 트레이스(CPU-BT, IO-BT, ...) 등

- (CPU-BT 및 IO-BT는 각각 CPU burst time, IO burst time을 의미함)

- (#Cycles는 수행 트레이스의 (CPU-BT, IO-BT) 쌍의 개수임; 다만, 마지막 Cycle에는 CPU-BT만 존재함)

- 출력

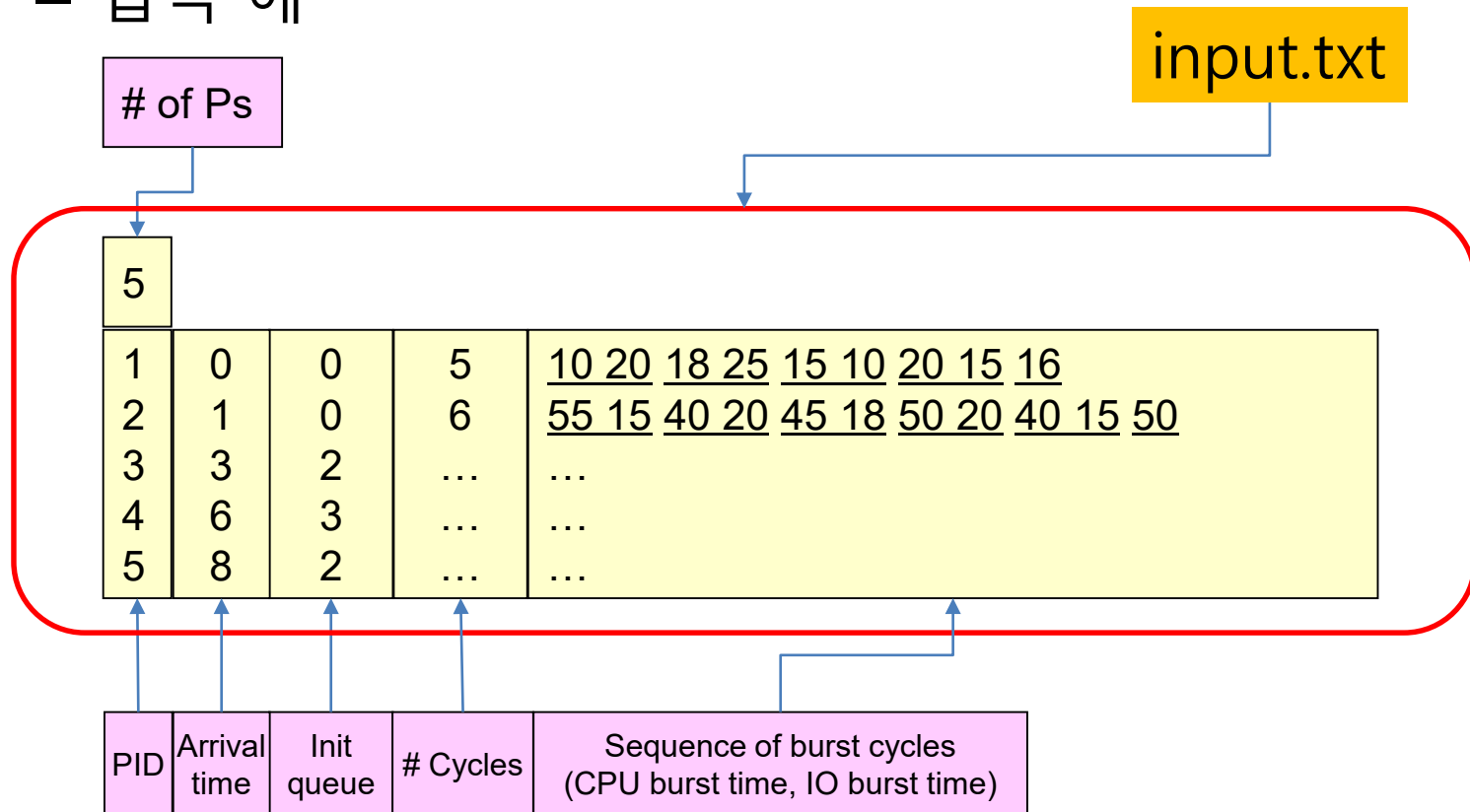
- 스케줄링 결과 (Gantt chart)

- 각 프로세스 별 Turnaround Time(TT) 및 Waiting Time(WT)

- 전체 프로세스의 평균 TT 및 평균 WT

Project-01

- 구현
 - 입력 예



Project-01

- 구현

- 고려 사항

- 모든 프로세스들의 IO burst는 병렬 진행 가능한 것으로 가정
 - 기타 고려사항이나 가정이 필요할 경우, 스스로 판단하여 rule을 정하고, 이를 결과 보고서에 명시해야 함
 - 출력 형태는 각자 자유롭게 결정;
단 출력 결과를 누구나 쉽게 파악할 수 있도록 해야 함
 - 다양한 입력을 사용하여 testing 하고, 그 결과를 보여야 함

- 개발 플랫폼

- Linux 시스템에서 C-언어 사용
 - 불가피한 경우 보고서에 이를 기재하면서 C++, Python 사용 가능

Project-01

- 제출물 및 제출기한
 - 제출물
 - 결과 보고서 파일
 - 설계/구현에 사용한 도구에 대한 설명
 - 설계/구현 내용 설명
 - 다양한 입력에 대한 실행 결과
 - 실행 결과를 보는 방법에 대한 설명
 - 최종 소스 코드 파일
 - 제출기한
 - 2022.10.07(금) 오후 5:00 까지 (Hard Deadline)
 - 제출물들을 '*OS41_2022_2_학번_이름_P01.zip*' 으로 압축하여 iCampus 과제 영역에 온라인 제출

Project-01

- 주의사항
 - Cheating이 발견될 경우 Project 점수로 0점을 받을 뿐만 아니라 본 교과목 성적을 받지 못하고 Fail 하게 됨
 - 공개 소스를 포함하여 외부 코드를 이용하거나 이를 어떠한 형태로든 활용하는 경우 모두 Cheating에 해당함

Q&A