

과제 #3 : Page Replacement Simulator

○ 과제 목표

페이지 교체 알고리즘 동작 방식 이해

○ 과제 내용

1. 다음 일곱 가지 페이지 교체 알고리즘을 C 프로그래밍 언어(개발환경 Linux/BSD, 개발컴파일러 GCC(맥에서는 xcode 등에서 gcc 별도 설치)로 프로그래밍하고 시스템에서 자동 입력과 사용자 (파일) 입력을 사용하여 각 페이지 교체 알고리즘에 대한 페이지 폴트 횟수를 측정하고, Optimal 알고리즘과 비교.

(1) Optimal (2) FIFO (3) LIFO (4) LRU (Least Recently Used) (5) LFU (Least Frequently Used) (6) SC (Second Chance / One handed Clock) 참조(Reference, Read) Bit 하나 사용 (7) ESC (Enhanced Second Chance / Two handed clock / NUR(Not Recently Used)) 참조(Reference) Bit와 Dirty(Write) Bit 두 개 사용.

2. 각 알고리즘에 대한 기본적인 내용은 인터넷을 찾아보고, 해당 알고리즘을 학생 스스로 확인하여 구현. 단, SC와 ESC는 Circular Queue(modular 연산)로 구현

3. 페이지 교체 알고리즘들이 책마다 일부 다르게 설명될 수 있으나, 일반적으로 통용되는 알고리즘을 기준으로 채점할 것임. 단, 구현한 알고리즘의 기준이 다소 다를 경우 해당 알고리즘이 설명된 문헌(인터넷, 책 등 통용 가능한 것)을 제시할 경우 허용할 것임. 어떤 알고리즘을 써야 하는지에 대한 질문에 대한 답은 하지 않을 예정.

○ 제약 조건

(1) 각 페이지 교체 알고리즘의 참조 페이지 스트링은 1~30까지 있다고 가정

(2) 참조 페이지 스트링을 입력 받는 방식은 (ㄱ) 랜덤 함수(rand() 등 사용)를 사용하여 매 프로그램이 수행될 때 마다 프로그램 내부에서 자동으로 생성하는 방법과 (ㄴ) 파일로 입력 받는 방법 두 가지 모두 구현.

(3) (2)의 (ㄱ) 방법과 (ㄴ) 방법 모두 최소 500개의 페이지 스트링이 원칙으로 함. 단, (ㄴ) 방법은 직접 수작업으로 만들어도 되나 랜덤함수를 이용해 만든 스트링 패턴을 파일로 저장하고 이를 오픈하여 사용해도 됨

(4) 페이지 프레임의 개수는 사용자 입력으로 최소 3에서 최대 10개의 프레임을 입력으로 받을 수 있어야 함

(5) 위 모든 페이지 교체 알고리즘은 사용자가 메뉴에서 선택해서 수행하게 해야 하며, 마지막 선택 메뉴는 일곱 개 모두 수행시키는 All이 되어야 함. 사용자 메뉴에서 알고리즘 선택은 최소 1개에서 최대 3개까지 허용해야 함.

(6) 시뮬레이션 결과가 표준 출력이 되고, 동시에 파일로도 저장하는 기능이 제공되어야 함. 파일로 저장하는 방법과 파일 이름은 자율적으로 처리 (예. 표준 출력과 함께 바로 자동으로 파일로 저장되어도 되고, 시뮬레이션 결과 표준 출력 후 특정 사용자 명령을 입력 받아 처리해도 됨)

(7) ESC의 경우 페이지 스트링에 R W 비트가 함께 표시 되어야 함. R W 비트도 제약조건 (2)의 형태로 구현

(8) 입력 에러 처리 등은 기본적인 에러 처리와 관련 usage는 자율적으로 반드시 처리해야 함. 예1) 아래 UI 메뉴, B에서 15를 입력한 경우, 사용자 입력 에러임을 알리고, 3~10범위를 Usage로 보여줘야 함.

(9) 프로그램 종료는 시뮬레이션 후 종료되어도 되고, 특정 키나 특정 명령어 입력으로 종료되어도 됨.

UI 메뉴 (메뉴는 학생들이 자율적으로 만들 수 있으나, 반드시 아래와 같은 기능이 포함되어야 함, 모든 것을 한꺼번에 명령어의 옵션으로 처리해도 됨)

A. Page Replacement 알고리즘 시뮬레이터를 선택하시오 (최대 3개)

(1) Optimal (2) FIFO (3) LIFO (4) LRU (5) LFU (6) SC (7) ESC (8) All

// All을 제외하고 최대 3개의 알고리즘을 선택할 수 있어야 하며, 스페이스로 구분하여 알고리즘 선택이 가능하게 해야 함

B. 페이지 프레임의 개수를 입력하시오.(3~10)

C. 데이터를 입력 방식을 선택하시오. (1,2)

(1) 랜덤하게 생성 // (1)을 선택하면 랜덤하게 페이지 스트링이 생성되고 선택한 알고리즘이 수행되기 전에 페이지 스트링이 표준출력되어야 함. ESC 알고리즘을 구현하기 위해 스트링 옆에 R인지 W(D)인지 표시되어야 함.

(2) 사용자 생성 파일 오픈 // (2)를 선택하면 생성된 파일 입력을 받는 모듈이 있어야 함.

D. 종료

결과 출력

모든 알고리즘들이 수행될 때 페이지 프레임 개수별로 실제 페이지가 교체된 내용과 페이지 폴트가 표시되어야 함. ESC 경우는 페이지별로 R, W가 표시되어야 함. 표시 방법은 학생들이 자율적으로 처리.

○ 과제 제출 마감

- 2022년 11월 21일 (월) 23시 59분 59초까지 구글 클래스룸으로 제출

- 제출물

✓ 소스코드 (makefile도 반드시 제출)

✓ 보고서

- 구현한 페이지 교체 알고리즘 동작 방식 설명
 - 구현한 페이지 교체 알고리즘의 Pseudo code
 - 구현한 스케줄링 알고리즘의 성능 분석 비교(자유롭게, 그래프 등)
 - 주석(라인별 수준)이 포함된 소스코드
 - 사용자 메뉴얼
- 1일 지연 제출마다 30% 감점. 4일 지연 제출 시 0점 처리 (이하 모든 설계 과제 동일하게 적용)

○ 필수 구현

- 제약조건 (1)(2)(3)(4)(5)(6)(8)(9) 기반 페이지 교체 알고리즘 최소 4개

○ 배점 기준

- (1)~(5) 알고리즘 당 : 8점, (6) 알고리즘 : 15점, (7) 알고리즘 : 30점, 보고서 15점
- 모든 알고리즘은 위 9가지 제약조건 (1)(2)(3)(4)(5)(6)(8)(9)에 따라 구현하지 않으면 제약조건 하나 미구현당 해당 점수에서 20% 감점. 제약조건 (7)은 ESC에만 포함되기 때문에 ESC를 구현할 경우 반드시 포함되어야 함