

2023 Fall / System Programming / Lab-1

주제: 가상 메모리 관리-페이지 교체 정책, 시그널 처리

1. 목표

- * 가상 메모리 주소와 물리 메모리 체계를 이해한다.
- * 시그널과 프로세스의 시그널 처리 방식을 이해한다.
- * 위의 개념을 적용한 프로그램 코드를 완성한다.
 - 주어진 프로그램은 input.txt에서 프로그램 주소를 하나씩 읽은 후, 해당 주소가 물리 메모리에 존재하는지 여부를 판별하며 이를 위해 페이지 테이블을 운용한다.
 - 만약 물리 메모리에 더 이상 프로그램 페이지를 넣을 수 없는 경우, 페이지 교체 정책이 교체(victim) 페이지를 지정한 후 해당 페이지가 저장된 물리 프레임에 필요한 페이지를 배치한다.

2. 과제

- * Part#1 - Background: 아래의 내용을 조사하여 레포트 작성
 - 가상주소와 물리주소의 차이
 - 가상 주소 체계의 장점
 - Page Table
 - Page Table Entry
 - Page Fault
 - Page Fault Handler & Replacement policy
 - Signal
 - Signal handler
- * Part#2 - Implementation: 주어진 소스코드(.c)를 완성
 - 총 4개의 “TODO” 항목을 구현
 - 구현 시, 추가적인 전역변수 선언 및 사용 금지
 - 모든 수정 및 추가한 소스코드에 대해 상세한 주석을 단 후, 보고서에 설명
 - 이미 구현된(완성된) 코드는 수정 금지
- * Part#3 - Evaluation: 구현한 프로그램에 대한 분석 및 이해
 - 구현한 항목뿐만 아니라, 전체 프로그램 코드의 동작에 대해 설명
 - Page offset 크기를 다르게 입력하여 결과물을 비교.
 - : e.g., page offset 크기를 11, 12, 13, 14 등으로 바꾸었을 시 생기는 변화에 대해 관찰하고 이 원인을 분석함.
 - Page replacement algorithm을 다르게 하여 결과물을 비교.
 - : 순차 교체 방식(d)과 랜덤 방식(r), 기타 교체 방식(a, b, c, ...)을 사용하였을

때 생기는 변화에 대해 관찰하고, 이에 대한 분석을 수행.

- 과제를 수행하면서 어려웠던 점, 이를 어떻게 해결하였는지(또는 해결하지 못한 이유)를 작성하여 보고서의 마지막에 기술.

3. 제출

- * 제출 내용 = 1) 구현한 소스코드.c + 2)보고서
- * 제출 기한: 12월 17일 (일요일) 23시 59분. 지각제출 불허
- * 보고서는 필요한 모든 내용을 하나의 문서 파일 안에 작성 후 제출

4. 평가

- * Part#1(30%) + Part#2(30%) + Part#3(40%)

5. 프로그램 수행 결과 화면 예시

- * 프로그램 시작: 입력값 page offset 13 bits, replacement algorithm: d

```
hello@ubuntu:~/SP$ ./a.out 13 d
입력된 페이지 별 크기 :8192Bytes
프로세스의 페이지 개수 :16384개
페이지 교체 알고리즘 :d
페이지 폴트 발생 : 페이지 3583를 프레임 0로 로드
페이지 폴트 발생 : 페이지 12697를 프레임 1로 로드
페이지 폴트 발생 : 페이지 2634를 프레임 2로 로드
페이지 폴트 발생 : 페이지 3952를 프레임 3로 로드
페이지 폴트 발생 : 페이지 2568를 프레임 4로 로드
페이지 폴트 발생 : 페이지 16211를 프레임 5로 로드
페이지 폴트 발생 : 페이지 0를 프레임 6로 로드
페이지 폴트 발생 : 페이지 3148를 프레임 7로 로드
페이지 폴트 발생 : 페이지 3386를 프레임 8로 로드
페이지 폴트 발생 : 페이지 3850를 프레임 9로 로드
페이지 폴트 발생 : 페이지 12487를 프레임 10로 로드
페이지 폴트 발생 : 페이지 15100를 프레임 11로 로드
```

- * 프로그램 수행 도중 인터럽트 발생

```
페이지 폴트 발생 : 페이지 2065를 프레임 32로 로드
페이지 폴트 발생 : 페이지 2047를 프레임 24로 로드
^C
(Interrupt) 현재까지 발생한 페이지 폴트의 수 : 120
페이지 폴트 발생 : 페이지 1626를 프레임 32로 로드
페이지 폴트 발생 : 페이지 2058를 프레임 30로 로드
페이지 폴트 발생 : 페이지 2055를 프레임 3로 로드
페이지 폴트 발생 : 페이지 2065를 프레임 52로 로드
페이지 폴트 발생 : 페이지 1763를 프레임 16로 로드
페이지 폴트 발생 : 페이지 1633를 프레임 58로 로드
페이지 폴트 발생 : 페이지 3068를 프레임 54로 로드
페이지 폴트 발생 : 페이지 2000를 프레임 45로 로드
페이지 폴트 발생 : 페이지 5679를 프레임 53로 로드
^C
(Interrupt) 현재까지 발생한 페이지 폴트의 수 : 129
페이지 폴트 발생 : 페이지 1642를 프레임 34로 로드
페이지 폴트 발생 : 페이지 1910를 프레임 50로 로드
페이지 폴트 발생 : 페이지 3729를 프레임 54로 로드
페이지 폴트 발생 : 페이지 1462를 프레임 55로 로드
페이지 폴트 발생 : 페이지 3068를 프레임 61로 로드
페이지 폴트 발생 : 페이지 1702를 프레임 57로 로드
^C
(Interrupt) 현재까지 발생한 페이지 폴트의 수 : 135
페이지 폴트 발생 : 페이지 6083를 프레임 44로 로드
페이지 폴트 발생 : 페이지 1284를 프레임 54로 로드
```

* 작업 완료 시

```
페이지 폴트 발생 : 페이지 4048를 프레임 32로 로드
페이지 폴트 발생 : 페이지 3134를 프레임 18로 로드
페이지 폴트 발생 : 페이지 3942를 프레임 34로 로드
페이지 폴트 발생 : 페이지 16211를 프레임 57로 로드
페이지 폴트 발생 : 페이지 0를 프레임 19로 로드
페이지 폴트 발생 : 페이지 2977를 프레임 51로 로드
프로세스가 완료되었습니다 . 종료 신호를 기다리는 중 ...
█
```

* 시그널을 발생시켜 프로세스를 종료

```
페이지 폴트 발생 : 페이지 2977를 프레임 51로 로드
프로세스가 완료되었습니다 . 종료 신호를 기다리는 중 ...
```

^C

(Interrupt) 현재까지 발생한 페이지폴트의 수 : 566

202101234/Hong gildong

---물리 프레임 별 접근 횟수 ----

```
[000]frame: 48
[001]frame: 152
[002]frame: 228
[003]frame: 80
[004]frame: 264
[005]frame: 45
[006]frame: 34
[007]frame: 76
[008]frame: 212
```

...

```
[055]frame: 60
[056]frame: 124
[057]frame: 471
[058]frame: 104
[059]frame: 132
[060]frame: 168
[061]frame: 608
[062]frame: 92
[063]frame: 104
```

페이지 폴트 횟수 : 566

hello@ubuntu:~/SP\$ █