

컴퓨터 구조 HOMEWORK 3

233 Decoder

저명한 연구실인 충북대학교 컴퓨터 시스템 연구실에서 새로운 CPU에 탑재될 디코더를 제작했다.

하지만 랩장의 실수로 디코더의 소스코드를 삭제했다.

다행이 미리 만들어 두었던 가이드라인 문서를 찾을 수 있었다.

가이드라인을 보고 233 디코더의 소스코드를 복원하라!

1. 233 Decoder는 memory.txt 파일에 적힌 기계어를 해독하는 기능을 가지고 있다.
2. 기계어는 2개의 명령어 비트와 3개의 주소 비트, 3개의 값 비트로 구성되어있다.
3. 기계어는 0과 1로만 구성되어있다.
4. 233 decoder는 명령어를 해독하여 크기 8의 정수형 배열을 조작한다.
5. 초기 배열은 모두 0으로 초기화 되어있다.
6. 명령어의 구성은 다음과 같다.
 - 6.1 00(store) 주소비트의 10진수 인덱스에 값 비트의 10진수를 할당한다.
 - 6.2 01(add) 주소비트의 10진수 인덱스에 값 비트의 10진수를 더한다.
 - 6.3 10(sub) 주소비트의 10진수 인덱스에 값 비트 10진수를 뺀다.
 - 6.4 11(swap) 주소비트의 10진수로 인덱스와 값 비트 10진수 인덱스의 값들을 교환한다.
 - 6.5 기계어의 모든 비트가 1인 경우 배열의 모든 값을 0으로 초기화 한다.
7. 구현한 233decoder에 첨부된 memory.txt 파일을 읽은 결과 화면을 캡처하여 보고서를 작성한다.
8. 제출 파일 - 233decoder의 소스코드, 개발 과정 보고서 pdf 파일
9. 제출 형태 - 소스코드 c 파일과 개발 과정 보고서 pdf 파일을 하나의 압축파일로 제출
10. 제출 파일명 - 학번.이름(영문).pdf ex) 2025000000.chaeunwoo.zip

명령어 가이드라인 필수 작성 양식 및 채점 비율

1. 각 명령어 비트 별 기능 설명 - 10점
 2. 코드 라인 별 주석 - 30점
 3. 예외 처리 - 10점
 4. 기계어 검증 - 50점
-

출력 예시

```
cslab@cslab192:~/hayong/HW$ ./233decoder
+-----+-----+-----+-----+-----+-----+-----+-----+
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Value | 0 | 0 | 0 | 3 | -3 | 0 | 0 | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

출력 결과가 위와 동일해야 정답 인정

과제에 대한 문의는 컴퓨터 구조 오픈채팅과 TA 개인 메시지를 통해 문의 바랍니다.