실습 11주차 과제 12161633 이영주

1) 프로그램 설명

이전 과제와 중복되는 내용의 설명은 생략하고 추가된 내용의 설명만 덧붙이겠다. checkArgv()에서 입력 파라미터가 1개라면 “Successfully loaded”라는 문장을 출력해준다. readAsmLine(line)에서는 comment line의 시작을 뜻하는 ‘.’값을 처리해준다. ‘.’이 입력된다면 그 줄은 모두 없는 것처럼 처리해준다. 입력된 문장들을 저장할 intermediate\_file을 초기화한다. Symboltable을 의미하는 SYMTAB은 처음에 아무것도 저장되어 있지 않으므로 초기화 시켜주고, operation table을 의미하는 OPTAB은 명령어들을 저장해준다. 시작을 의미하는 ‘START’를 읽었다면 0으로 초기화한 LOCCTR을 시작주소로 바꿔준다. 그 후, intermediate\_file에 주소는 없고, line\_label, line\_opcode, line\_operand를 넣어준다. 만약 line\_opcode가 END가 아니고, line\_label이 comment line의 시작을 의미하는 ‘.’이 아니고, line\_label이 비어있지 않는 경우, 그 label이 SYMTAB에 있으면 이미 찾은 label이 존재하는 것이기 때문에 “ERROR : duplicated symbol”을 출력해준다. 만약 SYMTAB에 없다면 처음 발견하는 것이므로 SYMTAB[line\_label]값을 LOCCTR로 바꿔준다. 이 과정이 끝나고 intermediate\_file에 (LOCCTR, line\_label, line\_opcode, line\_operand)를 저장한다. 만약 line\_opcode가 OPTAB에 저장되어 있다면 LOCCTR에 3을 더해준다. PC는 3씩 증가하기때문에 3을 더해주는 것이다. 아니면 line\_opcode에 ‘WORD’가 저장되어 있다면 WORD는 3 BYTE 단위이므로 3을 LOCCTR에 더해준다. ‘RESW’가 저장되었다면 WORD크기인 3에 line\_operand를 곱하여 LOCCTR에 더한다. ‘BYTE’라면 상수를 표현하는데 필요한 만큼의 바이트를 지정하는 것이다. 이때는 “implement later”을 출력해준다. line\_opcode가 ‘.’이라면 아무것도 출력하지않는다. 만약 위의 경우에 모두 해당하지 않는다면 OPTAB을 잘못 지정했거나 문법상 오류가 있는 것이므로 “invalid operation code”를 출력하고 전체적인 코드에 이상이 생긴 것이므로 종료한다. 모든 문장을 탐색하였다면 마지막에 저장된 LOCCTR에서 start\_addr을 빼면 프로그램의 길이가 나오게 된다. 이 값을 program\_length에 저장한다. 그 후, “program length is ‘program\_length’”를 출력하고, SYMTAB에 있는 값들을 모두 출력한다. Intermediate\_file에 저장된 라인들 또한 모두 출력해준다.

2) 출력 결과

