과목명 : 시스템 프로그래밍

담당 교수명 : 김지환

<<Assignment 3>>

서강대학교 컴퓨터공학과

학번: 20171690

이름: 정유석

**목차**

1. 프로그램 개요
2. 프로그램 설명
   1. 프로그램 흐름도
3. 분할 c 파일
   1. linkingLoader.c
      1. 설명
      2. 모듈 정의
         1. 모듈 이름 : progaddr()
            1. 기능
            2. 사용 변수
         2. 모듈 이름 : loader()
            1. 기능
            2. 사용 변수
         3. 모듈 이름 : loadMap()
            1. 기능
            2. 사용 변수
         4. 모듈 이름 : HRecord()
            1. 기능
            2. 사용 변수
         5. 모듈 이름 : DRecord()
            1. 기능
            2. 사용 변수
         6. 모듈 이름 : RRecord()
            1. 기능
            2. 사용 변수
         7. 모듈 이름 : TRecord()
            1. 기능
            2. 사용 변수
         8. 모듈 이름 : MRecord()
            1. 기능
            2. 사용 변수
         9. 모듈 이름 : newTRaddr()
            1. 기능
            2. 사용 변수
         10. 모듈 이름 : freeTRHead()
             1. 기능
             2. 사용 변수
         11. 모듈 이름 : setMem()
             1. 기능
             2. 사용 변수
         12. 모듈 이름 : addES()
             1. 기능
             2. 사용 변수
         13. 모듈 이름 : addRnN()
             1. 기능
             2. 사용 변수
         14. 모듈 이름 : searchESTAB()
             1. 기능
             2. 사용 변수
         15. 모듈 이름 : searchRN()
             1. 기능
             2. 사용 변수
         16. 모듈 이름 : haltLinkingLoader()
             1. 기능
             2. 사용 변수
         17. 모듈 이름 : isObjFile()
             1. 기능
             2. 사용 변수
         18. 모듈 이름 : printES()
             1. 기능
             2. 사용 변수
         19. 모듈 이름 : freeRN()
             1. 기능
             2. 사용 변수
   2. execution.c
      1. 설명
      2. 모듈 정의
         1. 모듈 이름 : printReg()
            1. 기능
            2. 사용 변수
         2. 모듈 이름 : executeProg()
            1. 기능
            2. 사용 변수
         3. 모듈 이름 : setBP()
            1. 기능
            2. 사용 변수
         4. 모듈 이름 : clearBP()
            1. 기능
            2. 사용 변수
         5. 모듈 이름 : printBP()
            1. 기능
            2. 사용 변수
         6. 모듈 이름 : searchBP()
            1. 기능
            2. 사용 변수
   3. opcodeAction.c
      1. 설명
      2. 모듈 정의
         1. 모듈 이름 : opAct()
            1. 기능
            2. 사용 변수
         2. 모듈 이름 : getRegPtr()
            1. 기능
            2. 사용 변수
         3. 모듈 이름 : write\_to\_memory()
            1. 기능
            2. 사용 변수
4. 구조체 정의
   1. ESNode
   2. EShead
   3. referNode
   4. newTR
   5. BPNode
5. 전역 변수 정의
   1. int PROGADDR
   2. Eshead\* ESTAB
   3. referNode\* referHead
   4. newTR\* TRHead
   5. BPNode\* BPHead
   6. EXEC\_ADDR
   7. EXEC\_LEN
   8. PROG\_START
   9. PROG\_END
   10. continuing
6. 코드

1. 프로그램 개요

2. 프로그램 설명

2.1 프로그램 흐름도

< SIC Shell Program Flow Chart >

3. 분할 c파일

**3.1 linkingLoader.c**

3.1.1 설명

3.1.2 모듈 정의

3.1.2.1 모듈 이름 : **progaddr(char\* addr)**

3.1.2.1.1 기능

3.1.2.1.2 사용 변수

**char\* addr** –

3.1.2.2 모듈 이름 : **loader(char\* param)**

3.1.2.2.1 기능

main

3.1.2.2.2 사용 변수

**char\* param** –

**char\*\* objFile** –

**int objCnt** –

**int i, CS** –

**int flag** –

**FILE\*\* objFP** –

**char line[MAX\_LINE\_LEN]** –

3.1.2.3 모듈 이름 : **loadMap(int objCnt)**

3.1.2.3.1 기능

parameter

3.1.2.3.2 사용 변수

**int objCnt** –

**int len** –

3.1.2.4 모듈 이름 : **HRecord(char\* line, int currentCS, char\* file)**

3.1.2.4.1 기능

3.1.2.4.2 사용 변수

**char\* line** –

**int currentCS** –

**char\* file** –

**int charPtr** –

**char tmp[7]** –

**int csAddr** –

**int i** –

3.1.2.5 모듈 이름 : **DRecord(char\* line, int currentCS, char\* file)**

3.1.2.5.1 기능

3.1.2.5.2 사용 변수

**char\* line** –

**int currentCS** –

**char\* file** -

**int charPtr** –

**char name[7]** –

**char addr[7]** –

3.1.2.6 모듈 이름 : **RRecord(char\* line, int currentCS, char\* file)**

3.1.2.6.1 기능

3.2.2.6.2 사용 변수

**char\* line** –

**int currentCS** –

**char\* file** -

**int charPtr** –

**int refAddr** –

**char referenceNum[7]** –

**char referenceName[7]** –

3.1.2.7 모듈 이름 : **TRecord(char\* line, int currentCS, char\* file)**

3.1.2.7.1 기능

3.1.2.7.2 사용 변수

**char\* line** –

**int currentCS** –

**char\* file** –

**int currentAddr** –

**int charPtr** –

**char strAddr[7]** –

**char tmp[7]** –

**int tLen** –

**int memVal** –

**int setPtr** –

3.1.2.8 모듈 이름 : **MRecord(char\* line, int currentCS, char\* file)**

3.1.2.8.1 기능

3.1.2.8.2 사용 변수

**char\* line** –

**int currentCS** –

**char\* file** -

**int charPtr** –

**char \_mAddr[7]** –

**char tmp[7]** –

**int mAddr** –

**int mLen** –

**int mVal** –

**int prevVal** –

3.1.2.9 모듈 이름 : **newTRaddr(int addr)**

3.1.2.9.1 기능

3.1.2.9.2 사용 변수

**int addr** –

**newTR\* pNew** –

**newTR\* pMove** –

3.1.2.10 모듈 이름 : **freeTRHead()**

3.1.2.10.1 기능

3.1.2.10.2 사용 변수

**newTR\* pFree** –

3.1.2.11 모듈 이름 : **setMem(int addr, int val, char\* file)**

3.1.2.11.1 기능

shell

3.1.2.11.2 사용 변수

**int addr** –

**int val** –

**char\* file** –

3.1.2.12 모듈 이름 : **addES(EShead\* ES, char\* name, char\* loc)**

3.2.2.12.1 기능

3.1.2.12.2 사용 변수

**Eshead\* ES** –

**char\* name** –

**char\* loc** –

**ESnode\* pNew** –

**ESnode\* ptmp** –

3.1.2.13 모듈 이름 : **addRN(char\* ref, int addr)**

3.1.2.13.1 기능

3.1.2.13.2 사용 변수

**char\* ref** –

**int addr** –

**referNode\* pNew** –

**int refNum** –

3.1.2.14 모듈 이름 : **searchESTAB(char\* name, int objCnt)**

3.1.2.14.1 기능

3.1.2.14.2 사용 변수

**char\* name** –

**int objCnt** –

**ESnode\* ptmp** –

3.1.2.15 모듈 이름 : **searchRN(char\* ref)**

3.1.2.15.1 기능

3.1.2.15.2 사용 변수

**char\* ref** –

**referNode\* ptmp** –

**int refNum** –

3.1.2.16 모듈 이름 :  **haltLinkingLoader(char\*\* objFile, FILE\*\* objFP, int objCnt)**

3.1.2.16.1 기능

3.1.2.16.2 사용 변수

**char\*\* objFile** –

**FILE\*\* objFP** –

**int objCnt** –

**ESnode\* pFree** –

3.1.2.17 모듈 이름 :  **isObjFile(char\* file)**

3.1.2.17.1 기능

3.1.2.17.2 사용 변수

**char\* file** –

**int ptr** –

3.1.2.18 모듈 이름 :  **printES(int objCnt)**

3.1.2.18.1 기능

3.1.2.18.2 사용 변수

**int objCnt** –

**int i** –

**ESnode ptmp** –

3.1.2.19 모듈 이름 :  **freeRN()**

3.1.2.19.1 기능

3.1.2.19.2 사용 변수

**referNode\* pFree** –

**3.2 execution.c**

3.2.1 설명

assembler

3.2.2 모듈 정의

3.2.2.1 모듈 이름 : **printReg()**

3.2.2.1.1 기능

메모리

3.2.2.1.2 사용 변수

3.2.2.2 모듈 이름 : **executeProg()**

3.2.2.2.1 기능

메모리의

3.2.2.2.2 사용 변수

**int opcode** –

**unsigned char flags** –

**unsigned char reg** –

**unsigned char ni** –

**int format** –

**int target** –

**newTR\* addr\_of\_new\_TRecord** –

**int cnt** –

**int MAX\_CNT** –

**int endFlag** –

**int byteSise** –

**int i** –

3.2.2.3 모듈 이름 : **setBP(char\* addr)**

3.2.2.3.1 기능

memory의 s

3.2.2.3.2 사용 변수

**char\* addr** –

**int bpAddr**–

3.2.2.4 모듈 이름 : **clearBP()**

3.2.2.4.1 기능

memor

3.2.2.4.2 사용 변수

**BPNode\* pFree** –

3.2.2.5 모듈 이름 : **printBP()**

3.2.2.5.1 기능

parameter

3.2.2.5.2 사용 변수

**BPNode\* ptmp** –

3.2.2.6 모듈 이름 : **searchBP()**

3.2.2.6.1 기능

parameter

3.2.2.6.2 사용 변수

**BPNode\* pMv** -

**3.3 opcodeAction.c**

3.3.1 설명

txt

3.3.2 모듈 설명

3.3.2.1 모듈 이름 : **opAct(int opcode, int format, int target, int flags)**

3.3.2.1.1 기능

명령어

3.3.2.1.2 사용 변수

**int opcode** –

**int format** –

**int target** –

**int flags** –

**unsigned char \_reg1, \_reg2** –

**int\* r1, \*r2** –

**unsigned ar ni, x, b, p, e** –

**int memVal** –

**int LOC** –

3.3.2.2 모듈 이름 : **getRegPtr(unsigned char reg)**

3.3.2.2.1 기능

opcode

3.3.2.2.2 사용 변수

**unsigned char reg** –

3.3.2.3 모듈 이름 : **write\_to\_memory(int LOC, int memVal)**

3.3.2.3.1 기능

주어진

3.3.2.3.2 사용 변수

**int LOC** –

**int memVal** –

**unsigned char** **b[3]** –

**int i** – loop counter로 사용되었다.

4. 구조체 정의

4.1 **ESNode**

사용한 명령어를 저장하는 node로 구조는 위와 같다. 명령어와 그 명령어의 순서를 저장하는 변수인 num과 hist, 그리고 linked list로 구현하기 위한 노드 포인터 link로 구성되어있다.

4.2 **EShead**

opcode.txt 파일에서 읽어온 operation에 관한 정보를 저장하는 node이다. opcode와 format을 저장하는 int형 변수, operation 이름을 저장할 operation, 그리고 linked list로 구현하기 위한 node pointer인 link로 구성되어있다.

4.3 referNode

4.4 newTR

4.5 BPNode

5. 전역 변수 정의

5.1 **int PROGADDR**

수행한

5.2 **EShead\* ESTAB**

opcode

5.3 **referNode\* referHead**

구현할

5.4 **newTR\* TRHead**

dump

5.5 **BPNode\* BPHead**

dump

5.6 **int EXEC\_ADDR**

dump

5.7 **int EXEC\_LEN**

dump

5.8 **int PROG\_START**

dump

5.9 **int PROG\_END**

dump

5.10 **int continuing**

dump

**6. 코드**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\* System Programming \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 20171690.c \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Yuseok \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 190325 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/