

시스템 S/W 실습11



학번 : 2016110056

학과 : 불교학부

이름 : 박승원

날짜 : 2016년 12월 7일

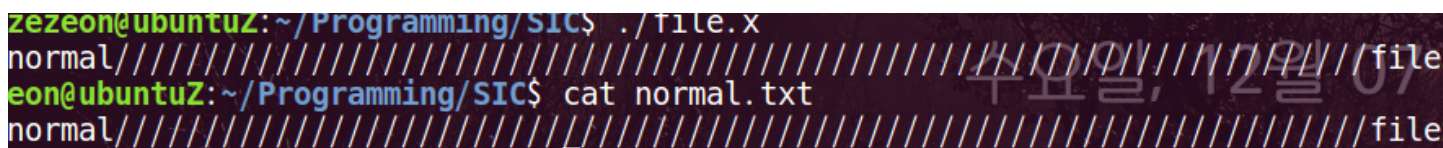


1. 파일 처리

Listing 1: file.c

```
#include <sys/ file .h>
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int i, fd;
    int charsRead;
    char buffer [4096];
    fd = open("normal.txt", O_CREAT | O_RDWR, 0600);
    write(fd, "normal", 6); /* write( int fd, char *buf, int count) */
    for(i=1; i<=70; i++) write(fd, "/0", 1);
    write(fd, " file ",4);
    close (fd);
    fd = open("normal.txt", O_RDONLY);
    charsRead = read(fd, buffer , 100);
    write (1, buffer , charsRead);
    close (fd);
}
```



```
zezeon@ubuntuZ:~/Programming/SIC$ ./file.x
normal////////////////////////////////////file
zeon@ubuntuZ:~/Programming/SIC$ cat normal.txt
normal////////////////////////////////////file
```

2. 다음에 주어진 MacroAssembly program MacroSrcfile.txt file을 읽어서 각 줄을 LABEL, OPCODE, OPERAND로 분리하여 Intfile에 출력하면서 Macro Definition Table MDT[]와 macro Name Table MNT[]를 생성하고, MDT[]와 MNT[]의 내용을 출력하는 Macro Processor PASS1에 해당하는 프로그램 MacroPass1를 구현하고 실습하시오. 단, C Program compile명령은 다음과 같고, 입력파일 MacroSrcfile.txt file은 e-class에서 download 할 수 있다.

```
$gcc -o MacroPass1 MacroPass1.c
```

```
$/MacroPass1 MacroSrcfile.txt
```

Listing 2: MacroPass1.c

```
#include<iostream>
#include<iomanip>
#include<fstream>
#include<sstream>
```

```

#include<vector>
#include<map>
using namespace std;

vector<array<string, 3>> instructions;
struct Macro
{
    std::string name;
    std::vector<std::string> args;
    int start;
    int end;
};

std::map<std::string, unsigned char> op_table = {
    {"lda", 0x00}, {"ldx", 0x04}, {"sta", 0x0c}, {"stx", 0x10},
    {"add", 0x18}, {"sub", 0x1c}, {"mul", 0x20}, {"div", 0x24},
    {"comp", 0x28}, {"jeq", 0x30}, {"jgt", 0x34}, {"jlt", 0x38},
    {"jsub", 0x48}, {"rsub", 0x4c},
    {"rd", 0xd8}, {"wd", 0xdc}, {"td", 0xe0},
    {"ldch", 0x50}, {"stch", 0x54}, {"addx", 0x19}
};

bool is_opcode(string s)
{
    return op_table.find(s) != op_table.end();
}

int main(int argc, char** v)
{
    ifstream f(v[1]);
    ofstream of("Int");

    // 라벨과 코드 오퍼랜드 구분하는 부분
    char c;
    string com[30];
    int n = 0;
    bool sp_flag = false;
    while(f >> noskipws >> c) {
        if(c != '\n') {
            if(c == '\t' || c == ' ') {

```

```

        if (! sp_flag ) n++;
        sp_flag = true ;
    } else {
        com[n] += c;
        sp_flag = false ;
    }
} else {
    instructions .push_back({com[0], com[1], com[2]});
    n = 0;
    for(int i=0; i<3; i++) com[i]. clear () ;
    sp_flag = false ;
}
}
for(auto& a : instructions ) {
    of << a[0] << ' ' << a[1] << ' ' << a[2] << endl;
}

```

Macro m;

vector<Macro> macros;

```

for(int i=0; i<instructions . size () ; i++) {
    if( instructions [i][1] == "macro") {
        m.name = instructions [i ][0];
        string s;
        stringstream ss{ instructions [i ][2]};
        while(getline (ss, s, ' , ')) m.args.push_back(s); // extract arguments
        m.start = i ;
    } else if( instructions [i ][1] == "mend\r") {
        m.end = i ;
        macros.push_back(m);
    }
}
for(auto& a : macros) {
    cout << "macro name : " << a.name << endl;
    for(int i=a.start ; i<a.end; i++) {
        for(auto& b : instructions [i ]) cout << b << ' ';
        cout << endl;
    }
}
}

```

}

```

zezeon@ubuntu21:~/Programming/SIC$ ./MacroPass1 MacroSrcfile.txt
macro name : plus
lus macro &x,&y,&z
lda &x
add &y
sta &z
macro name : rdbuff
dbuff macro &indey,&bufadr,&reclth
clear x
clear a
clear s
ldt #100
td =x'&indey'
jeq *-3
rd =x'&indey'
compr a,s
jeq *+11
stch &bufadr,x
tixr t
jlt *-19
stx &reclth

```

실행

시 매크로 부분만 따로 출력해준다.

```

zezeon@ubuntuZ:~/Programming/SICS$ cat Int
copy start 0
plus macro &x,&y,&z
    lda &x
    add &y
    sta &z
mend
rdbuff macro &indey,&bufadr,&reclth
    clear x
    clear a
    clear s
    ldt #100
    td =x'&indey'
    jeq *-3
    rd =x'&indey'
    compr a,s
    jeq *+11
    stch &bufadr,x
    tixr t
    jlt *-19
    stx &reclth
mend
first stl retadr
rdbuff fl,buffer,length
cloop lda length
    comp #0
    jeq endfil
    plus a,b,c
    j>cloop
endfil j<retadr
eof byte c'eof'
retadr resw 1
a word 100
b word 200
c resw 1
length resw 1
buffer resb 4096
end first

```

인스트럭션을 구분한 Int파

일의 내용을 출력해보았다.

소감

write(fd, "/0", 1)에서 한글자만 쓰기 때문에 /////로 나오는 것이었다. 항상 교수님께서 퀴즈를 하나씩 내주시는 것 같다. 개행문자가 유닉스와 윈도우즈간에 호환이 안 되어 소스 파일을 처리하는 데에 애를 먹었다.