

# Assignment#2-3

시스템 프로그래밍 실습

제출일: 4월 28일 일요일

분 반: 화요일

담당 교수: 신영주

학 번: 2015722025

학 과: 컴퓨터정보공학부

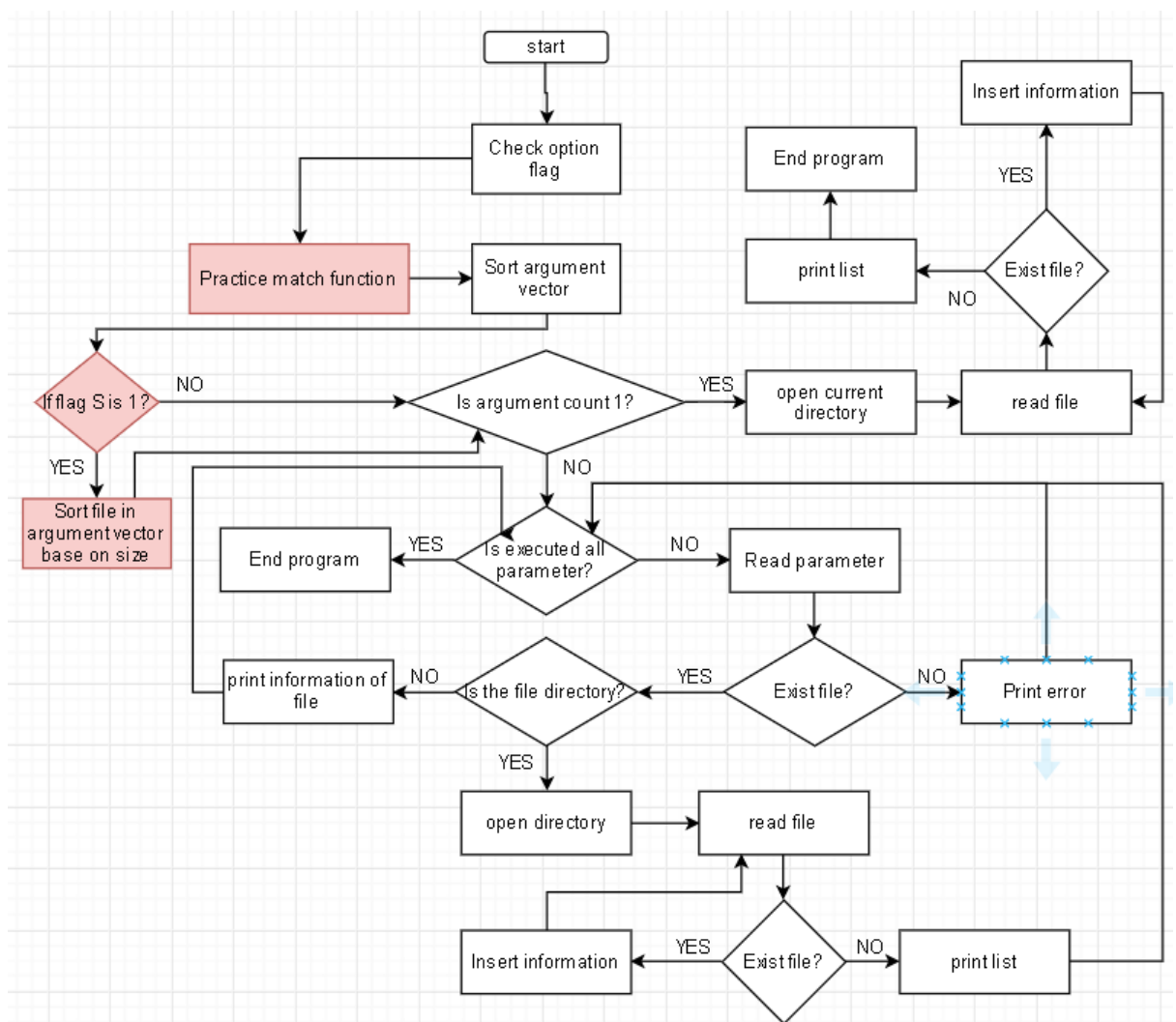
이 름: 정용훈

## 1. Introduction

해당 과제는 2-2과제와 마찬가지로 추가적인 옵션 -h, -S, -r 3가지의 옵션을 추가 구현하며, 새로 배우게 된 함수인 fnmatch를 사용하여 parameter에 \*, ?, [index]가 들어가는 경우 match되는 파일과 디렉토리를 출력하는 프로그램을 구현하는 것이 목적이 된다. 나머지 기능은 전 과제와 동일하게 동작한다.

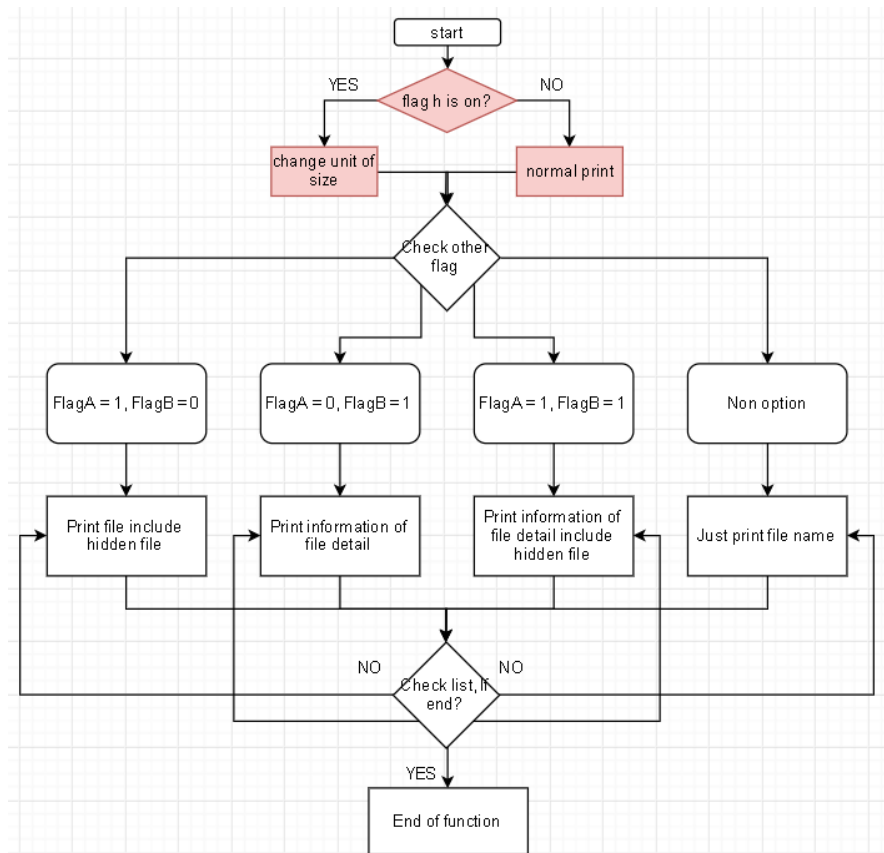
## 2. Flow Char

## Main



메인함수의 전체동작을 flow chart로 나타낸 것이다. 빨간 부분은 전 과제에 비교하였을 때 추가된 부분이며, 자세한 내용은 아래 기재되어있다.

## Printlist



받은 정보를 통하여 option의 flag값을 통해 어떤 정보들을 출력할지 결정한다. 이번 과제에서는 h 옵션이 추가 되어 size를 나타내는 단위가 추가적으로 결정되게 된다. 그렇기에 위 flow chart처럼 signal이 on되었을때는 단위를 바꿔서 출력하는 함수가 실행 될 수 있도록 설정만 해주면 된다.

## Printlist (More about option r)

다음은 옵션 r의 관련된 내용이다. Print 함수의 전체 동작은 위와 같지만 r 옵션이 들어 오면, print 자체를 반대로 해야 하기 때문에 포인터의 이동이 다음과 같이 변경된다. 구현 자체를 이중 링크드리스트로 했기에 방문순서를 변경 하는 코드는 간단하다.

기존 print



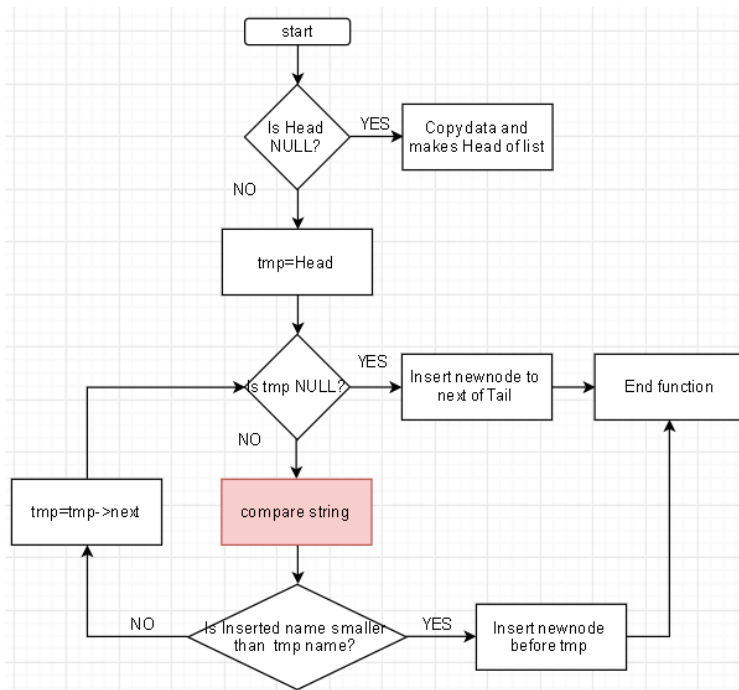
출력 순서 -> Head, 2, 3, 4, Tail

변경된 print



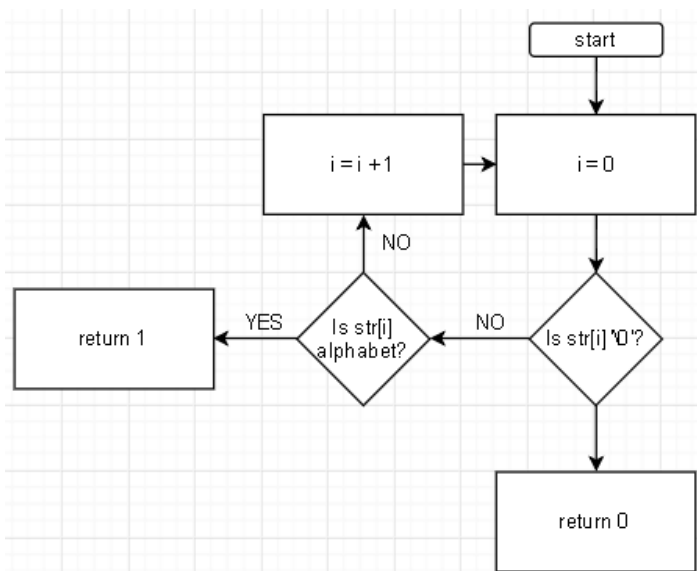
출력 순서 -> Tail, 4, 3, 2, Head

## Insertnode



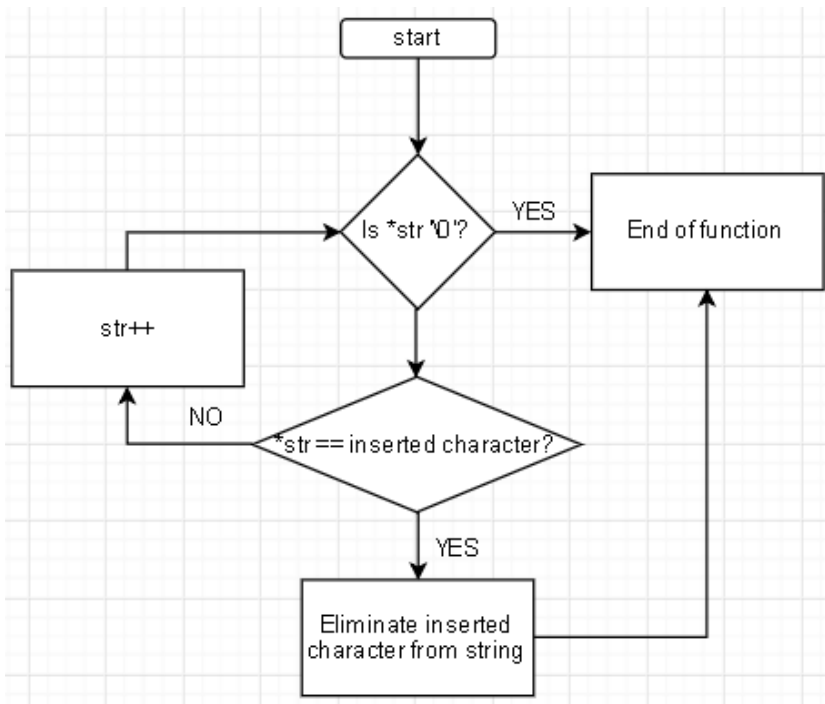
따로 sort함수를 사용하지 않고 insert를 실행 할 때부터 sort가 되며, list가 생성된다. Insert함수는 기존 함수와 동일하게 사용되므로 전체적인 변화는 없지만, S옵션을 사용하면 size를 비교해야 하므로 compare를 하는 부분이 변경되게 된다. 해당 문제는 함수는 같고 조건만 바꿔주어 해결하였다.

## Checkstring



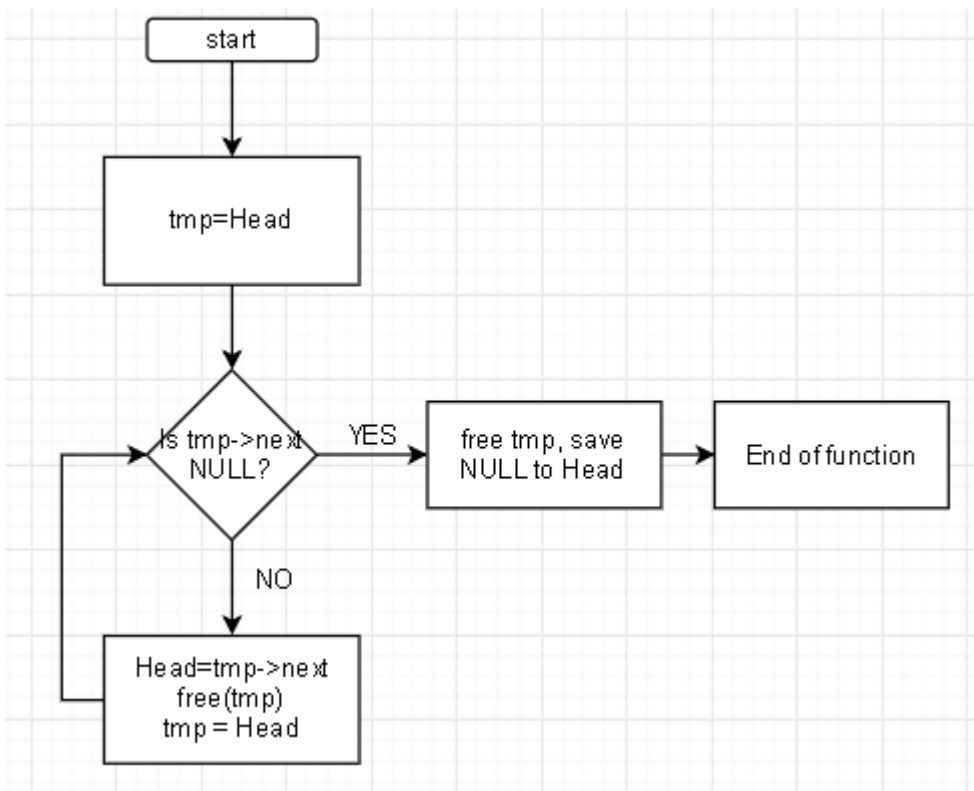
Eliminate와 연계되는 함수로써 알파벳이 아닌 문자로만 이루어진 string일 경우 '.'을 제거하지 않기 위한 함수다. 해당 함수도 변경사항이 없다.

## Eliminate



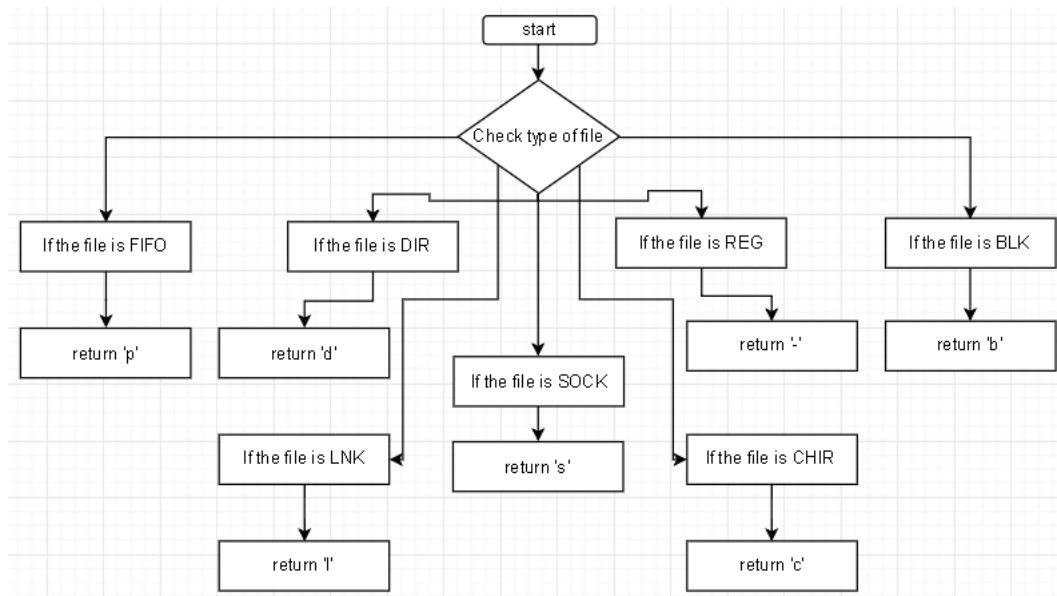
Hidden file의 이름을 비교하기 위해 '.'문자나 '/'문자를 제거하는 함수다.

## Deletelist



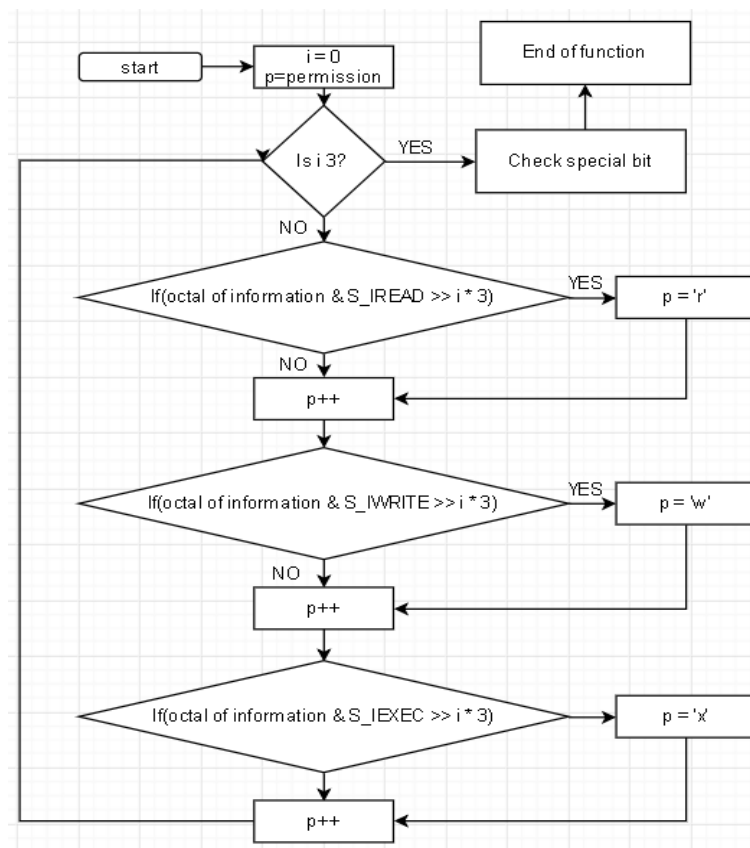
모든 정보 출력 후 linked list를 제거하는 함수다.

## printType



파일의 정보를 받아와 `st_mode`를 통하여 파일의 type을 정의하는 함수다

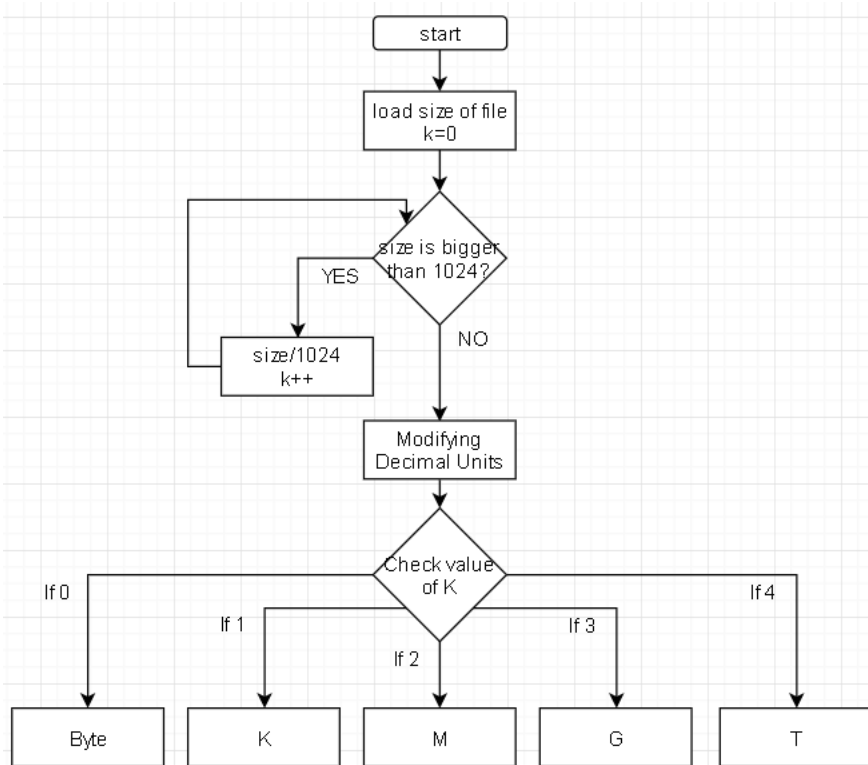
## printPerm



`St_mode`를 받아와 해당 파일의 permission을 확인하여, 최종적인 permission을 출력할 수 있도록 도와주는 함수다.

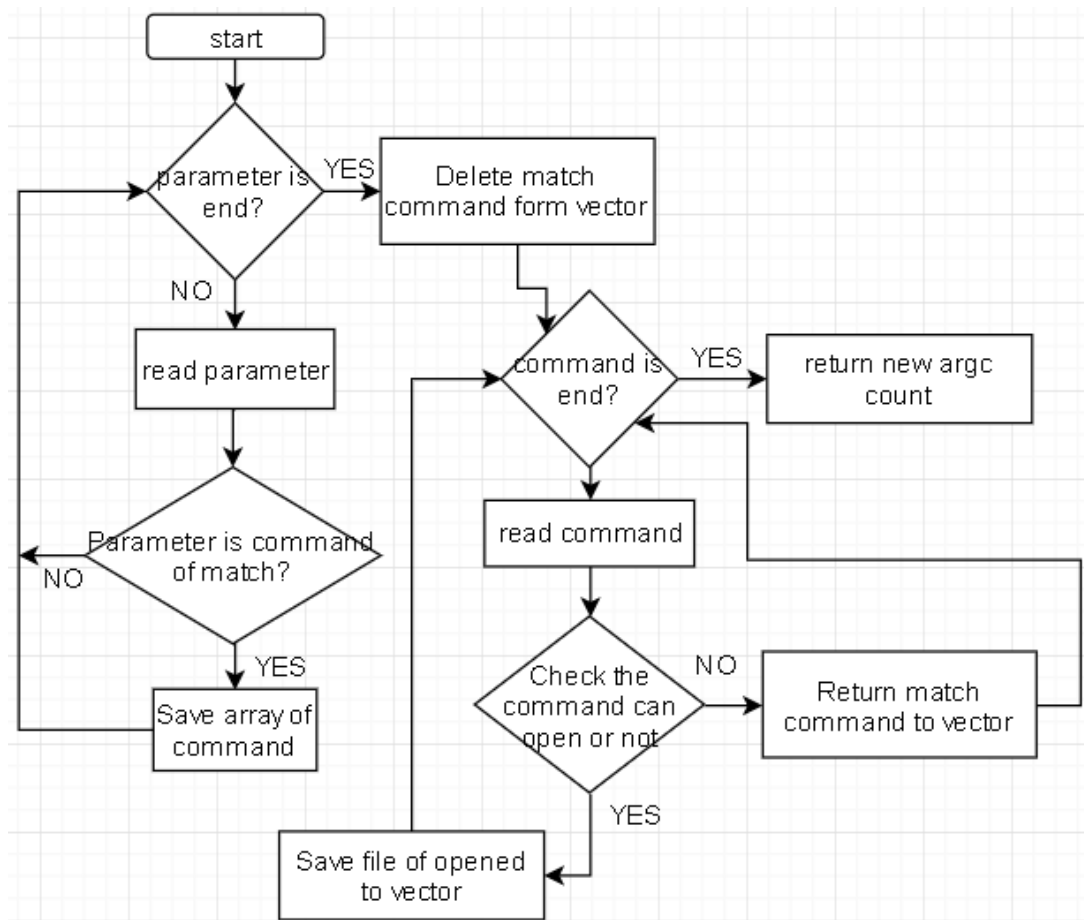
다음 설명하는 함수는 이번 과제를 진행하면서 새로 추가된 함수들이다.

### printOph



h옵션을 구현하기 위한 함수로써 단위를 맞추기 위해 Modifying과정이 필요하며, 그전에는 단순히 1024로 반복하여 나누어주면 된다. 실제 ls와 값이 차이가 나는데 이유는 직접 구현하면 값을 버리게 되고 실제 ls인 경우는 값을 반올림하기에 차이가 난다.

## Matchfunction



다음 함수는 match명령이 인자로 들어오는지 판단하며, 해당 명령을 다른 vector에 옮겨 두었다가 해당 명령과 match되는 파일들을 vector로 옮겨 전 과제에서 구현하였던 함수를 그대로 수행할 수 있도록 도와주는 함수다. 경로 없이 단순히 match명령만 들어오게 된다면, 현재 directory를 열어 match시킨후 match되는 파일들을 열어주면 되고 경로와 함께 들어온다면, 경로와 match명령을 따로 읽어 경로에 해당하는 file들을 모두 출력해주면 된다. 단 조건으로 match명령은 인자의 끝에만 들어와야 하고 그 전 ex) test/\*/\* 이런 식의 명령은 다른 예외로 처리해주어야 한다.



### 3. Pseudo code

#### Main

```
int main(int argc, char** argv)
{
    int c = 0;
    int height = 2;
    DIR *dirp;

    struct dirent *dir;
    struct stat buf;
    struct group *gid;
    struct passwd *uid;

    struct tm *time;
    char buff[256];

    int argcount = 0;
    int total=0;
    char swap[256];
    char Subargv[256][256]={'\0'};
    char match[256];
    int count=1;
    char NULLpath[256]={0};
```

```

////////////////////////////////Set option////////////////////////////////
while (repeat function before checking all option)
{
    switch (value of option)
    {
        if option is a
            flagA = 1; //on flag
            break;
        if option is l
            flagL = 1; //on flag
            break;
        if option is h
            flagh=1; //on flag
            break;
        if option is S
            flagS=1; //on flag
            break;
        if option is r
            flagr=1; //on flag
            break;
    }
}

////////////////////////////////

////////////////////////////////move parameter for other working////////////////////////////////
char paname[256][256]={'\0'};
char sortname[256][256]={'\0'};
while(repeat function before all argument vector copies) //move content of parameter
    strcpy(paname[i],argv[i]);
////////////////////////////////

    practice function of matching for checking command of match //find match

////////////////////////////////Sorting argument vector////////////////////////////////
    Move argument vector to other string(sortname)

    check character of '/' from sortname[0]

    char tmp[256]={'\0'};

    To sort of vector that is has each command
    //////////////////////////////////

```

```

////////////////////////////////move parameter////////////////////////////////
while(brfor read all argv)
{
    if(First name of argv is '-')
    {
        strcpy(Subargv[count],argv[i]);
        count++;
    }
}

while(brfor read all argv)
{
    if(File is not read)
    {
        strcpy(Subargv[count],argv[i]);
        count++;
    }
}

while(brfor read all argv)
{
    Read file;
    if(File is not directory)
    {
        strcpy(Subargv[count],argv[i]);
        count++;
    }
}

while(brfor read all argv)
{
    if(File is directory)
    {
        strcpy(Subargv[count],argv[i]);
        count++;
    }
}
////////////////////////////////

```

```

if(option is S) //check option of S(size)
    sortingfile(Subargv,argc); //practice sorting

if (parameter nubmer is 1)
{
    dirp = opendir(current directory)

    do
    {
        dir = readdir(dirp);

        if (File is not opened)
            break;

        else
        {
            lstat(dir->d_name,&buf);
            InsertNode;
        }

    } while (1);

    Print List
    return 0;
}

```

```

while (befor read all argv)
{
    if (File is not opended)
    {
        Print error;
        continue;
    }
    else
    {
        if (File is directory)
        {
            dirp = open directroy

            do
            {
                dir = readdir(dirp);

                if (Directory is not opended)
                    break;
                else
                {
                    lstat(dir->d_name,&buf);
                    InsertNode;
                }
            } while (1);

            Print List
        }
        else
            printfile(flagA,flagL,Subargv[i]);
    }
}

return 0;
}

```

## Printlist

```
void printlist(DIR *dirp, int flagA, int flagL,int total)
{
    struct group *gid;
    struct passwd *uid;
    struct tm *time;
    char buff[256];
    char cwd[256];
    struct stat buf;
    struct dirent *dir;

    if (option L is on)
    {
        Print current working directory;
        total block;
    }

    while (befor check all node)
    {
        if (option L is on)
        {
            print file information detail;

            if(The file is link file)
                strcat(filename, realpath);
        }
        print file name;
    }
    return;
}
```

(if option A is on, print include hidden file)

## Nodeinsert

```
void Nodeinsert(char* name)
{
    char Ename[256];
    char Etmpname[256];
    Node* tmp = Head;
    Node* prevnode;

    if (Head of list is NULL)
    {
        Makes newnode;
        strcpy(newnode->filename, name);
        newnode->next = NULL;
        newnode->prev = NULL;

        Head = newnode;
        return;
    }

    else
    {
        while (tmp is not NULL)
        {
            strcpy(Ename, name);
            strcpy(Etmpname, tmp->filename);

            if (checkstring(Ename))
                Eliminate character of '.'
            if (checkstring(Etmpname))
                Eliminate character of '.'

            if Etmpname is bigger than name)
            {
                if (tmp is Head)
                {
                    Makes newnode;
                    strcpy(newnode->filename, name);
                    newnode->prev = NULL;

                    newnode->next = Head;
                    Head->prev = newnode;
                    Head = newnode;
                    return;
                }
            }
        }
    }
}
```

```

else
{
    Makes newnode;
    strcpy(newnode->filename, name);
    newnode->next = tmp;
    newnode->prev = tmp->prev;
    tmp->prev->next = newnode;
    tmp->prev = newnode;
    return;
}
}
else continue;
}
Move prev to Tail;
Makes newnode;

strcpy(newnode->filename, name);
newnode->next = NULL;
prevnode->next = newnode;
newnode->prev = prevnode;
Tail = newnode;
return;
}
}

```



#### Function of others

```
void Eliminate(char *str, char ch)
{
    while (;before check all character of string;str++)
    {
        if (*str == ch)
        {
            strcpy(str, str + 1);
            str--;
            return;
        }
    }
}
```

```
void deletelist()
{
    Node* tmp = Head;
    for (; tmp->next != NULL;)
    {
        Head = tmp->next;
        free(tmp);
        tmp = Head;
    }
    free(tmp);
    Head = NULL;
}
```

```

char printType(mode_t mode)
{
    switch (mode & S_IFMT)
    {
        case S_IFREG:
            return('-');
        case S_IFDIR:
            return('d');
        case S_IFCHR:
            return('c');
        case S_IFBLK:
            return('b');
        case S_IFLNK:
            return('l');
        case S_FIFO:
            return('p');
        case S_IFSOCK:
            return('s');
    }

    return('?');
}

char *printPerm(mode_t mode)
{
    int i;
    char *p;
    static char perms[10];
    p = perms;
    strcpy(perms, "-----");

    for (i = 0; i < 3; i++)
    {
        if (mode & (S_IREAD >> i * 3))
            *p = 'r';

        p++;

        if (mode & (S_IWRITE >> i * 3))
            *p = 'w';

        p++;

        if (mode & (S_IEXEC >> i * 3))
            *p = 'x';

        p++;
    }

    if ((mode & S_ISUID) != 0)
        perms[2] = 's';

    if ((mode & S_ISGID) != 0)
        perms[5] = 's';

    if ((mode & S_ISVTX) != 0)
        perms[8] = 't';

    return(perms);
}

```

```

void printOph(long int size)
{
    int k=0,flag=0;
    double sub_size;

    sub_size=(double)size;

    for(Repeat until undivided)
    {
        if(sub_size>(double)1024)
            sub_size/=1024;

        else
            break;
    }

    Execute unit alignment process

    if(If the decimal point is not zero) //check first decimal point
        flag=1;

    int size_int=0;

```

---

```

    if(k==0)
    {

        size_int=(int)sub_size;
        printf(integer output of size_int);

    }

    else if(k==1)
    {
        if(flag==1)
        {
            size_int=(int)sub_size;
            printf(integer output of size_int,"K");
        }
        else
            printf(float output of sub_size,"K");
    }

    else if(k==2)
    {
        if(flag==1)
        {
            size_int=(int)sub_size;
            printf(integer output of size_int,"M");
        }
        else
            printf(float output of sub_size,"M");
    }
}

```

```

else if(k==3)
{
    if(flag==1)
    {
        size_int=(int)sub_size;
        printf(integer output of size_int,"G");
    }
    else
        printf(float output of sub_size,"G");
}

else if(k==4)
{
    if(flag==1)
    {
        size_int=(int)sub_size;
        printf(integer output of size_int,"T");
    }
    else
        printf(float output of sub_size,"T");
}
return;
}

```

```

int matchfunction(char (*paname)[256], int argc)
{
    char Matchcmd[256][256]={0};
    int matchcount=0;

    //////////////////////////////////////////searching match command////////////////////////////////////////.
    while(read all command)
    {
        while(read all character of one string)
        {
            if(string has '*' or '?') //if the command has '*' or '?'
            {
                copy string to command vector
                matchcount++; //increase
                break;
            }
            else if(string has '[x-y]' format) //if the command has 'index'
            {
                copy string to command vector
                matchcount++; //increase
                break;
            }
            else if(string has '[x]' format)
            {
                copy string to command vector
                matchcount++; //increase
                break;
            }
        }
    }

    //////////////////////////////////////////.

    //////////////////////////////////////////delete match cmd////////////////////////////////////////,

    Remove extracted commands from existing vectors

    //////////////////////////////////////////.

```

```

int newargc=argc; //integer
int flag=0; //integer
struct dirent *dir;
struct stat buf;
DIR *dirp;
char pathname[256][256]={0};
char cmd[256][256]={0};

while(checking all string) //for check match command
{
    To separate into path and command parts
}

while(Repeat by match count) //for check command
{
    int currentflag=0; //declare flag

    if(string length is 0) //check path name
        dirp=opendir(".");
    else
    {
        currentflag=1; //on flag
        dirp=opendir(pathname[j]); //open pathname directory
    }
    change directory;

    if(In case there is no matching command) //Check null
    {
        Save back to original vector;
        newargc++;
        continue;
    }
    do
    {
        dir = readdir(dirp); //read file
        if (dir == NULL) //check NULL
            break; //stop repeat function

        else
        {
            if(!fnmatch(pattern[i],file name,0)&&(dir->d_name[0]!='.')) //function of match
            {
                Perform a function that makes it the original path;

                Save the filename that matches the original vector;
                newargc++;
                flag=1; //on flag
            }
        }
    } while (1);

    if(flag==0) //if flag is 0
    {
        Save back to original vector;
        newargc++;
    }
    flag=0; //off flag
    change pointer dirp to start
}
return newargc; //return new vector count
}

```

## 4. Result

### PDF 예시

```
sp20157522025@ubuntu:~$ ./spls_final -lahS
Directory path: /home/sp20157522025
-rwxrwxrwx- 1 sp20157522025 sp20157522025 1.2M Apr 18 01:09 tue_2-2_2015722025.pdf
-rw-rw-r-- 1 sp20157522025 sp20157522025 37.6K Apr 25 06:16 spls_advanced.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 36.7K Apr 25 06:24 spls_final.c
-rwxrwxr-x 1 sp20157522025 sp20157522025 36.6K Apr 25 06:38 spls_final
-rw----- 1 sp20157522025 sp20157522025 35.5K Apr 25 06:25 .bash_history
-rw----- 1 sp20157522025 sp20157522025 24.8K Apr 25 06:34 .ICEauthority
-rw-r--r-- 1 sp20157522025 sp20157522025 12K Mar 29 09:14 .ACD.swp
-rw----- 1 sp20157522025 sp20157522025 12K Mar 16 08:40 .hello.swp
-rw-r--r-- 1 sp20157522025 sp20157522025 12K Apr 03 06:46 .Makefile.swo
-rw-r--r-- 1 sp20157522025 sp20157522025 12K Apr 25 05:08 .Makefile.swp
-rw-rw-r-- 1 sp20157522025 sp20157522025 8.7K Apr 18 01:09 examples.desktop
-rw-rw-r-- 1 sp20157522025 sp20157522025 8.1K Apr 18 01:09 A.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 7.2K Apr 18 01:09 optionsample.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 7.2K Apr 18 01:09 simple_ls.c
drwxr-xr-x 25 sp20157522025 sp20157522025 4K Apr 25 06:38 .
drwxr-xr-x 3 root root 4K Mar 15 19:55 ..
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 03 22:54 ...
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 22 21:49 A
drwxrwxr-x 3 sp20157522025 sp20157522025 4K Apr 22 21:12 abTde
drwx----- 16 sp20157522025 sp20157522025 4K Apr 02 03:20 .cache
drwx----- 3 sp20157522025 sp20157522025 4K Mar 21 01:45 .compiz
drwx----- 15 sp20157522025 sp20157522025 4K Mar 18 20:15 .config
drwxr-xr-x 2 sp20157522025 sp20157522025 4K Apr 18 01:09 Desktop
drwxr-xr-x 2 sp20157522025 sp20157522025 4K Apr 18 01:09 Documents
drwxr-xr-x 2 sp20157522025 sp20157522025 4K Apr 18 01:09 Downloads
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 18 01:09 ET
drwx----- 2 sp20157522025 sp20157522025 4K Mar 15 22:08 .gconf
drwx----- 3 sp20157522025 sp20157522025 4K Apr 25 06:34 .gnupg
-rw----- 1 sp20157522025 sp20157522025 4K Mar 16 08:42 .hello.swo
drwxrwxr-x 3 sp20157522025 sp20157522025 4K Apr 18 01:09 J
drwx----- 3 sp20157522025 sp20157522025 4K Mar 15 20:33 .local
drwx----- 5 sp20157522025 sp20157522025 4K Mar 18 20:51 .mozilla
drwxr-xr-x 2 sp20157522025 sp20157522025 4K Apr 18 01:09 Music
drwxr-xr-x 2 sp20157522025 sp20157522025 4K Apr 18 01:09 Pictures
drwxr-xr-x 2 sp20157522025 sp20157522025 4K Apr 18 01:09 Public
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 25 00:45 save
drwxr-xr-x 2 sp20157522025 sp20157522025 4K Apr 18 01:09 Templates
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 20 04:35 test
drwxrwxr-x 4 sp20157522025 sp20157522025 4K Apr 18 01:09 test2
```

```
sp20157522025@ubuntu:~$ ./spls_final '/Not_dir/*'
ls: cannot access '/Not_dir/*': No such file or directory
```

```
sp20157522025@ubuntu:~$ ./spls_final '/home/sp20157522025/*'
/home/sp20157522025/A.c
/home/sp20157522025/a.c
/home/sp20157522025/A1
/home/sp20157522025/A2
/home/sp20157522025/A[1-9]
/home/sp20157522025/AA
/home/sp20157522025/ab?de
/home/sp20157522025/abJde
/home/sp20157522025/B.c
/home/sp20157522025/C
/home/sp20157522025/C.c
/home/sp20157522025/examples.desktop
/home/sp20157522025/Makefile
/home/sp20157522025/optionsample.c
/home/sp20157522025/simple_ls.c
/home/sp20157522025/SON
/home/sp20157522025/spls_advanced.c
/home/sp20157522025/spls_final
/home/sp20157522025/spls_final.c
/home/sp20157522025/tue_2-2_2015722025.pdf
/home/sp20157522025/txt
/home/sp20157522025/txt1
/home/sp20157522025/txt2
/home/sp20157522025/txt8
```

Directory path: /home/sp20157522025/A

```
A
a
D
dddd
spls_advanced
spls_advanced.c
X
```

Directory path: /home/sp20157522025/abTde

```
AA
```

Directory path: /home/sp20157522025/Desktop

Directory path: /home/sp20157522025/Documents

```
sp20157522025@ubuntu:~$ ./spls_final 'text[1-4].txt'
text1.txt
text2.txt
text3.txt
sp20157522025@ubuntu:~$ ./spls_final 't??t?.txt'
text1.txt
text2.txt
text3.txt
text9.txt
```



## 직접 작성한 예시

```
sp20157522025@ubuntu:~$ ./spls_final test -l
Directory path: /home/sp20157522025/test
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 CC
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 HH
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 Makefile
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 oo
-rwxrwxr-x 1 sp20157522025 sp20157522025 9840 Apr 18 17:31 simple_ls
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 simple_ls.c
-rwxrwxr-x 1 sp20157522025 sp20157522025 36640 Apr 20 04:17 spls_advanced
-rwxrw-rw- 1 sp20157522025 sp20157522025 365912 Apr 02 05:35 tue_2-1_2015722025 (1).tar.gz
-rwxrw-rw- 1 sp20157522025 sp20157522025 419771 Apr 02 03:39 tue_2-1_2015722025.pdf
sp20157522025@ubuntu:~$ ./spls_final test -lr
Directory path: /home/sp20157522025/test
-rwxrw-rw- 1 sp20157522025 sp20157522025 419771 Apr 02 03:39 tue_2-1_2015722025.pdf
-rwxrw-rw- 1 sp20157522025 sp20157522025 365912 Apr 02 05:35 tue_2-1_2015722025 (1).tar.gz
-rwxrwxr-x 1 sp20157522025 sp20157522025 36640 Apr 20 04:17 spls_advanced
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 simple_ls.c
-rwxrwxr-x 1 sp20157522025 sp20157522025 9840 Apr 18 17:31 simple_ls
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 oo
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 Makefile
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 HH
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 CC
```

```
sp20157522025@ubuntu:~$ ./spls_final test -ls
Directory path: /home/sp20157522025/test
-rwxrw-rw- 1 sp20157522025 sp20157522025 419771 Apr 02 03:39 tue_2-1_2015722025.pdf
-rwxrw-rw- 1 sp20157522025 sp20157522025 365912 Apr 02 05:35 tue_2-1_2015722025 (1).tar.gz
-rwxrwxr-x 1 sp20157522025 sp20157522025 36640 Apr 20 04:17 spls_advanced
-rwxrwxr-x 1 sp20157522025 sp20157522025 9840 Apr 18 17:31 simple_ls
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 CC
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 HH
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 oo
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 simple_ls.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 Makefile
sp20157522025@ubuntu:~$ ./spls_final test -lsr
Directory path: /home/sp20157522025/test
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 Makefile
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 simple_ls.c
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 oo
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 HH
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 CC
-rwxrwxr-x 1 sp20157522025 sp20157522025 9840 Apr 18 17:31 simple_ls
-rwxrwxr-x 1 sp20157522025 sp20157522025 36640 Apr 20 04:17 spls_advanced
-rwxrw-rw- 1 sp20157522025 sp20157522025 365912 Apr 02 05:35 tue_2-1_2015722025 (1).tar.gz
-rwxrw-rw- 1 sp20157522025 sp20157522025 419771 Apr 02 03:39 tue_2-1_2015722025.pdf
sp20157522025@ubuntu:~$ ./spls_final test -lsrh
Directory path: /home/sp20157522025/test
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 Makefile
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 simple_ls.c
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 26 01:02 oo
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 26 01:02 HH
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 26 01:02 CC
-rwxrwxr-x 1 sp20157522025 sp20157522025 9.6K Apr 18 17:31 simple_ls
-rwxrwxr-x 1 sp20157522025 sp20157522025 35.7K Apr 20 04:17 spls_advanced
-rwxrw-rw- 1 sp20157522025 sp20157522025 357.3K Apr 02 05:35 tue_2-1_2015722025 (1).tar.gz
-rwxrw-rw- 1 sp20157522025 sp20157522025 409.9K Apr 02 03:39 tue_2-1_2015722025.pdf
```

```

sp20157522025@ubuntu:~$ ./spls_final test test2 -l
Directory path: /home/sp20157522025/test
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 CC
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 HH
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 Makefile
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 oo
-rwxrwxr-x 1 sp20157522025 sp20157522025 9840 Apr 18 17:31 simple_ls
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 simple_ls.c
-rwxrwxr-x 1 sp20157522025 sp20157522025 36640 Apr 20 04:17 spls_advanced
-rwxrwxr-x 1 sp20157522025 sp20157522025 365912 Apr 02 05:35 tue_2-1_2015722025 (1).tar.gz
-rwxrwxr-x 1 sp20157522025 sp20157522025 419771 Apr 02 03:39 tue_2-1_2015722025.pdf

```

```

Directory path: /home/sp20157522025/test2
drwxrwxr-x 3 sp20157522025 sp20157522025 4096 Mar 29 18:17 A
-rw-rw-r-- 1 sp20157522025 sp20157522025 73 Mar 29 06:28 A.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 28 22:44 AA
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 AB
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 26 04:16 aBcd
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Mar 29 18:06 ABCD
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 26 04:16 ACbd
-rw-rw-r-- 1 sp20157522025 sp20157522025 73 Mar 22 05:37 B.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 BB
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 CA
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 CC
-rw-rw-r-- 1 sp20157522025 sp20157522025 2425 Mar 28 23:42 ls.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 46 Apr 08 02:56 Makefile
-rwxrwxr-x 1 sp20157522025 sp20157522025 13544 Mar 28 04:04 spls
-rwxrwxr-x 1 sp20157522025 sp20157522025 24152 Apr 08 03:34 spls_advanced
-rw-rw-r-- 1 sp20157522025 sp20157522025 8368 Apr 08 03:39 spls_advanced.c

```

```

sp20157522025@ubuntu:~$ ./spls_final test test2 -lr
Directory path: /home/sp20157522025/test2
-rw-rw-r-- 1 sp20157522025 sp20157522025 8368 Apr 08 03:39 spls_advanced.c
-rwxrwxr-x 1 sp20157522025 sp20157522025 24152 Apr 08 03:34 spls_advanced
-rwxrwxr-x 1 sp20157522025 sp20157522025 13544 Mar 28 04:04 spls
-rw-rw-r-- 1 sp20157522025 sp20157522025 46 Apr 08 02:56 Makefile
-rw-rw-r-- 1 sp20157522025 sp20157522025 2425 Mar 28 23:42 ls.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 CC
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 CA
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 BB
-rw-rw-r-- 1 sp20157522025 sp20157522025 73 Mar 22 05:37 B.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 26 04:16 ACbd
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Mar 29 18:06 ABCD
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 26 04:16 aBcd
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 AB
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 28 22:44 AA
-rw-rw-r-- 1 sp20157522025 sp20157522025 73 Mar 29 06:28 A.c
drwxrwxr-x 3 sp20157522025 sp20157522025 4096 Mar 29 18:17 A

```

```

Directory path: /home/sp20157522025/test
-rwxrwxr-x 1 sp20157522025 sp20157522025 419771 Apr 02 03:39 tue_2-1_2015722025.pdf
-rwxrwxr-x 1 sp20157522025 sp20157522025 365912 Apr 02 05:35 tue_2-1_2015722025 (1).tar.gz
-rwxrwxr-x 1 sp20157522025 sp20157522025 36640 Apr 20 04:17 spls_advanced
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 simple_ls.c
-rwxrwxr-x 1 sp20157522025 sp20157522025 9840 Apr 18 17:31 simple_ls
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 oo
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 Makefile
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 HH
drwxrwxr-x 2 sp20157522025 sp20157522025 4096 Apr 26 01:02 CC

```

```

sp20157522025@ubuntu:~$ ./spls_final test test2 -lSh
Directory path: /home/sp20157522025/test
-rwxrw-rw- 1 sp20157522025 sp20157522025 409.9K Apr 02 03:39 tue_2-1_2015722025.pdf
-rwxrw-rw- 1 sp20157522025 sp20157522025 357.3K Apr 02 05:35 tue_2-1_2015722025 (1).tar.gz
-rwxrwxr-x 1 sp20157522025 sp20157522025 35.7K Apr 20 04:17 spls_advanced
-rwxrwxr-x 1 sp20157522025 sp20157522025 9.6K Apr 18 17:31 simple_ls
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 26 01:02 CC
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 26 01:02 HH
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 26 01:02 oo
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 simple_ls.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 Makefile

Directory path: /home/sp20157522025/test2
-rwxrwxr-x 1 sp20157522025 sp20157522025 23.5K Apr 08 03:34 spls_advanced
-rwxrwxr-x 1 sp20157522025 sp20157522025 13.2K Mar 28 04:04 spls
-rw-rw-r-- 1 sp20157522025 sp20157522025 8.1K Apr 08 03:39 spls_advanced.c
drwxrwxr-x 3 sp20157522025 sp20157522025 4K Mar 29 18:17 A
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Mar 29 18:06 ABCD
-rw-rw-r-- 1 sp20157522025 sp20157522025 2.3K Mar 28 23:42 ls.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 73 Mar 29 06:28 A.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 73 Mar 22 05:37 B.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 46 Apr 08 02:56 Makefile
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 28 22:44 AA
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 AB
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 26 04:16 aBcd
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 26 04:16 ACbd
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 BB
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 CA
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 CC

```

```

sp20157522025@ubuntu:~$ ./spls_final test test2 -lShr
Directory path: /home/sp20157522025/test2
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 CC
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 CA
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 BB
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 26 04:16 ACbd
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 26 04:16 aBcd
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 25 23:06 AB
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Mar 28 22:44 AA
-rw-rw-r-- 1 sp20157522025 sp20157522025 46 Apr 08 02:56 Makefile
-rw-rw-r-- 1 sp20157522025 sp20157522025 73 Mar 22 05:37 B.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 73 Mar 29 06:28 A.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 2.3K Mar 28 23:42 ls.c
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Mar 29 18:06 ABCD
drwxrwxr-x 3 sp20157522025 sp20157522025 4K Mar 29 18:17 A
-rw-rw-r-- 1 sp20157522025 sp20157522025 8.1K Apr 08 03:39 spls_advanced.c
-rwxrwxr-x 1 sp20157522025 sp20157522025 13.2K Mar 28 04:04 spls
-rwxrwxr-x 1 sp20157522025 sp20157522025 23.5K Apr 08 03:34 spls_advanced

Directory path: /home/sp20157522025/test
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 Makefile
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 simple_ls.c
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 26 01:02 oo
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 26 01:02 HH
drwxrwxr-x 2 sp20157522025 sp20157522025 4K Apr 26 01:02 CC
-rwxrwxr-x 1 sp20157522025 sp20157522025 9.6K Apr 18 17:31 simple_ls
-rwxrwxr-x 1 sp20157522025 sp20157522025 35.7K Apr 20 04:17 spls_advanced
-rwxrw-rw- 1 sp20157522025 sp20157522025 357.3K Apr 02 05:35 tue_2-1_2015722025 (1).tar.gz
-rwxrw-rw- 1 sp20157522025 sp20157522025 409.9K Apr 02 03:39 tue_2-1_2015722025.pdf

```

```

sp20157522025@ubuntu:~$ ls abTde/AA
1ABC 2DER 3TTT 4TFF 5avcxn file1 file2 file3 file4 file5
sp20157522025@ubuntu:~$ ls test
CC HH Makefile oo simple_ls simple_ls.c spls_advanced tue_2-1_2015722025 (1).tar.gz tue_2-1_2015722025.pdf
sp20157522025@ubuntu:~$ ./spls_final 'test/*' 'abTde/AA/*[1-4]'
abTde/AA/file1
abTde/AA/file2
abTde/AA/file3
abTde/AA/file4
test/Makefile
test/simple_ls
test/simple_ls.c
test/spls_advanced
test/tue_2-1_2015722025 (1).tar.gz
test/tue_2-1_2015722025.pdf

Directory path: /home/sp20157522025/test/CC
Directory path: /home/sp20157522025/test/HH
Directory path: /home/sp20157522025/test/oo

```

```
sp20157522025@ubuntu:~$ ./spls_final 'test/*' 'abTde/AA/*[1-4]' -r
test/tue_2-1_2015722025.pdf
test/tue_2-1_2015722025 (1).tar.gz
test/spls_advanced
test/simple_ls.c
test/simple_ls
test/Makefile
abTde/AA/file4
abTde/AA/file3
abTde/AA/file2
abTde/AA/file1
```

Directory path: /home/sp20157522025/test/oo

Directory path: /home/sp20157522025/test/HH

Directory path: /home/sp20157522025/test/CC

```
sp20157522025@ubuntu:~$ ./spls_final 'test/*' 'abTde/AA/*[1-4]' -rl
-rwxrwxrwx- 1 sp20157522025 sp20157522025 419771 Apr 02 03:39 test/tue_2-1_2015722025.pdf
-rwxrwxrwx- 1 sp20157522025 sp20157522025 365912 Apr 02 05:35 test/tue_2-1_2015722025 (1).tar.gz
-rwxrwxr-x 1 sp20157522025 sp20157522025 36640 Apr 20 04:17 test/spls_advanced
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 test/simple_ls.c
-rwxrwxr-x 1 sp20157522025 sp20157522025 9840 Apr 18 17:31 test/simple_ls
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 test/Makefile
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Apr 23 01:10 abTde/AA/file4
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Apr 23 01:10 abTde/AA/file3
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Apr 23 01:10 abTde/AA/file2
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Apr 23 01:10 abTde/AA/file1
```

Directory path: /home/sp20157522025/test/oo

Directory path: /home/sp20157522025/test/HH

Directory path: /home/sp20157522025/test/CC

```
sp20157522025@ubuntu:~$ ./spls_final 'test/*' 'abTde/AA/*[1-4]' -rls
-rwxrwxrwx- 1 sp20157522025 sp20157522025 419771 Apr 02 03:39 test/tue_2-1_2015722025.pdf
-rwxrwxrwx- 1 sp20157522025 sp20157522025 365912 Apr 02 05:35 test/tue_2-1_2015722025 (1).tar.gz
-rwxrwxr-x 1 sp20157522025 sp20157522025 36640 Apr 20 04:17 test/spls_advanced
-rwxrwxr-x 1 sp20157522025 sp20157522025 9840 Apr 18 17:31 test/simple_ls
-rw-rw-r-- 1 sp20157522025 sp20157522025 365 Apr 18 17:31 test/simple_ls.c
-rw-rw-r-- 1 sp20157522025 sp20157522025 38 Apr 01 04:36 test/Makefile
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Apr 23 01:10 abTde/AA/file4
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Apr 23 01:10 abTde/AA/file3
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Apr 23 01:10 abTde/AA/file2
-rw-rw-r-- 1 sp20157522025 sp20157522025 0 Apr 23 01:10 abTde/AA/file1
```

Directory path: /home/sp20157522025/test/oo

Directory path: /home/sp20157522025/test/HH

Directory path: /home/sp20157522025/test/CC



```
sp20157522025@ubuntu:~$ ./spls_final '/bin/*'
```

```
/bin/bash  
/bin/bsd-csh  
/bin/bunzip2  
/bin/busybox  
/bin/bzcat  
/bin/bzcmp  
/bin/bzdiff  
/bin/bzegrep  
/bin/bzexe  
/bin/bzfgrep  
/bin/bzgrep  
/bin/bzip2  
/bin/bzip2recover  
/bin/bzless  
/bin/bzmore  
/bin/cat  
/bin/chacl  
/bin/chgrp  
/bin/chmod  
/bin/chown  
/bin/chvt  
/bin/cp  
/bin/cpio  
/bin/csh  
/bin/dash  
/bin/date  
/bin/dd  
/bin/df  
/bin/dir  
/bin/dmesg  
/bin/dnsdomainname
```

```
sp20157522025@ubuntu:~$ ./spls_final '*'
A.c
a.c
A1
A2
A[1-9]
AA
ab?de
abJde
B.c
C
C.c
examples.desktop
Makefile
optionsample.c
simple_ls.c
SON
spls_advanced.c
spls_final
spls_final.c
text1.txt
text2.txt
text3.txt
text9.txt
tue_2-2_2015722025.pdf
txt
txt1
txt2
txt8
```

```
Directory path: /home/sp20157522025/A
A
a
D
dddd
run
spls_advanced
spls_advanced.c
spls_advanced.o
X

Directory path: /home/sp20157522025/abTde
AA

Directory path: /home/sp20157522025/Desktop

Directory path: /home/sp20157522025/Documents

Directory path: /home/sp20157522025/Downloads

Directory path: /home/sp20157522025/ET

Directory path: /home/sp20157522025/HONG
CC
HH
Makefile
oo
simple_ls
simple_ls.c
spls_advanced
tue_2-1_2015722025 (1).tar.gz
tue_2-1_2015722025.pdf

Directory path: /home/sp20157522025/J
A
K

Directory path: /home/sp20157522025/Min
CC
```

```
sp20157522025@ubuntu:~$ ./spls_final '*.*'
A.c
a.c
B.c
C.c
examples.desktop
optionsample.c
simple_ls.c
spls_advanced.c
spls_final.c
text1.txt
text2.txt
text3.txt
text9.txt
tue_2-2_2015722025.pdf
```

## 5. Conclusion

이번 과제를 진행하는데 있어 가장 까다로운 상황은 그 전 과제를 구현하면서 정해놓은 조건들이 이번 과제를 수행하기 위해 추가적인 조건과 예외처리를 해주어야 깔끔하게 잘 동작하는 점이 있었습니다. 또한 match와 argument vector를 사용하면서 출력을 위해 상당히 복잡하고 많은 sorting함수를 사용하는 문제가 까다로웠으며, 실질적으로 코드도 복잡하게 짜여있다. 또한 처음에는 argv에 있는 인자들을 직접 sorting하는 방법으로 문제 해결을 시도했지만 main함수로 직접 받아오는 argv를 쓰는데 있어 일반 array를 사용하는 것과 다른 부분이 많았다, 예를 들면 일반 array에서는 sorting이 정상적으로 동작하지만 argv에 해당 알고리즘을 적용하면, 임의의 성분이 없어지거나 변형이 되는 문제가 발생하는 것을 알게 되었다. 정확한 이유는 아직 모르지만 이 부분은 다음 구현을 위하여 짚고 넘어가야 할 문제 같다.