

SW 프로그래밍(2)

Final Presentation

20175528 이민지

<목 차>

1. 파일 내에서 해당 단어 찾기

2. 랭킹 매기기

3. 완성 코드



<파일 내에서 해당 단어 찾기>

⇒ 해당 단어가 파일에 있는 횟수가 높을 수록 앞쪽에 랭킹.

일치하는 단어가 있다면?

1. 모두 소문자
2. 모두 대문자
3. 첫글자만 대문자



일치하는 단어가 없다면?

⇒ 해당 단어와 파일의 단어 유사도가 높을수록 앞쪽에 랭킹.

```
typedef struct hello
{
    char filename[30];
    int yescount;
    int contain;
    int coin;
    int nocount;
}bello;
```

-> 파일 별 데이터를 저장하기 위한 구조체의 선언

```
typedef struct yes //단어 있는 경우
{
    char yesname[30];
    int yescount;
}Yes;

typedef struct include //없는데 포함하는 경우
{
    char innname[30];
    int contain;
}In;

typedef struct coincide //없는데 앞글자 일치 정도(유사도)
{
    char coincname[30];
    int coin;
    int coincount;
}Co;
```

-> 값 비교하기 편하기 위해 선언한 구조체

-> 단어 있는 경우

없는데 포함하는 경우

없는데 앞 글자 일치 정도

```
int main()
{
    char str[100], word[100], b1[100], b2[100];
    int count = 0, size, i, num = 0, j, m, n;
    int include = 0; // 사용자가 입력한 단어가 파일의 단어에 포함되는 경우
    int noword = 0; // 사용자가 입력한 단어가 파일의 단어에 앞부분에 일부 있을 때
    int nowordmax = 2; // 위의 숫자의 경우 중 제일 큰 경우(글자수가 가장 많이 일치할 경우)
    int nowordmaxc = 0;
    struct _finddata_t fd;
    long handle;
    int result = 1;
    int filecount = 0;
    char filename[30];
    char ChangeDir[60] = { "c:dataset-2nd\\" }; // 디렉토리 바꿔줄꺼임
    int rank;

    bello *score;
    Yes *have;
    In *in;
    Co *coin;
```

→ 사용한 변수(int 형 변수 rank는 아직 사용 안함)

```
int filecounter()
{
    struct _finddata_t fd1;
    long handle;
    int result = 1;
    int filecount = 0;

    handle = _findfirst("c:\\dataset-2nd\\*.*", &fd1);
    if (handle == -1) return;
    while (result != -1)
    {
        if ((strcmp(".", fd1.name)) != 0 && (strcmp(.., fd1.name)) != 0)
        {
            filecount++;
        }
        result = _findnext(handle, &fd1);
    }
    _findclose(handle);

    return filecount;
}
```

-> 폴더 내의 파일 개수를
세기 위한 filecounter라는
함수 생성

-> filecounter의 값을 이용하여 구조체
변수들의 메모리 크기 할당.

```
printf("%d \n", filecounter());

if ((score = (bello*)malloc(sizeof(bello)*filecounter())) == NULL)
{
    printf("score에 자리없음!");
    exit(1);
}

if ((have = (Yes*)malloc(sizeof(Yes)*filecounter())) == NULL)
{
    printf("have에 자리없음!");
    exit(1);
}

if ((in = (In*)malloc(sizeof(In)*filecounter())) == NULL)
{
    printf("in에 자리없음!");
    exit(1);
}

if ((coin = (Co*)malloc(sizeof(Co)*filecounter())) == NULL)
{
    printf("coin에 자리없음!");
    exit(1);
}
```

```
handle = _findfirst("c:\\dataset-2nd\\*.*", &fd);
if (handle == -1) return;
while (result != -1)
{
    if ((strcmp(".", fd.name)) != 0 && (strcmp("..", fd.name)) != 0)
    {
        strcpy(score[filecount].filename, fd.name);
        filecount++;
    }
    result = _findnext(handle, &fd);
}
_findclose(handle);
```

-> hello 구조체에 파일의 이름 저장.

```
printf("%d \n", filecount);
```

```
printf("원하는 단어를 입력하시오. : ");  
scanf("%s", str);
```

```
size = strlen(str);
```

-> 찾고자 하는
단어 입력.

```
if (96 < str[0] && str[0] < 123) //str이 소문자로  
{  
    strcpy(b1, str);  
    b1[0] = str[0] - 32; //str의 제일 첫번째 글자  
    strcpy(b2, str);  
   strupr(b2); //str을 전부 대문자로 바꿔준다.  
}  
  
else if (64 < str[0] && str[0] < 91) //첫 글자가  
{  
    if (64 < str[1] && str[1] < 91) // 두번째 글자  
{  
        strcpy(b1, str);  
        strlwr(b1);  
        b1[0] = b1[0] - 32; //str을 첫번째 문자를 소문자로  
        strcpy(b2, str);  
        strlwr(b2); //str을 전부 소문자로 바꿔준다  
    }  
  
    else if (96 < str[1] && str[1] < 123) // 두번째 글자가 대문자  
{  
        strcpy(b1, str);  
        b1[0] = str[0] + 32; //첫번째 문자를 소문자로  
        strcpy(b2, str);  
       strupr(b2); //str을 전부 대문자로 바꿔준다  
    }  
}
```

-> 정확한 측정을 위해
단어를 3가지 경우로 만
들어 준다.

1. 모두 대문자
2. 모두 소문자
3. 첫글자만 대문자

```

void frontchecker(char str[100])
{
    int size, i;
    char front;

    size = strlen(str);
    front = str[0];

    switch (front)
    {
        case '(':
            for (i = 0; i < size - 1; i++)
                str[i] = str[i + 1];
            str[size - 1] = '\0';
            break;
        case '-':
            for (i = 0; i < size - 1; i++)
                str[i] = str[i + 1];
            str[size - 1] = '\0';
            break;
        case '\'':
            for (i = 0; i < size - 1; i++)
                str[i] = str[i + 1];
            str[size - 1] = '\0';
            break;
        case '\"':
            for (i = 0; i < size - 1; i++)
                str[i] = str[i + 1];
            str[size - 1] = '\0';
            break;
        default:
            break;
    }
}

```

```

void lastchecker(char st[100])
{
    int size; //st의 크기
    char last;

    size = strlen(st);
    last = st[size - 1];

    switch (last)
    {
        case ',':
            st[size - 1] = '\0';
            break;
        case '-':
            st[size - 1] = '\0';
            break;
        case '.':
            st[size - 1] = '\0';
            break;
        case ')':
            st[size - 1] = '\0';
            break;
        case '!':
            st[size - 1] = '\0';
            break;
        case '~':
            st[size - 1] = '\0';
            break;
        case '?':
            st[size - 1] = '\0';
            break;
        case '\"':
            st[size - 1] = '\0';
            break;
        case '\'':
            st[size - 1] = '\0';
            break;
        default:
            break;
    }
}

```

-> 단어의 시작
(“ ‘ –

-> 단어의 끝
, .) ! ~ ? “ ‘ –

```

break;
case ',':
    st[size - 1] = '\0';
    break;
case '-':
    st[size - 1] = '\0';
    break;
case '.':
    st[size - 1] = '\0';
    break;
case ')':
    st[size - 1] = '\0';
    break;
case '!':
    st[size - 1] = '\0';
    break;
case '~':
    st[size - 1] = '\0';
    break;
case '?':
    st[size - 1] = '\0';
    break;
case '\"':
    st[size - 1] = '\0';
    break;
case '\'':
    st[size - 1] = '\0';
    break;
default:
    break;
}

```

-> 단어 찾기 시작.

```
if (chdir(ChangeDir) == 0) //작업 디렉토리 바꿔줌
{
    for (i = 0; i < filecount; i++)
    {
        count = 0;
        include = 0;
        nowordmax = 2;
        nowordmaxc = 0;
        FILE *fp = fopen(score[i].filename, "rt");

        if (fp == NULL)
        {
            printf("이 파일은 비어있습니다!");
            return 1;
        }
        do
        {
            fscanf(fp, "%s", word);
            frontchecker(word);
            lastchecker(word);

            if (strlen(word) == size) // strstr에서 at을
            {
                if (strstr(word, str) != NULL) // 입력한
                    count++;
                else if (strstr(word, b1) != NULL) //입
                    count++;
                else if (strstr(word, b2) != NULL) //
                    count++;
            }
        }
    }
}
```

변경) 각
각 2번씩
널어줌

```
        count++;
    }
    else
        continue;
} while (!feof(fp));

score[i].yescount = count;

if (score[i].yescount == 0) //단어가 없을 경우
{
    fseek(fp, 0L, SEEK_SET); //파일 제일 앞으로
    do
    {
        fscanf(fp, "%s", word);
        frontchecker(word);
        lastchecker(word);

        if (size < strlen(word)) // 파일에서 가
        {
            if (strstr(word, str) != NULL) //
                include++;
            else if (strstr(word, b1) != NULL)
                include++;
            else if (strstr(word, b2) != NULL)
                include++; //포함된 횟수 세는거
        }
        else
            continue;
    } while (!feof(fp));
    score[i].contain = include;
}
```

-> 단어가 있다면 bello 구조체의 yescount 항목에 저장.
없다면 포함 여부 측정. 포함한다면 bello 구조체의
contain 항목에 저장.

```

if (score[i].yescount == 0 && score[i].contain == 0)
{
    fseek(fp, 0L, SEEK_SET); //파일 제일 앞으로 이동.
    do
    {
        fscanf(fp, "%s", word);
        frontchecker(word);
        lastchecker(word);
        noword = 0;

        if (strcmp(str, word, 2) == 0)
        {
            noword = 2; //이미 2개 일치하므로
            for (num = 2; num < size; num++)
            {
                if (str[num] == word[num])
                    noword++;
                else
                    break;
            }
        }

        else if (strcmp(b1, word, 2) == 0)
        {
            noword = 2; //이미 2개 일치하므로
            for (num = 2; num < size; num++)
            {
                if (str[num] == word[num])
                    noword++;
                else
                    break;
            }
        }
    }
}

```

```

        }
    }

    else if ((strcmp(b2, word, 2)) == 0)
    {
        noword = 2; //이미 2개 일치하므로
        for (num = 2; num < size; num++)
        {
            if (str[num] == word[num])
                noword++;
            else
                break;
        }

        if (noword > nowordmax)
        {
            nowordmax = noword;
            nowordmaxc = 0; //max값이 바꼈으니까 초기화
        }

        else if (noword == nowordmax)
        {
            nowordmaxc++; //같은 경우가 많을 수 있음
        }
    }

} while (!feof(fp));
score[i].coin = nowordmax;
score[i].nocount = nowordmaxc;
fclose(fp);
}

```

39판

-> 찾고자하는 단어가 포함도 되지 않는다면 똑같은 글자수 측정 후 똑같은 글자수는 bello 구조체의 coin에 저장, 똑같은 글자수인 경우의 횟수는 nocount에 저장.

```
for (i = 0; i < filecount; i++)
{
    if (score[i].yescount > 0)
        printf("파일명 : %s에는 단어가 있습니다! 있는 개수 : %d \n", score[i].filename, score[i].yescount);
    else if (score[i].yescount == 0)
    {
        if (score[i].contain > 0)
            printf("파일명 : %s에는 단어가 없습니다!, 포함 횟수 : %d \n", score[i].filename, score[i].contain);
        else if (score[i].contain == 0)
        {
            if (score[i].nocount>0)
                printf("파일명 : %s에는 단어가 없습니다!, 일치하는 단어 수 : %d, 횟수 : %d \n", score[i].filename, score[i].coin, score[i].nocount);
            else if (score[i].nocount == 0)
                printf("파일명 : %s에는 그 어떤 비슷한 단어도 없습니다!\n", score[i].filename);
        }
    }
}
```

→ bello 구조체의 score 변수에 제대로 저장 된 것인지 확인용 코드!

```
printf("\n\n새로운 구조체에 넣은 뒤\n\n");
j = 0, n = 0, m = 0;
for (i = 0; i < filecount; i++)
{
    if (score[i].yescount > 0)
    {
        strcpy(have[j].yesname, score[i].filename);
        have[j].yescount = score[i].yescount;
        j++;
    }

    if (score[i].contain > 0)
    {
        strcpy(in[n].inname, score[i].filename);
        in[n].contain = score[i].contain;
        n++;
    }

    if (score[i].nocount > 0)
    {
        strcpy(coin[m].coinname, score[i].filename);
        coin[m].coin = score[i].coin;
        coin[m].coincount = score[i].nocount;
        m++;
    }
}
```

-> ranking을 매기기 편하게
case 별로 새로운 구조체에 저장!

```

for (i = 0; i < filecount; i++)
{
    if (have[i].yescount > 0)
    {
        printf("파일명 %s에는 단어가 %d번 있습니다.\n", have[i].yesname, have[i].yescount);
    }
}

for (i = 0; i < filecount; i++)
{
    if (in[i].contain > 0)
    {
        printf("파일명 %s에는 단어가 %d번 포함되어 있습니다.\n", in[i].inname, in[i].contain);
    }
}

for (i = 0; i < filecount; i++)
{
    if (coin[i].coincount > 0)
    {
        printf("파일명 %s에는 단어와 글자가 %d개 일치하는 것이 %d번 있습니다.\n", coin[i].coinname, coin[i].coin, coin[i].coincount);
    }
}

for (i = 0; i < filecount; i++)
{
    if (!(score[i].yescount > 0) && !(score[i].contain > 0))
    {
        if (!(score[i].nocount > 0))
            printf("파일명 %s에는 일치하거나 유사한 단어가 없습니다!\n", score[i].filename);
    }
}

```

→ 새로운 구조체에 제대로
저장되었는지 확인하기 위한
코드.

C:\WINDOWS\system32\cmd.exe

www.Bandicam.co.kr

```
500  
500  
원하는 단어를 입력하시오. : hello  
파일명 : 100385newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 3  
파일명 : 101678newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 2  
파일명 : 101797newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 2  
파일명 : 10193newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 4  
파일명 : 102444newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 102529newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 10264newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 7  
파일명 : 102760newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 3, 횟수 : 1  
파일명 : 10306newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 103816newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 7  
파일명 : 104277newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 104278newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 3, 횟수 : 1  
파일명 : 104405newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 105327newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 3, 횟수 : 1  
파일명 : 105756newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 105834newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 3, 횟수 : 1  
파일명 : 106247newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 10650newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 10663newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 106762newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 1  
파일명 : 107525newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 3, 횟수 : 1  
파일명 : 108359newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 109096newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 3, 횟수 : 2  
파일명 : 109906newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 2  
파일명 : 110108newsML.txt에는 그 어떤 비슷한 단어도 없습니다!  
파일명 : 110282newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 3, 횟수 : 1  
파일명 : 110434newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
```

파일명 : 237175newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 237697newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 4
파일명 : 237953newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 10
파일명 : 238090newsML.txt에는 단어가 없습니다!, 일치하는 단어 수 : 2, 횟수 : 10

파일명 236470newsML.txt에는 단어와 글자가 2개 일치하는 것이 4번 있습니다.
파일명 237697newsML.txt에는 단어와 글자가 2개 일치하는 것이 4번 있습니다.
파일명 237953newsML.txt에는 단어와 글자가 2개 일치하는 것이 10번 있습니다.
파일명 238090newsML.txt에는 단어와 글자가 2개 일치하는 것이 10번 있습니다.

British property group MEPC on Thursday said he was "We feel very confident about the whole of the profit statement. The only slight dis-
turbance is that the NAV has dropped slightly," said Tuckey. He pointed out the valuation date for the portfolio was August 31 and there had been
a 14 percent increase in pretax profits to 140 million pounds. In the rise in pretax profits to 140 million pounds, Tuckey said the NAV was below
the target level of 690 pence per share for 2001. "We are convinced this is a temporary write-down because the centre is still getting
a lot of business from the rest of the group," he said. And he defended MEPC's decision to give a target NAV of 690 pence per share for 2001.
"We're sticking our necks out...The portfolio is very different to what it was last year," he said. Tuckey said the full impact of the change of strategy in MEPC's portfolio would take
a year to play out.

C:\WINDOWS\system32\cmd.exe

www.Bandicam.co.kr

1000

1000

원하는 단어를 입력하시오. :

새로운 구조체에 넣은 뒤

파일명 118687newsML.txt에는 단어가 1번 있습니다.
파일명 121957newsML.txt에는 단어가 2번 있습니다.
파일명 123461newsML.txt에는 단어가 1번 있습니다.
파일명 123517newsML.txt에는 단어가 1번 있습니다.
파일명 125313newsML.txt에는 단어가 3번 있습니다.
파일명 126597newsML.txt에는 단어가 1번 있습니다.
파일명 176224newsML.txt에는 단어가 1번 있습니다.
파일명 183684newsML.txt에는 단어가 1번 있습니다.
파일명 19345newsML.txt에는 단어가 1번 있습니다.
파일명 206567newsML.txt에는 단어가 1번 있습니다.
파일명 217358newsML.txt에는 단어가 1번 있습니다.
파일명 242319newsML.txt에는 단어가 1번 있습니다.
파일명 242519newsML.txt에는 단어가 1번 있습니다.
파일명 27007newsML.txt에는 단어가 1번 있습니다.
파일명 364638newsML.txt에는 단어가 2번 있습니다.
파일명 123004newsML.txt에는 단어가 1번 포함되어 있습니다.
파일명 147528newsML.txt에는 단어가 1번 포함되어 있습니다.
파일명 21567newsML.txt에는 단어가 2번 포함되어 있습니다.
파일명 101797newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 102444newsML.txt에는 단어와 글자가 2개 일치하는 것이 4번 있습니다.
파일명 10306newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 103904newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.

} -> korea 입력 시 18번
나오는 것을 확인 가능!



<랭킹 매기기>

```
for (k = 0; k < filecount - 1; k++)
{
    for (h = 0; h < filecount - k - 1; h++)
    {
        if ((have+h)->yescount < (have + h+1)->yescount)
        {
            temp1 = have[h];
            have[h] = have[h + 1];
            have[h + 1] = temp1;
        }
        else if (!(have[h].yescount > 0))
            break;
    }
}
```

```
for (i = 0; i < filecount; i++)
{
    if ((in+i)->contain > 0)
    {
        for (k = 0; k < filecount - 1; k++)
        {
            for (h = 0; h < filecount - k - 1; h++)
            {
                if ((in + h)->contain < (in + h+1)->co
                {
                    temp2 = in[h];
                    in[h] = in[h + 1];
                    in[h + 1] = temp2;
                }
            }
        }
    }
}
```

```
for (i = 0; i < filecount; i++)
{
    if ((coin+i)->coincount > 0)
    {
        for (k = 0; k < filecount - 1; k++)
        {
            for (h = 0; h < filecount - k - 1; h++)
            {
                if ((coin+h)->coin < (coin + h+1)->coin)
                {
                    temp3 = coin[h];
                    coin[h] = coin[h + 1];
                    coin[h + 1] = temp3;
                }
                else if ((coin + h)->coin == (coin + h + 1)->coin)
                {
                    if ((coin + h)->coincount < (coin + h+1)->coincount)
                    {
                        temp3 = coin[h];
                        coin[h] = coin[h + 1];
                        coin[h + 1] = temp3;
                    }
                }
            }
        }
    }
}
```

-> 새로운 구조체에 저장한 각각의 값을 bubble sort를 이용하여 오름차순으로 재 정렬한다.

```

rank = 1;
printf("-----단어 있는 경우-----\n");
for (i = 0; i < filecount; i++)
{
    if ((have + i)->yescount > 0)
    {
        printf(" [순위%d] - %s, %d번\n", rank, (have + i)->yesname, (have+i)->yescount);
        rank++;
    }
}

printf("-----단어는 없지만 포함하는 경우-----\n");
for (i = 0; i < filecount; i++)
{
    if ((in + i)->contain > 0)
    {
        printf(" [순위%d] - %s, %d번\n", rank, (in + i)->inname, (in+i)->contain);
        rank++;
    }
}

```

→ 각 정렬한 값을 순서대로 출력한다.

```
printf("-----단어가 없지만 유사한 단어는 있는 경우-----\n");
for (i = 0; i < filecount; i++)
{
    if (((coin + i)->coincount > 0)
    {
        printf(" [순위%d] - %s %d글자 %d번\n", rank, (coin + i)->coinnname, (coin+i)->coin, (coin+i)->coincount);
        rank++;
    }
}

printf("-----포함하거나 유사한 단어조차도 없는 경우-----\n");
for (i = 0; i < filecount; i++)
{
    if (!((score+i)->yescount > 0) && !((score+i)->contain > 0))
    {
        if (!((score + i)->nocount > 0))
        {
            printf(" [순위%d] - %s\n", rank, (score + i)->filename);
            rank++;
        }
    }
}
```

<Test 1>



파일명 : 138497newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 139110newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 139624newsML.txt에는 단어가 없습니다!, 일치하는 단어
 파일명 : 139755newsML.txt에는 단어가 없습니다!, 일치하는 단어
 파일명 : 140051newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 14014newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 140355newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 140372newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 140545newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 141391newsML.txt에는 단어가 없습니다!, 일치하는 단어
 파일명 : 141413newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 141414newsML.txt에는 단어가 없습니다!, 일치하는 단어
 파일명 : 141943newsML.txt에는 단어가 없습니다!, 일치하는 단어
 파일명 : 142116newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 142460newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
 파일명 : 142624newsML.txt에는 단어가 없습니다!, 일치하는 단어
 파일명 : 142661newsML.txt에는 단어가 있습니다! 있는 개수 : 1

- [순위0] - 115389newsML.txt, 1번
- [순위7] - 142661newsML.txt, 1번**
- [순위8] - 143081newsML.txt, 1번
- [순위9] - 144035newsML.txt, 1번
- [순위10] - 163094newsML.txt, 1번
- [순위11] - 163409newsML.txt, 1번
- [순위12] - 186199newsML.txt, 1번
- [순위13] - 188988newsML.txt, 1번
- [순위14] - 192053newsML.txt, 1번
- [순위15] - 193495newsML.txt, 1번
- [순위16] - 197734newsML.txt, 1번
- [순위17] - 198232newsML.txt, 1번
- [순위18] - 236594newsML.txt, 1번
- [순위19] - 239149newsML.txt, 1번
- [순위20] - 263451newsML.txt, 1번
- [순위21] - 263741newsML.txt, 1번
- [순위22] - 309597newsML.txt, 1번

-----단어는 없지만 포함하는 경우-----

수 : 2 횟수 : 2

새로운 구조체에 넣은 뒤

파일명 101886newsML.txt에는 단어가 1번 있습니다.
 파일명 102211newsML.txt에는 단어가 1번 있습니다.
 파일명 104067newsML.txt에는 단어가 1번 있습니다.
 파일명 115389newsML.txt에는 단어가 1번 있습니다.
 파일명 120683newsML.txt에는 단어가 3번 있습니다.
 파일명 125085newsML.txt에는 단어가 2번 있습니다.
파일명 142661newsML.txt에는 단어가 1번 있습니다.
 파일명 143081newsML.txt에는 단어가 1번 있습니다.
 파일명 144035newsML.txt에는 단어가 1번 있습니다.
 파일명 163094newsML.txt에는 단어가 1번 있습니다.
 파일명 163409newsML.txt에는 단어가 1번 있습니다.
 파일명 186199newsML.txt에는 단어가 1번 있습니다.
 파일명 188988newsML.txt에는 단어가 1번 있습니다.
 파일명 192053newsML.txt에는 단어가 1번 있습니다.
 파일명 193495newsML.txt에는 단어가 1번 있습니다.
 파일명 197734newsML.txt에는 단어가 1번 있습니다.
 파일명 198232newsML.txt에는 단어가 1번 있습니다.
 파일명 236594newsML.txt에는 단어가 1번 있습니다.
 파일명 239149newsML.txt에는 단어가 1번 있습니다.
 파일명 263451newsML.txt에는 단어가 1번 있습니다.
 파일명 263741newsML.txt에는 단어가 1번 있습니다.
 파일명 309597newsML.txt에는 단어가 1번 있습니다.
 파일명 136300newsML.txt에는 단어가 1번 포함되어 있습니다.
 파일명 136958newsML.txt에는 단어가 1번 포함되어 있습니다.
 파일명 137498newsML.txt에는 단어가 1번 포함되어 있습니다.
 파일명 138243newsML.txt에는 단어가 1번 포함되어 있습니다.
 파일명 270046newsML.txt에는 단어가 1번 포함되어 있습니다.
 파일명 270084newsML.txt에는 단어가 1번 포함되어 있습니다.

The St. Louis-based airline said on Thursday that Chief Executive Officer Jeffrey Erickson's departure comes just four months after Chief Financial Officer Robert Pei Erickson, who became chief executive officer in August 1994, brought the airline through "Erickson was the one who really steered them through the first half of the turnaround." The company had a \$14.3 million third-quarter loss and said its yield "I do TWA's "We Indus □ 대/소문자 구분(C)"Whatever they do to replace Erickson, they need a hands-on manager, not a messiah," Boyd said several managers at United Airlines might be interested in the job. He also earlier this year, Erickson, who was criticized by families of the Flight 800 victim "I can't help believing the crash just took a lot out of Jeff," said Greenbaum, who

찾기

찾을 내용

다음 찾기(F)

취소

방향

 위로(U) 아래로(D)

C:\ 선택 C:\WINDOWS\system32\cmd.exe

파일명 : 27668newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 2
파일명 : 277010newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 3, 횟수 : 1
파일명 : 277117newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 277513newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 1
파일명 : 277885newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 4, 횟수 : 4
파일명 : 279102newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 2
파일명 : 279442newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 279474newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 4, 횟수 : 3
파일명 : 279477newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 2
파일명 : 280467newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 280485newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 2
파일명 : 280504newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 2
파일명 : 281200newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 281220newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 281893newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 28301newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 1
파일명 : 28462newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 3, 횟수 : 2
파일명 : 284741newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 28480newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 285403newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 3, 횟수 : 1
파일명 : 288069newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 288397newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 289108newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 3, 횟수 : 1
파일명 : 289165newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 2
파일명 : 28928newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 3, 횟수 : 3
파일명 : 289578newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 1
파일명 : 290125newsML.txt에는 그 어떤 비슷한 단어도 없습니다!
파일명 : 290471newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 1
파일명 : 29051newsML.txt에는 단어가 없습니다!. 일치하는 단어 수 : 2, 횟수 : 1
파일명 : 290521newsML.txt에는 그 어떤 비슷한 단어도 없습니다!

[순위27] - 270046newsML.txt, 1번
[순위28] - 270084newsML.txt, 1번

-----단어가 없지만 유사한 단어는 있는 경우-----

[순위29] - 277885newsML.txt 4글자 4번

[순위30] - 279474newsML.txt 4글자 3번
[순위31] - 233150newsML.txt 4글자 1번
[순위32] - 216670newsML.txt 3글자 8번
[순위33] - 109096newsML.txt 3글자 6번
[순위34] - 124585newsML.txt 3글자 5번
[순위35] - 272513newsML.txt 3글자 5번
[순위36] - 123170newsML.txt 3글자 4번
[순위37] - 207570newsML.txt 3글자 4번
[순위38] - 102529newsML.txt 3글자 3번
[순위39] - 107525newsML.txt 3글자 3번
[순위40] - 146870newsML.txt 3글자 3번
[순위41] - 226801newsML.txt 3글자 3번
[순위42] - 28928newsML.txt 3글자 3번
[순위43] - 105834newsML.txt 3글자 2번
[순위44] - 106247newsML.txt 3글자 2번
[순위45] - 123524newsML.txt 3글자 2번

C:\ 선택 C:\WINDOWS\system32\cmd.exe

파일명 276094newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.
파일명 27668newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.
파일명 277010newsML.txt에는 단어와 글자가 3개 일치하는 것이 1번 있습니다.
파일명 277513newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 277885newsML.txt에는 단어와 글자가 4개 일치하는 것이 4번 있습니다.
파일명 279102newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.
파일명 279474newsML.txt에는 단어와 글자가 4개 일치하는 것이 3번 있습니다.
파일명 279477newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.
파일명 280485newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.
파일명 280504newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.
파일명 28301newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 28462newsML.txt에는 단어와 글자가 3개 일치하는 것이 2번 있습니다.
파일명 285403newsML.txt에는 단어와 글자가 3개 일치하는 것이 1번 있습니다.
파일명 289108newsML.txt에는 단어와 글자가 3개 일치하는 것이 1번 있습니다.
파일명 289165newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.
파일명 28928newsML.txt에는 단어와 글자가 3개 일치하는 것이 3번 있습니다.
파일명 289578newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 290471newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 29051newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 29098newsML.txt에는 단어와 글자가 3개 일치하는 것이 1번 있습니다.
파일명 291066newsML.txt에는 단어와 글자가 3개 일치하는 것이 1번 있습니다.
파일명 293044newsML.txt에는 단어와 글자가 3개 일치하는 것이 1번 있습니다.
파일명 294330newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 294343newsML.txt에는 단어와 글자가 3개 일치하는 것이 1번 있습니다.
파일명 294823newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 297151newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.
파일명 297895newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.
파일명 299375newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 299957newsML.txt에는 단어와 글자가 2개 일치하는 것이 1번 있습니다.
파일명 300439newsML.txt에는 단어와 글자가 2개 일치하는 것이 2번 있습니다.

China has jailed a Tibetan music expert touring the Himalayan region on a U.S. scholarship, Ngawang Choephel, 30, who was travelling in Tibet as a Fulbright scholar to produce a study. The young scholar confessed to having been sent to Tibet by "the Dalai (Lama) clique." U.S. officials had no immediate comment on the case.

He was visiting "under the pretext of collecting information on Tibetan folk songs and culture," In accordance with the Criminal Law of the People's Republic of China... The Internationl Commission on Human Rights has condemned the sentence, which carries a prison term of four years.

China has condemned the sentence, which carries a prison term of four years. The sentence is about anyone with possible links to the exiled Dalai Lama. "It is the most severe handed down in a political case since 1950," Robbie Barnett of the London-based International Commission on Human Rights said. "It does seem to be a provocative decision in political terms because it names three people."

"It appears to mean the end of any sense that China is susceptible to the international community," Barnett said. Ngawang Choephel left Tibet as a child shortly after the Chinese takeover in 1950 and grew up in India. The young man, who once worked as a dance teacher in India, travelled to Tibet in 1982 to gain permission to travel to his birthplace. Ngawang Choephel renounced his Indian citizenship and became a citizen of the People's Republic of China. He was last reported to have been seen in Nyari prison in Xigaze, Tibet's second largest city. "It is very unlikely that he was a spy," said Barnett. "He was genuinely making the trip to collect information on Tibetan folk songs and culture."

찾기

찾을 내용

다음 찾기(F)

취소

방향

 위로(U) 아래로(D) 대/소문자 구분(C)

<완성 코드>

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <iostream.h>
4  #include <conio.h>
5  #include <stdlib.h>
6
7  //파일 경로 초 3군데 지정해줘야 함!
8
9  typedef struct hello
10 {
11     char filename[30]; //해당 파일의 이름
12     int yescount; //단어가 있는 경우 몇번 있는지
13     int contain; //단어를 몇번 포함하는지
14     int coin; //몇 글자가 일치하는지
15     int nocount; //일치하는 경우 몇번 있는지
16 }hello;
17
18  typedef struct yes //단어 있는 경우
19 {
20     char yesname[30];
21     int yescount;
22 }Yes;
23
```

```
23
24     typedef struct include //없는데 포함하는 경우
25     {
26         char innname[30];
27         int contain;
28     } In;
29
30     typedef struct coincide //없는데 앞글자 일치 정도(유사도)
31     {
32         char coinname[30];
33         int coin;
34         int coincount;
35     } Co;
36
37     int filecounter();
38     void lastchecker(char st[100]);
39     void frontchecker(char strr[100]);
```

```
40
41     int main()
42     {
43         char str[100], word[100], b1[100], b2[100];
44         int count = 0, size, i, num = 0, j, m, n, k, h;
45         int include = 0; // 사용자가 입력한 단어가 파일의 단어에 포함되는 경우
46         int noword = 0; // 사용자가 입력한 단어가 파일의 단어에 앞부분에 일부 있을 때
47         int nowordmax = 2; // 위의 숫자의 경우 중 제일 큰 경우(글자수가 가장 많이 일치할 경우)
48         int nowordmaxc = 0;
49         struct _finddata_t fd;
50         long handle;
51         int result = 1;
52         int filecount = 0;
53         char filename[30];
54         char ChangeDir[60] = { "c:\\dataset-2nd\\***" }; // 디렉토리 바꿔줄꺼임, 꼭 지정
55         int rank;
56
57         bello *score;
58         Yes *have;
59         In *in;
60         Co *coin;
```

```
61
62     Yes temp1;
63     In temp2;
64     Co temp3;
65
66
67     if ((score = (belIo*)malloc(sizeof(belIo)*filecounter())) == NULL)
68     {
69         printf("score에 자리없음!");
70         exit(1);
71     }
72
73     if ((have = (Yes*)malloc(sizeof(Yes)*filecounter())) == NULL)
74     {
75         printf("have에 자리없음!");
76         exit(1);
77     }
78
79     if ((in = (In*)malloc(sizeof(In)*filecounter())) == NULL)
80     {
81         printf("in에 자리없음!");
82         exit(1);
83     }
```

```
85     if ((coin = (Co*)malloc(sizeof(Co)*filecounter())) == NULL)
86     {
87         printf("coin에 자리없음!");
88         exit(1);
89     }
90
91     handle = _findfirst("c:\dataset-2nd\*.*", &fd); //파일 경로 꼭 지정!
92     if (handle == -1) return;
93     while (result != -1)
94     {
95         if ((strcmp(".", fd.name)) != 0 && (strcmp("../", fd.name)) != 0)
96         {
97             strcpy(score[filecount].filename, fd.name);
98             filecount++;
99         }
100        result = _findnext(handle, &fd);
101    }
102    _findclose(handle);
103
104    printf("원하는 단어를 입력하시오. : ");
105    scanf("%s", str);
106
107    size = strlen(str);
```

```
09     if (96 < str[0] && str[0] < 123) //str이 소문자로만 구성될 경우 -> 대문자가 포함 된 경우에도 검색 가능해야 하므로.
10     {
11         strcpy(b1, str);
12         b1[0] = str[0] - 32; //str의 제일 첫번째 글자를 대문자로 바꿔준다.
13         strcpy(b2, str);
14        strupr(b2); //str을 전부 대문자로 바꿔준다.
15     }
16
17     else if (64 < str[0] && str[0] < 91) //첫 글자가 대문자일때
18     {
19         if (64 < str[1] && str[1] < 91) // 두번째 글자도 대문자인경우
20         {
21             strcpy(b1, str);
22             strlwr(b1);
23             b1[0] = b1[0] - 32; //str을 첫번째 문자를 제외하고 전부 소문자로 바꿔준다.
24             strcpy(b2, str);
25             strlwr(b2); //str을 전부 소문자로 바꿔준다.
26     }
```

```
27
28     else if (96 < str[1] && str[1] < 123) // 두번째 글자가 소문자인 경우
29     {
30         strcpy(b1, str);
31         b1[0] = str[0] + 32; //첫번째 문자를 소문자로 바꿔준다.
32         strcpy(b2, str);
33        strupr(b2); //str을 전부 대문자로 바꿔준다.
34     }
35 }
36
37 if (chdir(ChangeDir) == 0) //작업 디렉토리 바꿔줌
38 {
39     for (i = 0; i < filecount; i++)
40     {
41         count = 0;
42         include = 0;
43         nowordmax = 2;
44         nowordmaxc = 0;
45         FILE *fp = fopen(score[i].filename, "rt");
46
47         if (fp == NULL)
48         {
49             printf("이 파일은 비어있습니다!");
50             return 1;
```

```
152     do
153     {
154         fscanf(fp, "%s", word);
155         frontchecker(word);
156         lastchecker(word);
157         frontchecker(word);
158         lastchecker(word);
159
160         if (strlen(word) == size) // strstr에서 at을 입력했는데 that도 at으로 인식하는 경우를 막기 위해.
161         {
162             if (strstr(word, str) != NULL) // 입력한 단어와 일치하는 경우
163                 count++;
164             else if (strstr(word, b1) != NULL) // 입력한 단어가 소문자일때 대문자로 제일 첫 글자를 바꾼 경우
165                 count++;
166             else if (strstr(word, b2) != NULL) // 입력한 단어가 소문자일때 모두 대문자로 바꾼 경우
167                 count++;
168         }
169         else
170             continue;
171     } while (!feof(fp));
172 }
```

```
175 if ((score + i)->yescount == 0) //단어가 없을 경우
176 {
177     fseek(fp, 0L, SEEK_SET); //파일 제일 앞으로 이동.
178     do
179     {
180         fscanf(fp, "%s", word);
181         frontchecker(word);
182         lastchecker(word);
183         frontchecker(word);
184         lastchecker(word);
185
186     if (size < strlen(word)) // 파일에서 가져온 단어에 사용자가 입력한 단어가 포함되는 경우
187     {
188         if (strstr(word, str) != NULL) // 입력한 단어와 일치하는 경우
189             include++;
190         else if (strstr(word, b1) != NULL) //입력한 단어가 소문자일때 대문자로 제일 첫 글자를 바꾼 경우
191             include++;
192         else if (strstr(word, b2) != NULL) // 입력한 단어가 소문자일때 모두 대문자로 바꾼 경우
193             include++; //포함된 횟수 세는거임.
194     }
195     else
196         continue;
197     } while (!feof(fp));
198     (score + i)->contain = include;
199 }
```

```
199
200
201     }
202
203     if ((score + i)->yescount == 0 && (score + i)->contain == 0)
204     {
205         fseek(fp, 0L, SEEK_SET); //파일 제일 앞으로 이동.
206         do
207         {
208             fscanf(fp, "%s", word);
209             frontchecker(word);
210             lastchecker(word);
211             frontchecker(word);
212             lastchecker(word);
213             noword = 0;
214
215             if (strncmp(str, word, 2) == 0)
216             {
217                 noword = 2; //이미 2개 일치하므로
218                 for (num = 2; num < size; num++)
219                 {
220                     if (str[num] == word[num])
221                         noword++;
222                     else
223                         break;
224                 }
225             }
226         }
```

```
223     }
224
225     else if (strncmp(b1, word, 2) == 0)
226     {
227         noword = 2; //이미 2개 일치하므로
228         for (num = 2; num < size; num++)
229         {
230             if (str[num] == word[num])
231                 noword++;
232             else
233                 break;
234         }
235     }
236
237     else if ((strncmp(b2, word, 2)) == 0)
238     {
239         noword = 2; //이미 2개 일치하므로
240         for (num = 2; num < size; num++)
241         {
242             if (str[num] == word[num])
243                 noword++;
244             else
245                 break;
246         }
247     }
```

```
248
249     if (noword > nowordmax)
250     {
251         nowordmax = noword;
252         nowordmaxc = 1; //max값이 바꼈으니까
253     }
254     else if (noword == nowordmax)
255     {
256         nowordmaxc++; //같은 경우가 많을 수록 앞에
257     }
258
259 } while (!feof(fp));
260 (score + i)->coin = nowordmax;
261 (score + i)->nocount = nowordmaxc;
262 }
263 fclose(fp);
264 }
265 }
```

```
62
63     j = 0, n = 0, m = 0;
64     for (i = 0; i < filecount; i++)
65     {
66         if ((score + i)->yescount > 0)
67         {
68             strcpy((have + j)->yesname, (score + i)->filename);
69             (have + j)->yescount = (score + i)->yescount;
70             j++;
71         }
72
73         if ((score + i)->contain > 0)
74         {
75             strcpy(in[n].inname, (score + i)->filename);
76             (in + n)->contain = (score + i)->contain;
77             n++;
78         }
79
80         if ((score + i)->nocount > 0)
81         {
82             strcpy((coin + m)->coinname, (score + i)->filename);
83             (coin + m)->coin = (score + i)->coin;
84             (coin + m)->coincount = (score + i)->nocount;
85             m++;
86         }
87     }
```

```
88
89     for (k = 0; k < filecount - 1; k++)
90     {
91         for (h = 0; h < filecount - k - 1; h++)
92         {
93             if ((have + h)->yescount < (have + h + 1)->yescount)
94             {
95                 temp1 = have[h];
96                 have[h] = have[h + 1];
97                 have[h + 1] = temp1;
98             }
99             else if (!(have[h].yescount > 0))
100                 break;
101         }
102     }
103 }
```

```
604  
605     for (i = 0; i < filecount; i++)  
606     {  
607         if ((in + i)->contain > 0)  
608         {  
609             for (k = 0; k < filecount - 1; k++)  
610             {  
611                 for (h = 0; h < filecount - k - 1; h++)  
612                 {  
613                     if ((in + h)->contain < (in + h + 1)->contain)  
614                     {  
615                         temp2 = in[h];  
616                         in[h] = in[h + 1];  
617                         in[h + 1] = temp2;  
618                     }  
619                 }  
620             }  
621         }  
622     }  
623 }
```

```
22 }
23
24     for (i = 0; i < filecount; i++)
25     {
26         if ((coin + i)->coincount > 0)
27         {
28             for (k = 0; k < filecount - 1; k++)
29             {
30                 for (h = 0; h < filecount - k - 1; h++)
31                 {
32                     if ((coin + h)->coin < (coin + h + 1)->coin)
33                     {
34                         temp3 = coin[h];
35                         coin[h] = coin[h + 1];
36                         coin[h + 1] = temp3;
37                     }
38                 else if ((coin + h)->coin == (coin + h + 1)->coin)
39                 {
40                     if ((coin + h)->coincount < (coin + h + 1)->coincount)
41                     {
42                         temp3 = coin[h];
43                         coin[h] = coin[h + 1];
44                         coin[h + 1] = temp3;
45                     }
```

```
640         if ((coin + h)->coincount < (coin + h + 1)->coincount)
641         {
642             temp3 = coin[h];
643             coin[h] = coin[h + 1];
644             coin[h + 1] = temp3;
645         }
646     }
647 }
648 }
649 }
650 }
651
652 rank = 1;
653 printf("-----단어 있는 경우-----\n");
654 for (i = 0; i < filecount; i++)
655 {
656     if ((have + i)->yescount > 0)
657     {
658         printf(" [순위%d] - %s, %d번\n", rank, (have + i)->yesname, (have + i)->yescount);
659         rank++;
660     }
661 }
```

```
63 printf("-----단어는 없지만 포함하는 경우-----\n");
64 for (i = 0; i < filecount; i++)
65 {
66     if ((in + i)->contain > 0)
67     {
68         printf(" [순위%d] - %s, %d번\n", rank, (in + i)->inname, (in + i)->contain);
69         rank++;
70     }
71 }
72
73 printf("-----단어가 없지만 유사한 단어는 있는 경우-----\n");
74 for (i = 0; i < filecount; i++)
75 {
76     if ((coin + i)->coincount > 0)
77     {
78         printf(" [순위%d] - %s %d글자 %d번\n ", rank, (coin + i)->coinname, (coin + i)->coin, (coin + i)->coincount);
79         rank++;
80     }
81 }
```

```
83 printf("-----포함하거나 유사한 단어조차도 없는 경우----- \n");
84 for (i = 0; i < filecount; i++)
{
85     if (!((score + i)->yescount > 0) && !((score + i)->contain > 0))
86     {
87         if (!((score + i)->nocount > 0))
88         {
89             printf(" [순위%d] - %s\n ", rank, (score + i)->filename);
90             rank++;
91         }
92     }
93 }
94
95
96 return 0;
97
98 }
```

```
97 }
98
99 int filecounter()
100 {
101     struct _finddata_t fd1;
102     long handle;
103     int result = 1;
104     int filecount = 0;
105
106     handle = _findfirst("c:\dataset-2nd\*\.*", &fd1); //파일경로 지정!
107     if (handle == -1) return;
108     while (result != -1)
109     {
110         if ((strcmp(".", fd1.name)) != 0 && (strcmp("..", fd1.name)) != 0)
111         {
112             filecount++;
113         }
114         result = _findnext(handle, &fd1);
115     }
116     _findclose(handle);
117
118     return filecount;
119 }
120 }
```

```
21 void lastchecker(char st[100])
22 {
23     int size; //st의 크기
24     char last;
25
26     size = strlen(st);
27
28     last = st[size - 1];
29
30     switch (last)
31     {
32     case ',':
33         st[size - 1] = '\0';
34         break;
35     case '.':
36         st[size - 1] = '\0';
37         break;
38     case ')':
39         st[size - 1] = '\0';
40         break;
41     case '|':
42         st[size - 1] = '\0';
43         break;
44     case '~':
```

```
39         st[size - 1] = '\0';
40     break;
41 case '|':
42     st[size - 1] = '\0';
43     break;
44 case '~':
45     st[size - 1] = '\0';
46     break;
47 case '?':
48     st[size - 1] = '\0';
49     break;
50 case '#':
51     st[size - 1] = '\0';
52     break;
53 case '^':
54     st[size - 1] = '\0';
55     break;
56 default:
57     break;
58 }
59 }
60
61 void frontchecker(char strr[100])
62 {
```

```
60
61     void frontchecker(char strr[100])
62     {
63         int size, i;
64         char front;
65
66         size = strlen(strr);
67
68         front = strr[0];
69
70         switch (front)
71         {
72             case '(':
73                 for (i = 0; i < size - 1; i++)
74                     strr[i] = strr[i + 1];
75                 strr[size - 1] = '\0';
76                 break;
77             case ')':
78                 for (i = 0; i < size - 1; i++)
79                     strr[i] = strr[i + 1];
80                 strr[size - 1] = '\0';
81                 break;
82             case '[':
83                 for (i = 0; i < size - 1; i++)
```

```
69
70     switch (front)
71     {
72         case '(':
73             for (i = 0; i < size - 1; i++)
74                 strr[i] = strr[i + 1];
75             strr[size - 1] = '\0';
76             break;
77         case ')':
78             for (i = 0; i < size - 1; i++)
79                 strr[i] = strr[i + 1];
80             strr[size - 1] = '\0';
81             break;
82         case '[':
83             for (i = 0; i < size - 1; i++)
84                 strr[i] = strr[i + 1];
85             strr[size - 1] = '\0';
86             break;
87         default:
88             break;
89     }
90 }
```

<Test 1>



C:\ 선택 C:\WINDOWS\system32\cmd.exe

원하는 단어를 입력하시오. : air

-----결과-----

-----단어 있는 경우-----

- [순위1] - 140545newsML.txt, 13번
- [순위2] - 144314newsML.txt, 13번
- [순위3] - 126597newsML.txt, 11번
- [순위4] - 206567newsML.txt, 11번
- [순위5]** - 123461newsML.txt, 9번
- [순위6] - 108091newsML.txt, 6번
- [순위7] - 133504newsML.txt, 6번
- [순위8] - 166432newsML.txt, 6번
- [순위9] - 105222newsML.txt, 5번
- [순위10] - 156695newsML.txt, 5번
- [순위11] - 10245newsML.txt, 4번



선택 C:\WINDOWS\system32\cmd.exe

[순위61]	-	147521newsML.txt	, 1번
[순위62]	-	147557newsML.txt	, 1번
[순위63]	-	148847newsML.txt	, 1번
[순위64]	-	159589newsML.txt	, 1번
[순위65]	-	172811newsML.txt	, 1번
[순위66]	-	174581newsML.txt	, 1번
[순위67]	-	196713newsML.txt	, 1번
[순위68]	-	198061newsML.txt	, 1번
[순위69]	-	199220newsML.txt	, 1번
[순위70]	-	199226newsML.txt	, 1번
[순위71]	-	202020newsML.txt	, 1번
[순위72]	-	203933newsML.txt	, 1번
[순위73]	-	206557newsML.txt	, 1번
[순위74]	-	228103newsML.txt	, 1번
[순위75]	-	243004newsML.txt	, 1번
[순위76]	-	293006newsML.txt	, 1번
[순위77]	-	293216newsML.txt	, 1번
[순위78]	-	31892newsML.txt	, 1번
[순위79]	-	33299newsML.txt	, 1번
		단어는 없지만 포함하는	
[순위80]	-	110697newsML.txt	, 12번
[순위81]	-	111981newsML.txt	, 12번
[순위82]	-	111971newsML.txt	, 8번
[순위83]	-	138342newsML.txt	, 8번
[순위84]	-	185661newsML.txt	, 7번
[순위85]	-	224980newsML.txt	, 7번
[순위86]	-	196127newsML.txt	, 6번
[순위87]	-	126345newsML.txt	, 5번
[순위88]	-	219521newsML.txt	, 5번
[순위89]	-	219645newsML.txt	, 5번

Leading Chinese dissident Chen Ziming has been placed under tight police surveillance since his release from prison last month. "Some conditions (after his parole) are not as good as when (he) was in prison," his sister said. "When he was in prison he could still go downstairs for some air," she said. Chen Ziming has been under tight police surveillance since he was released from prison last month. Asked about Chen's plans, the sister said he did not talk about anything. The dissident needs police permission to see a doctor, the sister said. He is being monitored by police and reporters. Uniformed and plainclothes officers are keeping Chen under house arrest, Chen's sister said: "(I'm) afraid to let him out of the house because he may bring trouble to the family." We are happy to see Chen Ziming again. Of course, (we) are happy (he) came back home. Chen was sentenced to 15 years in prison in 1991 for engaging in "counterrevolutionary activities." Chen had been singled out by Beijing authorities as the "black hand" behind Wang's imprisonment. He won medical parole in 1994 in a move widely seen as part of a success story. Chen was returned to prison on June 25, 1995, after joining calls for Beijing's release. His release on Wednesday came seven days after a Beijing court took less than two hours to rule that Chen's 15-year sentence was unconstitutional. Wang's imprisonment was the latest in a series of crushing blows to China's dissident community.

[순위70] - 199226newsML.txt, 1번
 [순위71] - 202020newsML.txt, 1번
 [순위72] - 203933newsML.txt, 1번
 [순위73] - 206557newsML.txt, 1번
 [순위74] - 228103newsML.txt, 1번
 [순위75] - 243004newsML.txt, 1번
 [순위76] - 293006newsML.txt, 1번
 [순위77] - 293216newsML.txt, 1번
 [순위78] - 31892newsML.txt, 1번
 [순위79] - 33299newsML.txt, 1번

-----단어는 없지만 포함하는 경우-----

[순위80] - 110697newsML.txt, 12번
 [순위81] - 111981newsML.txt, 12번
 [순위82] - 111971newsML.txt, 8번
 [순위83] - 138342newsML.txt, 8번
 [순위84] - 185661newsML.txt, 7번
 [순위85] - 224980newsML.txt, 7번
 [순위86] - 196127newsML.txt, 6번
 [순위87] - 126345newsML.txt, 5번
 [순위88] - 219521newsML.txt, 5번
 [순위89] - 219645newsML.txt, 5번
 [순위90] - 222129newsML.txt, 5번
 [순위91] - 348501newsML.txt, 5번
 [순위92] - 18320newsML.txt, 4번
 [순위93] - 18519newsML.txt, 4번
 [순위94] - 185611newsML.txt, 4번
 [순위95] - 216670newsML.txt, 4번
 [순위96] - 24300newsML.txt, 4번
 [순위97] - 26976newsML.txt, 4번
 [순위98] - 27569newsML.txt, 4번

www.Bandicam.co.kr

Rupert Murdoch predicted he would win his legal battle with Ted Turner and News Corp. in Australia.

In remarks published in News Corp.'s Courier Mail newspaper in Brisbane, Murdoch said on Thursday that Murdoch's lawsuit to block the merger of Turner and News Corp. will continue.

"We've got a good stoush (fight) there," Murdoch said. "We'll win through."

Turner said on Thursday that Murdoch's lawsuit to block the merger of Murdoch and News Corp. will continue.

Murdoch has been involved in a pitched battle in recent months over the damages from Time Warner and Turner, which have 9 million subscribers and when they merged last week for comparing Murdoch to Adolf Hitler.

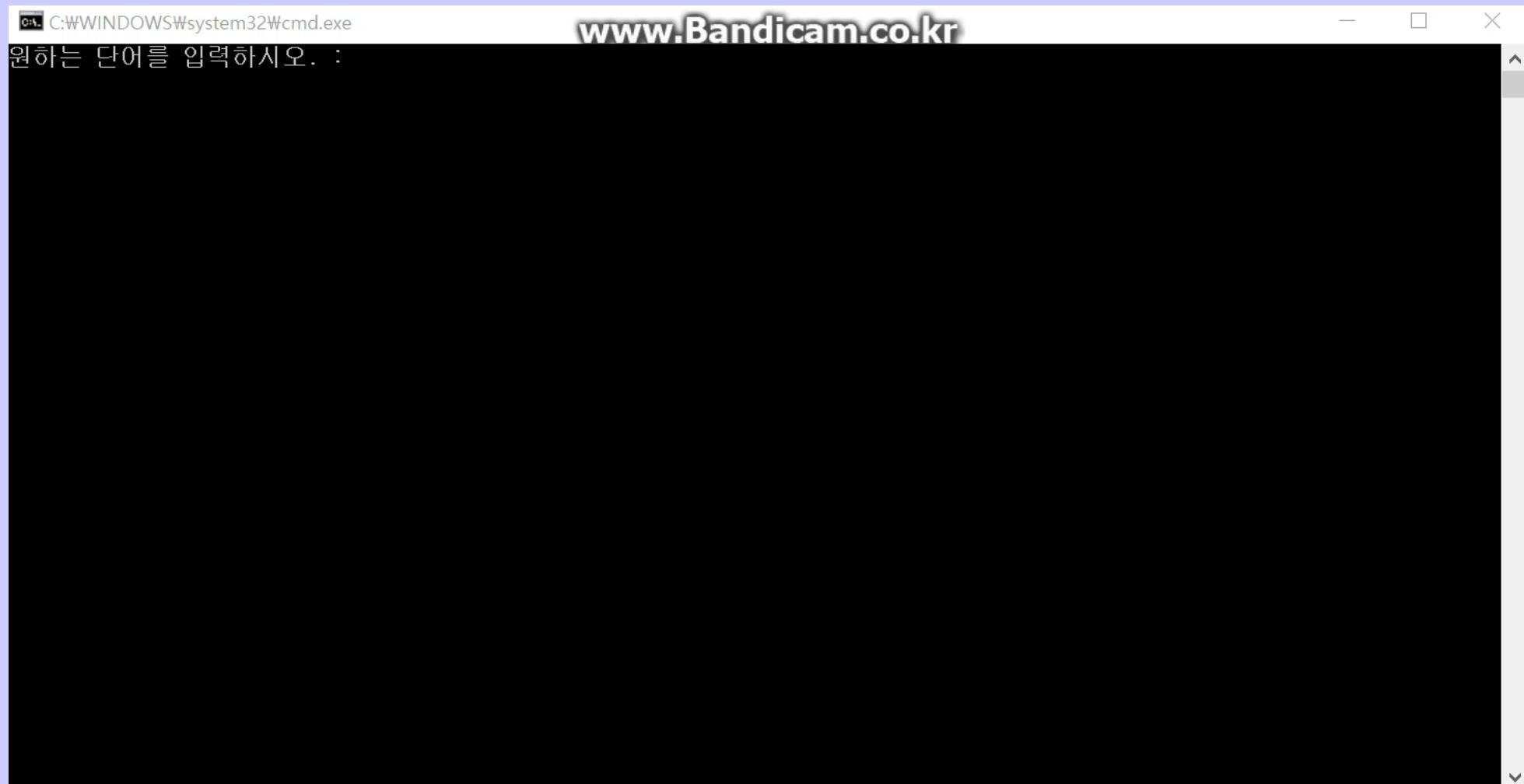
Murdoch's outlook for News Corp.'s profits.

"I don't know if the American economy holds where it is and the British one does, yes," Murdoch said after releasing its results in late August that a 20 percent stake in Murdoch's company.

Murdoch also said he had no plans to buy out John Fairfax Holdings, which He said Australian-based rival media owner Kerry Packer was also unlikely to sell.

"I don't want to buy Fairfax. I don't believe Mr. (Kerry) Packer wants to sell," Murdoch said.

<Test 2>



선택 C:\WINDOWS\system32\cmd.exe

원하는 단어를 입력하시오. : Korea

--결과-----

-----단어 있는 경우-----

- | | | |
|--------|---------------------|----|
| [순위1] | - 125313newsML.txt, | 3번 |
| [순위2] | - 121957newsML.txt, | 2번 |
| [순위3] | - 364638newsML.txt, | 2번 |
| [순위4] | - 118687newsML.txt, | 1번 |
| [순위5] | - 123461newsML.txt, | 1번 |
| [순위6] | - 123517newsML.txt, | 1번 |
| [순위7] | - 126597newsML.txt, | 1번 |
| [순위8] | - 176224newsML.txt, | 1번 |
| [순위9] | - 183684newsML.txt, | 1번 |
| [순위10] | - 19345newsML.txt, | 1번 |
| [순위11] | - 206567newsML.txt, | 1번 |
| [순위12] | - 217358newsML.txt, | 1번 |
| [순위13] | - 242319newsML.txt, | 1번 |
| [순위14] | - 242519newsML.txt, | 1번 |
| [순위15] | - 27007newsML.txt, | 1번 |

[문서18] *greetnewone.txt*, 그는 단어는 없지만 포함하는 The

- [순위16] - 21567newsML.txt, 2번
[순위17] - 123004newsML.txt, 1번
[순위18] - 147528newsML.txt, 1번
-----단어가 없지만 유사한
[순위19] - 128844newsML.txt 3글
[순위20] - 239149newsML.txt 3글
[순위21] - 128405newsML.txt 3글
[순위22] - 236594newsML.txt 3글

"Generally speaking relations are on the upturn," said one diplomat. A row over human rights scarred Christopher's last visit to China in 1999. The visit would mark the highest point in relations since ties plunged after the 1989 Tiananmen Square massacre. In Washington, State Department spokesman Nicholas Burns said Lord had raised human rights "quite importantly" during his visit.

www.Bandicam.co.kr

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

Korean Air Lines Ltd's cargo load factor dropped 3.2 percent in the six months to end-June. The cumulative cargo tonnage carried to end-June was 448,000 tonnes, a rise of 10.3 percent from 5 million tonnes in June last year. International cargo in June was up 11.1 percent on last year to 60,000 tonnes. Domestic cargo in the month reached 16,000 tonnes, an increase of 6.7 percent on the month. Load factors for the month were 77.7 percent for international cargo in June compared to 77.1 percent in May. The cumulative totals are down 0.1 percent on the month compared to last year. Cargo kilometres (FTK) totals for June were up 10.3 percent from 5 million to 6 million. There was a 1.1 percent increase in the number of passengers carried.

선택 C:\WINDOWS\system32\cmd.exe

원하는 단어를 입력하시오. : Korea

결과

-----단어 있는 경우-----

- [순위1] - 125313newsML.txt, 3번
- [순위2] - 121957newsML.txt, 2번
- [순위3] - 364638newsML.txt, 2번
- [순위4] - 118687newsML.txt, 1번
- [순위5] - 123461newsML.txt, 1번
- [순위6] - 123517newsML.txt, 1번
- [순위7] - 126597newsML.txt, 1번
- [순위8] - 176224newsML.txt, 1번
- [순위9] - 183684newsML.txt, 1번
- [순위10] - 19345newsML.txt, 1번
- [순위11] - 206567newsML.txt, 1번
- [순위12] - 217358newsML.txt, 1번
- [순위13] - 242319newsML.txt, 1번
- [순위14] - 242519newsML.txt, 1번
- [순위15] - 27007newsML.txt, 1번

-----단어는 없지만 포함하는 경우-----

- [순위16] - 21567newsML.txt, 2번
- [순위17] - 123004newsML.txt, 1번
- [순위18] - 147528newsML.txt, 1번

-----단어가 없지만 유사한 단어는 있는 경우-----

- [순위19] - 128844newsML.txt 3글자 6번
- [순위20] - 239149newsML.txt 3글자 5번
- [순위21] - 128405newsML.txt 3글자 4번
- [순위22] - 236594newsML.txt 3글자 4번



선택 C:\WINDOWS\system32\cmd.exe

- [순위1] - 125313newsML.txt, 3번
- [순위2] - 121957newsML.txt, 2번
- [순위3] - 364638newsML.txt, 2번
- [순위4] - 118687newsML.txt, 1번
- [순위5] - 123461newsML.txt, 1번
- [순위6] - 123517newsML.txt, 1번
- [순위7] - 126597newsML.txt, 1번
- [순위8] - 176224newsML.txt, 1번
- [순위9] - 183684newsML.txt, 1번
- [순위10] - 19345newsML.txt, 1번
- [순위11] - 206567newsML.txt, 1번
- [순위12] - 217358newsML.txt, 1번
- [순위13] - 242319newsML.txt, 1번
- [순위14] - 242519newsML.txt, 1번
- [순위15] - 27007newsML.txt, 1번

-----단어는 없지만 포함하는 경우-----

- [순위16] - 21567newsML.txt, 2번
- [순위17] - 123004newsML.txt, 1번
- [순위18] - 147528newsML.txt, 1번

-----단어가 없지만 유사한 단어는 있는 경우-----

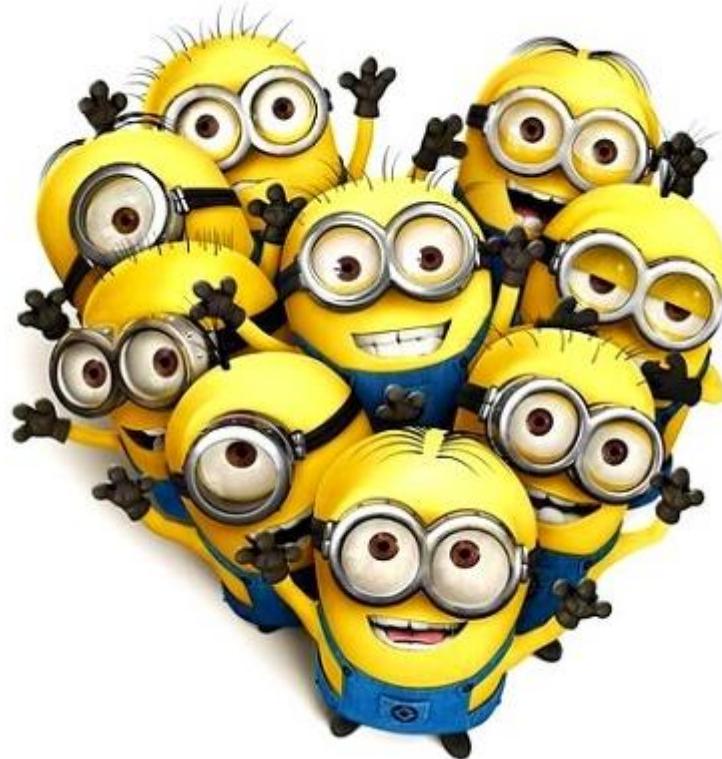
- [순위19] - 128844newsML.txt 3글자 6번
- [순위20] - 239149newsML.txt 3글자 5번
- [순위21] - 128405newsML.txt 3글자 4번
- [순위22] - 236594newsML.txt 3글자 4번
- [순위23] - 122806newsML.txt 3글자 2번
- [순위24] - 128393newsML.txt 3글자 2번
- [순위25] - 205953newsML.txt 3글자 2번
- [순위26] - 207961newsML.txt 3글자 2번
- [순위27] - 113639newsML.txt 3글자 1번
- [순위28] - 125627newsML.txt 3글자 1번

128405newsML - 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

www.Bandicam.co.kr

Germany's David Prinosil ended rising British star Tim Henman's bid for
Prinosil beat Henman 6-4 6-3 in the semifinals of the \$475,000 Czech Ind
Seventh seed Henman, who had not lost a set all week, looked shaky on hi
But in the 10th game of the match, Henman's serve failed him, his double
Henman, who has now reached seven semifinals but never gone further, fou
The British number one recovered to gain two break points of his own in
Princ 찾기
Henma 찾을 내용 다음 찾기(B)
"I'm 방향
Prince 대/소문자 구분(C) 위로(U) 아래로(D)
Damm 취소
Victims on the way have increased world number three Michael Chang and e
For Korda the win came as a relief following claims in the local media t
The former world number five has suffered several injuries in the past t
"I felt a lot of pressure at the beginning of the match and I've never s



감사합니다!!