**과제 5. 끝말잇기**



|  |  |
| --- | --- |
| **과 목 :** | **프로그래밍** |
| **제출 일자 :** | **2018. 12. 08** |
| **담당 교수 :** | **김승태** |
| **학 과 :** | **사회복지학부** |
| **학 번 :** | **20170299** |
| **이 름 :** | **김유빈** |

**과제 5. 끝말 잇기**

1. 사전파일의 단어 개수(n) 새기

2. ‘동적할당’ -> n개의 배열 포인터

\*\* 처음에 따로 메모리(배열생성)에 저장하지 않고 파일 입출력만으로 하려고 했으나

프로그램 성능이 눈에 띄게 안 좋아져서

이를 포기하고 ‘동적할당’으로 프로그램 실행 중간에 배열 개수를 알아도 상관이 없었다.

**<단어 검열 순서>**

1. if(연결되는 단어인지)

else

printf(“연결되는 단어를 입력하세요.);

2. if(5글자인지)

else

printf(“5글자를 입력하세요.);

3. if(사전에 있는 단어인지)

else

printf(“사전에 있는 단어를 입력하세요.);

**4. if(명사인지)**

**-> 사전에 있는 단어 인데 명사가 아니다!**

else

printf(“명사를 입력하세요.);

1. if(연결되는 단어인지)

: 컴퓨터 단어 끝 == 사용자 단어 첫

2. if(5글자인지)

: strlen(user)

3. if(사전에 있는 단어인지) + if(명사인지)

: while(i !=n)일 때 까지 사전 단어인지 검사

사전파일 단어와 사용자 단어와 일치여부 판단

<힙>

abaca : n.마닐라삼(필리핀 주산)

abaci : n.abcus의 복수형

aback : ad.뒤로

abacus : n.계산기

abaft : ad.(배의)고물에

abandon : vt.버리다

abandoned : a.버림받은

abandoner : n.유기자

abandonment : n.포기

………

그 다음에 ‘명사’인지 판단

주의점>> token: 원본 문장(pstr[i])을 훼손시킴

* 사본 str 필요
* pstr[i]에 명사인지를 확인

char (\*pstr) [100]

**\*\*\*\*\*\*\*\*\*\*컴퓨터 단어 생성\*\*\*\*\*\*\*\*\*\***

**처음 생성>>**

while (1) {

**last = rand() % n;**

strcpy(str, pstr[last]);

token = strtok(str, " : ");

if (strlen(token) == 5)

if (strstr(pstr[last], "n.") != NULL) {

strcpy(word, token);

break;

}

}

n : 사전파일 단어의 개수 (따로 반복문을 돌려서 구함)

last = rand() % n; -> 0 ~ (n-1)의 배열 중 임의의 단어 선택

strcpy(str, pstr[last]); -> 사본 만들기

**그 이후에 생성>>**

생성된 컴퓨터 단어가 5글자 + 명사라는 조건 다 되면

중복되는 컴퓨터 단어 찾기(중요)

**count** : **comword(컴퓨터단어배열)**에서 단어 개수

for (j = 0;j <= count;j++)

if (!strcmp(comword[j], token)) {

j = -1; ->특수한 값 입력

break;

}//중복 검사

if (j != -1) {

strcpy(word, token);

strcpy(comword[++count], word);

break;

}//중복이 아니라면

만약 컴퓨터 단어 조건 충족 X

-> if (last + 1 < n)

last++;

else if(last + 1 >=n)

last = rand() % n;

**프로그램 성능 향상**을 위해 LAST(pstr의 번지)를 한 칸만 이동

