

# 백준 1316 그룹 단어 체커

202020988 조아영

# 문제

입력된 단어들 중 그룹단어의 개수 출력

- 그룹단어 ?

: 단어에 존재하는 모든문자에 대해 각 문자가 연속해서 나타나는 경우 만 존재하는 단어

ex) ccazzzzbb, kin -> 그룹단어 0

aa**bbb**ccb -> 그룹단어 x (b가 떨어져서 나타남)

# 아이디어

1. 단어에 등장하는 알파벳을 배열에 저장하여, 등장했던 단어인지, 처음 등장하는 단어인지 판단  
-> 등장했던 알파벳이 떨어져서 다시 등장 -> 그룹단어 x
2. 메인함수에 반복문 여러개, 복잡해보이고 한눈에 파악 어려움  
-> 단어 입력될 때 마다 그룹단어인지 판단하는 함수 호출,  
그룹 단어의 개수를 셈

# 코드

- 메인 함수

입력받은 N만큼 반복문 실행

문자열 입력 -> 함수 호출  
(그룹단어면 1, 아니면 0 반환)

결과 출력

```
result = 0;
```

```
for (int i = 0; i < N; i++)  
{  
    scanf("%s", word);  
    result += check_word(word);  
}
```

```
printf("%d", result);
```

# 코드

- 그룹단어인지 확인하는 함수

사용된 알파벳을  
저장할 문자열 배열 선언

문자열의 첫번째  
알파벳을 저장

```
int check_word(char w[]) //그룹단어면 1, 아니면 0
{
    int len, index;
    char check[26];

    len = strlen(w);
    index = 0;

    check[index++] = w[0];
```

# 코드

문자열의 길이만큼 반복문 실행

문자열의 이전 알파벳이랑 동일

-> 이미 check배열에 저장되었고,  
그룹문자의 조건에 어긋나지 않음

-> 다음 문자 확인

check배열에 이미 저장된 문자인지 확인

-> 동일한 문자가 있으면 0 반환

check 배열에 문자 추가

```
for (int i = 1; i < len; i++)  
{  
    if (w[i] == w[i - 1])  
        continue;  
  
    for (int j = 0; j < index; j++)  
    {  
        if (check[j] == w[i])  
        {  
            return 0;  
        }  
    }  
  
    check[index++] = w[i];  
}  
  
return 1;  
}
```

감사합니다 !