

Lea atentamente la definicion de las funciones [strlen](#) y [strtok](#), de la libreria [string.h](#); y [calloc](#), [malloc](#) y [free](#). Utilice los conocimientos adquiridos para pasar esta prueba.

Desarrolle un programa que separe las palabras de la oración, y lo ingrese en una cadena. Luego buscar la palabra más larga y y su posición.

Implemente las siguiente funciones

int PalabraMasLarga

1. Información

- int palabras_count
- char** palabras
- int* posicionPalabra

int cantidadTotalPalabras

1. Información

- char* separadores
- char* oracion

char** SeparandoPalabras

1. Información

- char* oracion
- char* separadores
- int* cantidadPalabras

void PartiendoCadenas

1. Información

- char* separadores
- char* oracion

2. Parametros de Impresión

- int cantidadPalabras
- char** misPalabras
- int tamanoPalabra

Se deberá imprimir, la cantidad de palabras totales en la oracion, la palabra más larga y su largo.

Input Format

```
- .,() [] {} <>  
Existen muchas formas de animar a la lectura. De hecho, lo dificil (que es lo que casi  
siempre logramos), es hacer que los alumnos detesten leer. Debemos tener en cuenta tanto las  
malas practicas como las buenas.
```

Constraints

- Borrar parametros int posicionPalabra_count, int cantidadPalabras_count, y int* result_count => **SE GENERAN AUTOMATICAMENTE**
- Ignore el tamaño de los separadores.
- Recuerde que no conoce el tamaño de la oración.
- Preste atención al crear la cadena de palabras, REVISE PUNTEROS DOBLES

Output Format

```
Hay 37 palabras, pero la mas larga [practicass -> 9 letras]
```

Sample Input 0

```
- .,() [] {} <>  
Existen muchas formas de animar a la lectura. De hecho, lo dificil (que es lo que casi  
siempre logramos), es hacer que los alumnos detesten leer. Debemos tener en cuenta tanto las  
malas practicas como las buenas.
```

Sample Input 1

```
- .,() [] {} <>  
Las horas, demasiadas horas, y mientras el Sol corria libre, la Luna se acabo cansando,  
hasta el punto de ir apartandose lentamente hasta dejarlo de nuevo al descubierto.
```

Sample Output 1

```
Hay 28 palabras, pero la mas larga [apartandose -> 11 letras]
```

Sample Input 2

```
- .,() [] {} <>  
MITOS: Aunque no son propiamente mitos, en este grupo incluyo muchas historias religiosas y  
fabulas mitologicas. Hay que comentar que muchas de esas historias religiosas son versiones  
con nombres de santos de la mitologia romana y griega, pues de esta procede gran parte de la  
cultura occidental.
```

Sample Output 2

```
Hay 47 palabras, pero la mas larga [propiamente -> 11 letras]
```