



---

# EJERCICIOS C

---

Ángel González García



## Contenido

Ejemplo 1. ....	4
Ejemplo 2. ....	6
¿Qué hace el siguiente código? .....	6
De haber algún problema, ¿Podrías decir en que línea se encuentra? .....	6
Define la regla que se incumple y propon una alternativa correcta siguiendo el SEI CERT C.....	7
Ejemplo 3. ....	8
¿Que hace el siguiente segmento de código si invocamos la función func con un 0? .....	8
De haber algun problema ¿Podrias decir la linea en la que se encuentra? .....	8
Crea un fichero con un main y ejecuta el segmento de código.....	9
Propón una solución al ejemplo que cumpla con las normal del CMU .....	9
Realiza un analisis estático del código erróneo y copia en tu solución el resultado. Utiliza las herramientas: (a) rats (b) cppchecker (c) splint (d) vera++ (e) valgrind .....	10

```
angel@angel-X541UJ:~/Escritorio/DPS$ splint code.c
Splint 3.1.2 --- 21 Feb 2021

code.c: (in function func)
code.c:9:14: Fall through case (no preceding break)
    Execution falls through from the previous case (use /*@fallthrough@*/ to mark
    fallthrough cases). (Use -casebreak to inhibit warning)
code.c:11:17: Fall through case (no preceding break)
code.c:7:19: Statement after switch is not a case: int i = 4
    The first statement after a switch is not a case. (Use -firstcase to inhibit
    warning)
code.c:3:13: Function exported but not used outside code: f
    A declaration is exported, but not used outside this module. Declaration can
    use static qualifier. (Use -exportlocal to inhibit warning)
code.c:20:1: Definition of f
code.c:5:6: Function exported but not used outside code: func
code.c:14:1: Definition of func

Finished checking --- 5 code warnings
```

```
./summit_xl_pad/src/summit_xl_pad.cpp:31: line is longer than 100 characters
./summit_xl_pad/src/summit_xl_pad.cpp:50: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:51: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:53: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:54: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:54: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:55: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:56: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:57: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:59: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:67: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:71: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:72: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:77: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:78: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:79: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:81: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:82: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:83: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:84: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:85: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:86: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:88: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:88: line is longer than 100 characters
./summit_xl_pad/src/summit_xl_pad.cpp:89: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:90: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:90: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:91: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:92: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:93: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:94: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:95: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:96: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:97: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:98: horizontal tab used
```

..... 10

```
angel@angel-X541UJ:~/Escritorio/DPS$ vera++ code.c
code.c:1: no copyright notice found
code.c:13: closing curly bracket not in the same line or column
code.c:14: closing curly bracket not in the same line or column
code.c:20: closing curly bracket not in the same line or column
code.c:27: closing curly bracket not in the same line or column
code.c:29: trailing empty line(s)
angel@angel-X541UJ:~/Escritorio/DPS$
```

Ejercicio 1..... 11

¿Qué hace el siguiente segmento de código? .....	11
¿Para qué se utiliza la variable vaeol? .....	11
Ejercicio 2.....	12
¿Qué hace el siguiente segmento de código? .....	12
Comenta que reglas/recomendaciones se están rompiendo aquí. Tambien entran pasadas.....	12
El programa permite mostrar el codigo desensamblado de la aplicacion, adjunta alguna captura. / Instala la herramienta perf para realizar el profiling de la aplicacion. Se puede instalar con apt. ....	12
¿Podrías decir cuál es la instrucción que más tiempo de CPU requiere? Adjunta una captura y describe la razón.....	13
¿Podrías decir cuál es la instrucción que más tiempo de CPU requiere? Adjunta una captura y describe la razón.....	13

## Ejemplo 1.

```
#include <stdio.h>

const char* p;

char* funcion1(void) {
    char array[10] = "Mi Cadena"
    return array; }

void funcion2(void) {
    const char cstr[] = "Todo va bien";
    p = cstr; }

void funcion3(void) {
    printf("%s\n", p); }

int main(void) {
    p = funcion1();
    funcion2();
    funcion3();
    return 0;}
```

Se incumple la norma DCL30-C. Declare objects with appropriate storage duration.

Vemos que el array declarado en la función 1 desaparece al retornar de la función 1.

Como solución se puede aplicar lo siguiente: Le damos una asignación dinámica al array y controlamos la forma de actuar del código en caso de que valga NULL

```
#include <stdio.h>
```

```
const char* p;

char* funcion1(void) {
char* array = (char*)malloc(10 * sizeof(char));

    if (array == NULL) {
        // Manejo de error adecuado

        return NULL; }
return array; }

void funcion2(void) {
    const char cstr[] = "Todo va bien";
    p = cstr; }

void funcion3(void) {
    printf("%s\n", p); }

int main(void) {
    p = funcion1();
    funcion2();
    funcion3();
    return 0;}
```

## Ejemplo 2.

```
#include <stdlib.h>

struct flexArrayStruct {
    int num;
    int data[1];
};

void func(size_t arraysize) {
    struct flexArrayStruct *structP = (struct flexArrayStruct *)
        malloc(sizeof(struct flexArrayStruct) + sizeof(int) * (arraysize - 1));
    if (structP == NULL) {
        /* Handle malloc failure */
    }
    structP->num = arraysize;
    for (size_t i = 0; i < arraysize; ++i) {
        structP->data[i] = 1;
    }
}
```

### ¿Qué hace el siguiente código?

Este código intenta crear y utilizar una estructura con un arreglo flexible. La función `func` aloja dinámicamente espacio suficiente para la estructura `flexArrayStruct` y un arreglo de `int` de tamaño `arraysize`, luego inicializa el campo `num` con el tamaño del arreglo y asigna el valor 1 a cada elemento del arreglo `data`.

### De haber algún problema, ¿Podrías decir en que línea se encuentra?

`Data` está declarado como un arreglo de un solo elemento, pero luego se utiliza como si su tamaño fuera `arraysize`, por lo que el problema estaría en la declaración de `Data` o en el último fragmento en el que se accede a `data[i]`

## Define la regla que se incumple y propon una alternativa correcta siguiendo el SEI CERT C

DCL38-C. Use the correct syntax when declaring a flexible array member.

```
#include <stdlib.h>

struct flexArrayStruct {
    int num;
    int data[];
};

void func(size_t arraysize) {
    struct flexArrayStruct *structP = malloc(sizeof((struct flexArrayStruct
*+arraysize)))
    malloc(sizeof(struct flexArrayStruct) + sizeof(int) * (arraysize - 1));
    if (structP == NULL) {
        /* Handle malloc failure */
    }
    structP->num = arraysize;
    for (size_t i = 0; i < arraysize; ++i) {
        structP->data[i] = 1;
    }
}
```



## Ejemplo 3.

```
#include <stdio.h>

extern void f(int i);

void func(int expr) {

    switch (expr) {

        int i = 4;

        f(i);

        case 0:

            i = 17;

        default:

            printf("%d\n", i);

    }

}
```

¿Que hace el siguiente segmento de código si invocamos la función func con un 0?

Lo normal sería que pasara a f un argumento con valor 17, pero en este caso antes del case 0 se está llamando a la función, además de darle a i valor 4, además, al no haber un break, incluso para 0 se irá por el default.

De haber algún problema ¿Podrías decir la línea en la que se encuentra?

Las líneas 7 y 8 hacen una declaración de una variable dentro de un switch antes de los case

Crea un fichero con un main y ejecuta el segmento de código

```
1  #include <stdio.h>
2
3  extern void f(int i);
4
5  void func(int expr) {
6      switch (expr) {
7          int i = 4;
8          f(i);
9          case 0:
10             i = 17;
11             default:
12                 printf("%d\n", i);
13         }
14     }
15
16     int main(void) {
17         func(0);
18         return 0;
19     }
20
21
```

17

Recibimos el print de i, que valdrá 17, por lo que entra por el default y también por el case 0

Propón una solución al ejemplo que cumpla con las normal del CMU

```
#include <stdio.h>

extern void f(int i);

void func(int expr) {

    int i = 4;

    switch (expr) {

        case 0:

            i = 17;

        default:

            f(i);

            printf("%d\n", i);

    }

}
```

Realiza un análisis estático del código erróneo y copia en tu solución el resultado. Utiliza las herramientas: (a) rats (b) cppchecker (c) splint (d) vera++ (e) valgrind

```
angel@angel-X541UJ:~/Escritorio/DPS$ splint code.c
Splint 3.1.2 --- 21 Feb 2021

code.c: (in function func)
code.c:9:14: Fall through case (no preceding break)
    Execution falls through from the previous case (use /*@fallthrough@*/ to mark
    fallthrough cases). (Use -casebreak to inhibit warning)
code.c:11:17: Fall through case (no preceding break)
code.c:7:19: Statement after switch is not a case: int i = 4
    The first statement after a switch is not a case. (Use -firstcase to inhibit
    warning)
code.c:3:13: Function exported but not used outside code: f
    A declaration is exported, but not used outside this module. Declaration can
    use static qualifier. (Use -exportlocal to inhibit warning)
    code.c:20:1: Definition of f
code.c:5:6: Function exported but not used outside code: func
    code.c:14:1: Definition of func

Finished checking --- 5 code warnings
```

```
./summit_xl_pad/src/summit_xl_pad.cpp:31: line is longer than 100 characters
./summit_xl_pad/src/summit_xl_pad.cpp:50: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:51: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:53: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:54: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:54: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:55: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:56: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:57: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:59: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:67: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:71: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:72: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:77: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:78: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:79: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:81: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:82: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:83: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:84: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:85: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:86: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:88: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:88: line is longer than 100 characters
./summit_xl_pad/src/summit_xl_pad.cpp:89: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:90: trailing whitespace
./summit_xl_pad/src/summit_xl_pad.cpp:90: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:91: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:92: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:93: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:94: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:95: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:96: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:97: horizontal tab used
./summit_xl_pad/src/summit_xl_pad.cpp:98: horizontal tab used
```

```
angel@angel-X541UJ:~/Escritorio/DPS$ vera++ code.c
code.c:1: no copyright notice found
code.c:13: closing curly bracket not in the same line or column
code.c:14: closing curly bracket not in the same line or column
code.c:20: closing curly bracket not in the same line or column
code.c:27: closing curly bracket not in the same line or column
code.c:29: trailing empty line(s)
angel@angel-X541UJ:~/Escritorio/DPS$
```

## Ejercicio 1.

### ¿Qué hace el siguiente segmento de código?

Este código define una función `average` que calcula el promedio de una serie de números enteros. La función acepta un número indefinido de argumentos enteros, siendo el primero obligatorio. Utiliza la variable especial `va_list` para acceder a estos argumentos variables. El ciclo `while` continúa sumando y contando los argumentos hasta que encuentra el valor especial `va_eol` (que se ha definido como `-1`), que indica el final de la lista de argumentos. Finalmente, la función devuelve el promedio de los números sumados, o `0` si no se sumó ningún número.

### ¿Para qué se utiliza la variable `vaeol`?

Al pasar `-1` de argumento cuando se llama a la función, `va_eol` sirve como una señal para detener el procesamiento de más argumentos.

Se utiliza como marcador del final de los argumentos variables pasados a la función `average`.

## Ejercicio 2.

### ¿Qué hace el siguiente segmento de código?

El programa calcula el factorial de un número predeterminado (12 si no se proporcionan argumentos) o de una serie de números desde 0 hasta j-1 si se proporciona un argumento al programa. j se determina por el primer argumento pasado al programa convertido de cadena a entero con atoi.

Comenta que reglas/recomendaciones se están rompiendo aquí.

También entran pasadas

DCL19-C. Minimize the scope of variables and functions

El programa permite mostrar el código desensamblado de la aplicación, adjunta alguna captura. / Instala la herramienta perf para realizar el profiling de la aplicación. Se puede instalar con apt.

```
Desensamblado de la sección load0:

fffffffff8f283d20 <load0>:
    push    %rbp
    mov     %rdi,%r8
    mov     %rsp,%rbp
    push    %r15
    push    %r14
    push    %r13
    push    %r12
    push    %rbx
    sub     $0x20,%rsp
    mov     0x18(%rdi),%r14
    mov     0x20(%rdi),%r9
    mov     0x28(%rdi),%r10
20:    mov     %r14,%r15
    mov     %r14,%rbx
    shr     $0x3,%r15
    xor     %bl,%bl
    mov     %r15d,%r12d
    and     $0xf,%r12d
    cmp     $0x2,%r12d
    ↓ ja     ec
    test    %r12d,%r12d
    ↓ jne    145
46:    xor     %r13d,%r13d
    xor     %ecx,%ecx
4b:    mov     (%rbx),%rax
    xor     %al,%al
    cmp     %rax,%rbx
```



¿Podrías decir cuál es la instrucción que más tiempo de CPU requiere? Adjunta una captura y describe la razón.

Samples: 9 of event 'cycles:P', Event count (approx.): 2305953

Children	Self	Command	Shared Object	Symbol
+ 63,19%	63,19%	a.out	[kernel.kallsyms]	[k] mtree_range_walk
+ 63,19%	0,00%	a.out	ld-linux-x86-64.so.2	[.] _dl_map_object
+ 63,19%	0,00%	a.out	ld-linux-x86-64.so.2	[.] memset
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] asm_exc_page_fault
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] exc_page_fault
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] do_user_addr_fault
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] handle_mm_fault
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] __handle_mm_fault
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] handle_pte_fault
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] do_fault
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] do_cow_fault
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] __anon_vma_prepare
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] find_mergeable_anon_vma
+ 63,19%	0,00%	a.out	[kernel.kallsyms]	[k] mas_walk
+ 32,46%	32,46%	a.out	[kernel.kallsyms]	[k] mas_wr_node_store
+ 32,46%	0,00%	a.out	[unknown]	[k] 0x00007efda96eb08b
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] entry_SYSCALL_64_after_
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] do_syscall_64
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] __x64_sys_execve
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] do_execveat_common.isra
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] bprm_execve
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] bprm_execve.part.0
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] exec_binprm
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] search_binary_handler
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] load_elf_binary
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] elf_map
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] vm_munmap
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] __vm_munmap
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] do_vmi_munmap
+ 32,46%	0,00%	a.out	[kernel.kallsyms]	[k] do_vmi_align_munmap

Mtree\_range\_walk es la que mas consume, ya que podemos ver que la columna “Self” presenta un 63,19%, no es directamente del programa sino del kernel

¿Podrías decir cuál es la instrucción que más tiempo de CPU requiere? Adjunta una captura y describe la razón.

Samples: 43K of event 'cycles:P', Event count (approx.): 37253501749

Children	Self	Command	Shared Object	Symbol
+ 99,99%	0,00%	a.out	libc.so.6	[.] libc_start_call_main
+ 99,99%	0,00%	a.out	a.out	[.] main
+ 99,99%	99,86%	a.out	a.out	[.] fib
0,06%	0,00%	a.out	[kernel.kallsyms]	[k] asm_common_interrupt
0,06%	0,00%	a.out	[kernel.kallsyms]	[k] common_interrupt
0,06%	0,00%	a.out	[kernel.kallsyms]	[k] asm_sysvec_apic_timer_irqs
0,06%	0,00%	a.out	[kernel.kallsyms]	[k] sysvec_apic_timer_interrupt

Fib es la instrucción que mas consume, posiblemente porque main llama a fib en todas sus iteraciones.