

Programa la función *elimHex* con el siguiente encabezado:

```
typedef unsigned long uint64_t;
uint64_t elimHex(uint64_t x, int h);
```

El parámetro *x* corresponde a un entero de 64 bits sin signo que debe ser visto como 16 cifras hexadecimales (de 4 bits cada cifra). La función *elimHex* debe entregar el resultado de eliminar todas las apariciones de la cifra *h* en *x*. Ejemplos de uso:

```
uint64_t rc1= elimHex(0x3a0ff0a3, 3); // rc1 es 0x00a0ff0a
uint64_t rc2= elimHex(0x3a0ff0a4, 0); // rc2 es 0x003affa4
uint64_t rc3= elimHex(0x3a0fe0b3, 0xf); // rc3 es 0x03a0e0b3
uint64_t rc4= elimHex(0x3a0fe0b3, 0xd); // rc3 es 0x3a0fe0b3
```

Restricciones:

- Ud. no puede usar los operadores de multiplicación, división o módulo (* / %). Use los operadores de bits eficientemente.
- Se descontará medio punto por no usar el estilo de indentación de Kernighan como se explica en [esta sección](#) de los apuntes.
- El estándar de C no especifica el resultado para desplazamientos mayores o iguales al tamaño del operando. *Sanitize* rechaza el desplazamiento $x \ll nbits$ cuando *nbits* es mayor o superior a la cantidad de bits de *x*.

Instrucciones

Baje *t1.zip* de U-cursos y descomprímalo. El directorio *T1* contiene los archivos (a) *test-elim.c* que prueba si su tarea funciona y compara su eficiencia con la solución del profesor, (b) *prof.ref-x86_64* y *prof.ref-aarch64* con los binarios ejecutables de la solución del profesor, (c) *elim.h* que incluye el encabezado de la función pedida, y (d) *Makefile* que le servirá para compilar y ejecutar su tarea. Ud. debe programar la función *elimHex* en el archivo *elim.c*.

Pruebe su tarea bajo Debian 12 nativo o virtualizado con VirtualBox, Vmware, QEmu o WSL 2. **Ejecute el comando *make* sin parámetros.** Le mostrará las opciones que tiene para compilar su tarea. Estos son los requerimientos para aprobar su tarea:

- *make run* debe felicitarlo por aprobar este modo de ejecución. Su solución no debe ser 80% más lenta que la solución del profesor.
- *make run-g* debe felicitarlo.
- *make run-san* debe felicitarlo y no reportar ningún problema como por ejemplo desplazamientos indefinidos.

Cuando pruebe su tarea con *make run* asegúrese que su computador esté configurado en modo alto rendimiento y que no estén corriendo otros procesos intensivos en uso de CPU al mismo tiempo. De otro modo podría no lograr la eficiencia solicitada.

Entrega

Ud. solo debe entregar por medio de U-cursos el archivo *elim.zip* generado por el comando *make zip*. **A continuación es muy importante que descargue de U-cursos el mismo archivo que subió, luego descargue nuevamente los archivos adjuntos y vuelva a probar la tarea tal cual como la entregó.** Esto es para evitar que Ud. reciba un 1.0 en su tarea porque entregó los archivos equivocados. Créame, sucede a menudo por ahorrarse esta verificación. Se descontará medio punto por día de atraso. No se consideran los días de receso, sábados, domingos o festivos.