

Apellidos, Nombre:

DNI:

Instrucciones

Realice los ejercicios tal como se detallan en cada uno de ellos. El profesor puede proporcionar folios vacíos para poder anotar lo que se necesite para la resolución del examen. El contenido de esos folios no se evaluará pero serán entregados al profesor.

Descargue el zip proporcionado en Moodle llamado *AlternativaBloqueII-ApellidosNombre.zip*. **Descomprímalo** y **renombre** la carpeta sustituyendo *ApellidosNombre* por los suyos. Por ejemplo, si mi nombre es Antonio Manuel Durán Rosal, el directorio pasa a llamarse *AlternativaBloqueII-DuranRosalAntonioManuel* (no incluir **ni espacios ni caracteres especiales, como la ñ o tildes**).

El directorio contiene una carpeta por cada ejercicio de programación que debe resolverse utilizando un editor de texto plano y su correspondiente compilación y ejecución en el sistema operativo Linux. **Al profesor sólo se le subirán los códigos fuente sin incluir el ejecutable**. Aunque a continuación se detallan los ejercicios y el qué hay que realizar, a lo largo de los ficheros aparece la palabra **TODO** que indicará lo mismo.

Reglas y aclaraciones para todos los ejercicios

- Obligatorio el uso del calificador `const`.
- Usar las directivas `#ifndef`, `#define` y `#endif`; o `#pragma once`.
- No modificar nada de los archivos `main.cpp`, sólo descomentar aquellas partes que se vayan completando para que compile.
- Por lo general, podrá incluir las funciones que necesite en los archivos `.h` y `.cpp`, pero se evaluará el funcionamiento de los apartados que se piden.
- Comprobar los **TODO**.
- Ante cualquier duda, contactar con el profesor.

Hay que entregar

- En Moodle el directorio completo en formato `.zip` con el nombre *AlternativaBloqueII-ApellidosNombre.zip* (poner los apellidos y nombre del alumno).
- **Antes de darle a enviar** los documentos en Moodle, **llamar al profesor** para comprobar que todo es correcto. **No se aceptarán envíos sin la supervisión del profesor.**

Confirmando que he leído las instrucciones del examen. Firma y hora de entrega:

Ejercicio 1**Tiempo estimado: 50min****Puntuación: 4.5**

En el Reino Champiñón, un misterioso desajuste en el Gran Motor Numérico está causando fallos en puentes móviles, ascensores y bloques sorpresa. Según el Toad Ingeniero, el problema surge al intentar multiplicar números enormes: el motor tarda tanto que los *Goombas* se cuelan por los circuitos y los desactivan.

Para evitar que el caos se extienda, la Princesa Peach envía una misión urgente a los hermanos **Mario** y **Luigi**, famosos por resolver averías imposibles en tuberías matemáticas y engranajes aritméticos. Su tarea: rediseñar el módulo de multiplicación usando técnicas avanzadas de descomposición numérica.

Los números que deben multiplicar son dos enteros (**long int**):

$$u \quad \text{y} \quad v$$

cada uno con exactamente n dígitos. Como los conductos del motor solo permiten transportar la mitad de las cifras a la vez, se requiere dividir cada número:

$$u = 10^{n/2}a + b, \quad v = 10^{n/2}c + d$$

donde a, b, c, d tienen $n/2$ dígitos.

Se pide:

- a) **(2.25 puntos) Restaurar el Módulo de Cuatro Engranajes (mario4)**. Durante la inspección, Mario encuentra un panel antiguo con el esquema clásico del motor, donde se indica que el producto se computaba siguiendo:

$$u \cdot v = ac \cdot 10^n + (ad + bc) \cdot 10^{n/2} + bd.$$

El módulo original usaba cuatro engranajes de multiplicación, uno para cada bloque intermedio. Para reactivarlo, debe implementarse un algoritmo recursivo:

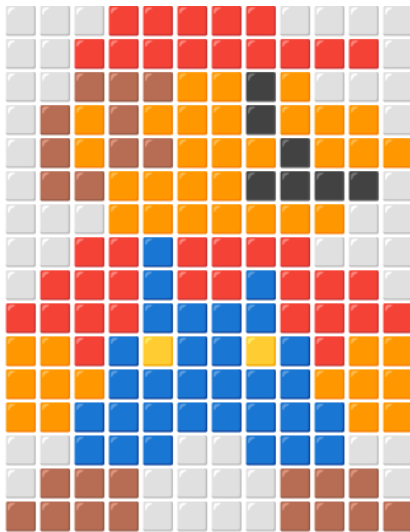
- el caso base es cuando $n = 1$,
- en el resto de casos, se realizan las multiplicaciones recursivas: ac , ad , bc y bd .

- b) **(2.25 puntos) Activar el Ahorro de Potencia Verde (luigi3)**. Mientras Mario trabaja con los engranajes, Luigi investiga la zona de mantenimiento y descubre un ducto secreto con un bloque luminoso. Al golpearlo, aparece inscrita la siguiente identidad energética:

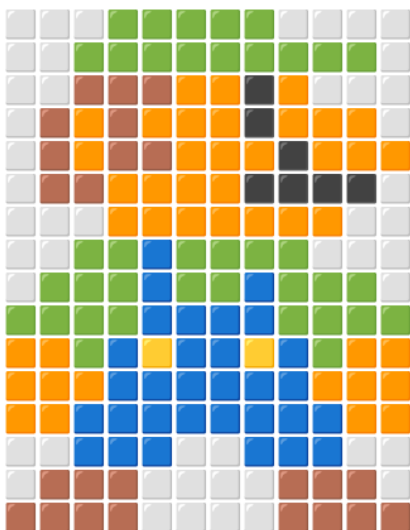
$$ad + bc = (a - b)(d - c) + ac + bd.$$

Esta relación permite reducir el número de multiplicaciones necesarias, optimizando el consumo del Motor Numérico. A partir de ella debe implementarse una versión alternativa del cálculo del producto con únicamente tres multiplicaciones recursivas, conocida como **luigi3**.

La salida por pantalla sería:



```
1 Resultado: 1357699038045678
2 Tiempo: 1.3e-05 s
```



```
1 Resultado: 1357699038045678
2 Tiempo: 1e-06 s
```

Ejercicio 2

Tiempo estimado: 60min

Puntuación: 5.5

El Reino Champiñón está preparando su gran **Festival de Cortos Estelares**, un evento que reunirá a Toads, Yoshis, Koopas y demás habitantes del reino para disfrutar de proyecciones cinematográficas provenientes de todos los mundos del universo.

La Princesa Peach, preocupada porque Bowser pueda arruinar el evento si este dura demasiado, ha pedido ayuda a Mario y Luigi para que organicen los cortos de la manera más eficiente posible. El festival contará con una lista de *cortos* (cada uno con identificador, nombre y duración) y varias *salas* del Castillo Real donde podrán proyectarse en paralelo.

El problema: **si el festival se alarga demasiado, Bowser llegará con sus esbirros antes de que termine**. Por tanto, debemos distribuir los cortos en las salas de forma que la duración total del festival sea **lo más pequeña posible**.

Contáis con un esqueleto formado por:

- `corto.h` y `corto.cpp`: representan un *corto del Reino Champiñón*, con su identificador, nombre y duración.
- `sala.h` y `sala.cpp`: representan una sala del castillo, con su identificador, lista de cortos y el tiempo total ocupado.
- `solucionfestival.h` y `solucionfestival.cpp`: almacenan el conjunto de salas, el tiempo total del festival y el número de salas disponibles.
- `estadofestival.h` y `estadofestival.cpp`: representan un estado intermedio del proceso, para que Mario y Luigi puedan avanzar o retroceder en la búsqueda.
- `problemafestival.h` y `problemafestival.cpp`: contienen el algoritmo de *backtracking* de Mario y el algoritmo *voraz* de Luigi.

La misión consiste en completar el algoritmo para ayudar al Reino Champiñón a organizar el festival antes de que Bowser aparezca (**41 minutos**).

Se pide:

- Implemente la sobrecarga del operador `<` en la clase `Corto` (**0.5 puntos**).
- Implemente la función `addCortoInSala` (**0.25 puntos**).
- Implemente la función `deleteCortoInSala` (**0.25 puntos**).
- Implemente la función `avanza` (**0.25 puntos**).
- Implemente la función `retrocede` (**0.25 puntos**).
- Implemente la función `getAlternativas` (**0.5 puntos**).
- Implemente la función `getMejorAlternativa` (**0.75 puntos**).

- h) Implemente la función `esFinal` (0.25 puntos).
- i) Implemente la función `ejecutaBacktracking` (0.25 puntos).
- j) Implemente la función `bt` (0.75 puntos).
- k) Implemente la función `ejecutaVoraz` (0.25 puntos).
- l) Implemente la función `vorazR` (0.75 puntos).
- m) Implemente la función `actualizarMejorSolucion` (0.5 puntos).

La salida por pantalla sería:

```
1 Mostramos la lista de cortos introducidos:
2 (Corto: 1, Nombre: Super Mario y la Estrella Dorada, Duracion: 10)
3 (Corto: 2, Nombre: Luigi y la Mansión Fantasma, Duracion: 14)
4 (Corto: 3, Nombre: La Invasión de los Koopas, Duracion: 20)
5 (Corto: 4, Nombre: Carrera en la Rainbow Road, Duracion: 21)
6 (Corto: 5, Nombre: Toad y el Bosque Champiñón, Duracion: 10)
7 (Corto: 6, Nombre: Yoshi: Aventura en Isla Huevo, Duracion: 12)
8 (Corto: 7, Nombre: Bowser: Fuego y Maldades, Duracion: 13)
9 (Corto: 8, Nombre: Goombas al Ataque, Duracion: 8)
10 (Corto: 9, Nombre: Peach y el Festival Estelar, Duracion: 11)
11
12 Ordenamos la lista segun el criterio de la clase Corto:
13 (Corto: 4, Nombre: Carrera en la Rainbow Road, Duracion: 21)
14 (Corto: 3, Nombre: La Invasión de los Koopas, Duracion: 20)
15 (Corto: 2, Nombre: Luigi y la Mansión Fantasma, Duracion: 14)
16 (Corto: 7, Nombre: Bowser: Fuego y Maldades, Duracion: 13)
17 (Corto: 6, Nombre: Yoshi: Aventura en Isla Huevo, Duracion: 12)
18 (Corto: 9, Nombre: Peach y el Festival Estelar, Duracion: 11)
19 (Corto: 1, Nombre: Super Mario y la Estrella Dorada, Duracion: 10)
20 (Corto: 5, Nombre: Toad y el Bosque Champiñón, Duracion: 10)
21 (Corto: 8, Nombre: Goombas al Ataque, Duracion: 8)
22
23 Algoritmo Voraz (Luigi) *****
24
25 Tiempo total: 42
26 Numero de salas: 3
27 Sala: 0
28 Tiempo: 42
29 Cortos:
30 (Corto: 4, Nombre: Carrera en la Rainbow Road, Duracion: 21)
31 (Corto: 9, Nombre: Peach y el Festival Estelar, Duracion: 11)
32 (Corto: 5, Nombre: Toad y el Bosque Champiñón, Duracion: 10)
33 Sala: 1
34 Tiempo: 40
35 Cortos:
36 (Corto: 3, Nombre: La Invasión de los Koopas, Duracion: 20)
37 (Corto: 6, Nombre: Yoshi: Aventura en Isla Huevo, Duracion: 12)
38 (Corto: 8, Nombre: Goombas al Ataque, Duracion: 8)
39 Sala: 2
40 Tiempo: 37
41 Cortos:
42 (Corto: 2, Nombre: Luigi y la Mansión Fantasma, Duracion: 14)
```

```
43 (Corto: 7, Nombre: Bowser: Fuego y Maldades, Duracion: 13)
44 (Corto: 1, Nombre: Super Mario y la Estrella Dorada, Duracion: 10)
45
46 Algoritmo Backtracking (Mario) *****
47
48 Tiempo total: 40
49 Numero de salas: 3
50 Sala: 0
51 Tiempo: 40
52 Cortos:
53 (Corto: 4, Nombre: Carrera en la Rainbow Road, Duracion: 21)
54 (Corto: 9, Nombre: Peach y el Festival Estelar, Duracion: 11)
55 (Corto: 8, Nombre: Goombas al Ataque, Duracion: 8)
56 Sala: 1
57 Tiempo: 40
58 Cortos:
59 (Corto: 3, Nombre: La Invasión de los Koopas, Duracion: 20)
60 (Corto: 1, Nombre: Super Mario y la Estrella Dorada, Duracion: 10)
61 (Corto: 5, Nombre: Toad y el Bosque Champiñón, Duracion: 10)
62 Sala: 2
63 Tiempo: 39
64 Cortos:
65 (Corto: 2, Nombre: Luigi y la Mansión Fantasma, Duracion: 14)
66 (Corto: 7, Nombre: Bowser: Fuego y Maldades, Duracion: 13)
67 (Corto: 6, Nombre: Yoshi: Aventura en Isla Huevo, Duracion: 12)
```