

**Instituto Tecnológico y de Estudios Superiores de Monterrey**  
**Campus Estado de México**  
**Escuela de Ingeniería y Ciencias, Región Ciudad de México**  
**Departamento de Computación**

## Examen de Compiladores

Profesor: Ariel Ortiz Ramírez

Clave y grupo: Tc3002b.201

Nombre: \_\_\_\_\_

Matrícula: \_\_\_\_\_

*Apegándome al Código de Ética de los Estudiantes del Tecnológico de Monterrey, me comprometo a que mi actuación en este examen esté regida por la honestidad académica. En congruencia con el compromiso adquirido con dicho código, realizaré este examen de forma honesta y personal, para reflejar, a través de él, mi conocimiento y aceptar, posteriormente, la evaluación obtenida.*

Firma: \_\_\_\_\_

**MUY IMPORTANTE:** Este examen se debe resolver de manera individual. Cualquier tipo de deshonestidad será sancionada con una calificación de 1/100 y será reportada al Comité de Integridad Académica de la institución. Esta sanción es para todas las personas involucradas en la falta.

## Instrucciones

Lee cuidadosamente todas las preguntas de este examen y responde en las hojas proporcionadas por separado.

**Tiempo límite:** 90 minutos.

1. (20%) Escribe una *expresión regular* que permita reconocer fechas válidas usando el formato YYYY-MM-DD definido en el estándar ISO 8601. Se debe considerar lo siguiente para procurar que la gran mayoría de las fechas reconocidas sean válidas (aunque no necesariamente todas):

- El año YYYY puede ir del 0000 al 9999.
- En el mes MM el primer dígito debe ser 0 o 1. Si el primer dígito es 0, el segundo debe ser cualquier dígito del 1 al 9. Si el primer dígito es 1, el segundo dígito debe ser 0, 1 o 2.
- En el día DD el primer dígito debe ser 0, 1, 2 o 3. Si el primer dígito es 0, el segundo debe ser cualquier dígito del 1 al 9. Si el primer dígito es 1 o 2, el segundo dígito debe ser cualquier dígito del 0 al 9. Si el primer dígito es 3, el segundo dígito debe ser 0 o 1.

Así, por ejemplo, las siguientes fechas deben ser reconocidas por la expresión regular:

- 2023-04-28
- 1521-08-13
- 9999-12-31

Sin embargo, las siguientes fechas deben ser rechazadas:

- 0000-00-00
- 2023-04-32
- 1998-13-15

2. (20%) Explica brevemente para qué sirven los paquetes de Python *Arpeggio* y *Wasmtime* utilizados en clase.
3. (20%) Se tiene la siguiente gramática PEG:

```
foo    = bar baz quz? EOF
bar    = (corge / grault)*
baz    = (garply / waldo / fred)+
quz    = plugh / thud
corge  = 'a'
grault = 'b'
garply = 'c'
waldo  = 'd'
fred   = 'e'
plugh  = 'f'
thud   = 'g'
```

Suponiendo que la producción raíz es `foo`, dibuja el árbol de parseo (*parse tree*) que se genera al reconocer la siguiente cadena de entrada:

abcedg

4. (20%) Se tiene la siguiente definición de módulo en *WebAssembly text format* (WAT).

```
(module
  (func
    (export "mystery")
    (param $n i32)
    (result i32)
    (local $a i32)
    (local $b i32)
    i32.const 0
    local.set $a
    i32.const 1
    local.set $b
    loop
      local.get $b
      local.get $b
      local.get $a
      i32.add
      local.set $b
      local.set $a
      local.get $n
      i32.const 1
      i32.sub
      local.set $n
      local.get $n
      br_if 0
    end
    local.get $a
  )
)
```

¿Qué valor devuelve la función `mystery` al ser invocada con cada uno de los valores (parámetro `n`) indicados a continuación?

- i - `mystery(1)`
- ii - `mystery(2)`
- iii - `mystery(5)`
- iv - `mystery(7)`

5. (20%) Escribe un módulo completo en *WebAssembly text format* (WAT) que contenga una función exportada llamada `average`. Dicha función debe recibir como entrada tres argumentos de tipo `f64` (punto flotante de 64 bits) y debe devolver un valor de tipo `f64` con el promedio aritmético de los tres valores de entrada.

Por ejemplo, `average(1.25, 5.25, -14.0)` debe devolver `-2.5`.