

Programación Avanzada (1113) UNLaM

Evaluación Final Virtual Febrero 02 /02/ 2021

Ejercicio 1

java complejidad computacional

Escribir la clase **Serie**, con un método que resuelva en forma eficiente, en cuanto a su complejidad computacional, el valor del término N de la serie.

La serie es similar a Fibonacci, pero donde cada término es la suma de los 3 anteriores (en lugar de 2). Ej.: **0, 0, 1, 1, 2, 4, 7, 13, ...**

Realizar un Main que calcule para 1.000.000 de números al azar, entre 1 y 100, el correspondiente valor de la serie.

Expresar la complejidad computacional de la solución pedida.

NOTA: la buena resolución de este ejercicio es indispensable para la aprobación.

Ejercicio 2

prolog

Codifique en Prolog las reglas necesarias para obtener el término N en una serie similar a Fibonacci, pero donde cada término es la suma de los 3 anteriores (en lugar de 2).

Ej.: **0, 0, 1, 1, 2, 4, 7, 13, ...**

Ejercicio 3

```
class Alpha {
    String getType() { return "alpha"; }
}

class Beta extends Alpha {
    String getType() { return "beta"; }
}

class Gamma extends Beta {
    String getType() { return "gamma"; }

    public static void main(String[] args) {
        Gamma g1 = new Alpha();
        Gamma g2 = new Beta();
        System.out.println(g1.getType() + " " + g2.getType());
    }
}
```

¿Cuál es la salida?

Ejercicio 4

El grafo de la figura representa las distancias entre los distintos edificios de una universidad y los caminos entre ellos.

Se pide:

- Decidir si un vigilante puede salir de A, recorrer todos los edificios una sola vez y volver al punto de partida. Justificar.
- Decidir si un vigilante puede recorrer todos los caminos una sola vez. Justificar.

