

Desarrollo de Aplicaciones Móviles en Java

Lab 1: Evaluación Técnica

Completa los siguientes problemas y enseña tus soluciones a los instructores. Queremos entender la habilidad y nivel de programación de cada participante. Debes estar preparado para explicar cómo funciona y trabaja tu código.

No esperamos que todos terminen todos los problemas. Solo trata de hacer lo más que puedas. Para cada problema se va a proveer un archivo esqueleto (Number1.java, Number2.java, ...)

1. El código en el archivo Number1.java imprime "Hello, World" a la pantalla. Modifica el código para que imprima "Hola, AITI".
2. Agrega una línea de código a el archivo Number2.java para que imprima $(9 \times N + 2)$. Por ejemplo, el código debe imprimir 38 si $N=4$.
3. El operador % regresa el restante de un número cuando se divide con otro número. Por ejemplo, $4 \% 3$ es 1 porque 4 es solo 1 más que un múltiple de 3. Similarmente, $9 \% 3$ es 0 porque 9 es un múltiple de 3. Este operador se puede utilizar con un variable ($x \% 3$). Agrega declaraciones "if-then" al código en el archivo Number3.java para que imprima "Div 3" si N es divisible por 3 ($N \% 3$ es 0), "Div 5" si N es divisible por 5 y "Div3Div5" si N es divisible por 3 y 5. Si N no es divisible por 3 ni por 5 solo imprime el número N. Por ejemplo, si $N=17$, imprime 17. Si $N=15$, imprime "Div3Div5"

4. a) Usa una repetición (for loop) y el código del problema #3 para imprimir lo siguiente:

```
1
2
Div3
4
Div5
Div3
7
8
Div3
Div5
11
Div3
13
14
Div3Div5
```

- b) Usa repeticiones anidadas (nested for loops) para imprimir lo siguiente:

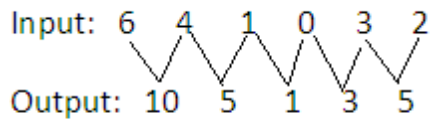
```
1
12
123
1234
```

12345
123456

Crédito extra: Resuelve este problema solo usando una línea de repetición (for loop) – Sin usar repeticiones anidadas (nested for loops).

5. Escribe código para que imprima los elementos de una matriz (array) separados con comas. Por ejemplo, {1,2,3,4} debe imprimir como: 1 2 3 4.
6.
 - a) Escribe código para reducir una matriz (array) como muestra el siguiente ejemplo:

Input: 6 4 1 0 3 2
Output: 10 5 1 3 5



Donde $10 = 6 + 4$, $5 = 4 + 1$, $1 = 1 + 0$...

- b) Agrega código para reducir el “array” una segunda vez.
 - c) Agrega código para que reduzca el “array” a un solo numero.
7.
 - a) En Java, un `double` representa un número flotante (como 3.141). Escribe un método que toma un “array” de `doubles` y un `double` como argumentos. El método debe multiplicar cada elemento del “array” por ese `double`. Por ejemplo, si los argumentos del método son {1.2, 2.1} y 2.0, el resultado debe ser {2.4, 4.2}. No te preocupes si los números no son exactos – puede haber diferencias en redondeo. Estos errores (o inexactitudes) son comunes con números `double`.
 - b) Agrega un método en este mismo archivo que toma un “array” de `doubles` y un `double` como argumentos. Debe de multiplicar cada otro elemento en el “array” por ese `double` (el primer elemento, tercero, quinto...). Por ejemplo, si los argumentos del método son {1.2, 2.1, 1.1} y 2.0, el resultado debe ser {2.4, 2.1, 2.2}.
 - c) Agrega un método en este mismo archivo que toma un “array” de `doubles` y dos números `doubles` como argumentos. Debe multiplicar los elementos nones (primero, tercero, quinto...) por el primer numero `double` y multiplicar los elementos pares (segundo, cuarto, sexto...) por el segundo numero `double`. Por ejemplo, si los argumentos son {1.2, 2.1, 1.1}, 2.0 y 3.0, el resultado debe ser {2.4, 6.3, 2.2}.

Trata de escribir este método en solo dos líneas de código. Puedes asumir que ninguno de los dos argumentos `double` es cero.