

Prueba Teórica y Práctica de Java (Nivel Intermedio)

Johan Camilo López Girón

Fecha Inicio: 2021-06-20 11:02:35 Fecha Fin: 2021-06-20 11:49:00

Puntaje Total:

83/100

Resultado:

El candidato demuestra tener habilidades para realizar tareas de programación, pudiendo realizar tareas de complejidad media sin mayores problemas. Las habilidades de programación del candidato son mayores al promedio y reflejan experiencia y/o buenas aptitudes para la programación.



Habilidad en Programación Alto

Tiempo Total: 46 Minutos 25 Segundos

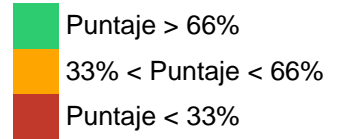
Percentil: 68%

Tiempo en la Página de la Prueba: 62%

Información Candidato	Información Prueba	Información Instancia
<p>Id: 97838</p> <p>Nombre: Johan Camilo López Girón</p> <p>Notas: -</p> <p>Registrado: 2021-06-18</p> <p>Email: johan.lopez-g@outlook.com</p>	<p>Nombre/ID: Prueba Teórica y Práctica de Java (Nivel Intermedio) / 349</p> <p>Descripción Prueba Teórica y Práctica de Java de nivel medio, incluye solo preguntas de programación de nivel intermedio. Se recomienda utilizar esta prueba para evaluar candidatos Senior o Semi-Senior para posiciones que requieren realizar tareas convencionales de programación. Esta prueba no es recomendable para perfiles junior ni para posiciones con requerimientos de muy alta complejidad.</p> <p>Nivel de Dificultad: Medio</p>	<p>Límite de Días: -</p> <p>Tiempo Límite: 120 minutes</p> <p>Tipo: TST</p> <p>Proceso: Developer Kafka (Intermedio)</p> <p>Monitoreo Web: No</p>

Resumen por Categoría

Descripción	Resultado	Puntaje (Valor)
Programación		53/60
Resultado Categoría:		

Resultado Categoría:

Resumen por Pregunta

Descripción	Categoría	Nivel de Dificultad	Puntaje
1.- La combinación de comandos Java para capturar excepciones es:	PRG T	Fácil	10 de 10
2.- ¿Cual de las siguientes implementa correctamente el inicio de un bucle en Java?	PRG T	Fácil	10 de 10
3.- Cual de las siguientes alternativas declara, construye e inicializa correctamente el arreglo myList?	PRG T	Medio	0 de 10
4.- Que método se utiliza para iniciar la ejecución de un thread?	PRG T	Medio	10 de 10
5.- Crear Histograma en base a arreglo	PROG	Medio	14 de 20
Escenarios de Validación:	100 %		
Eficiencia del Código:	0%		
6.- Encuentra el número con la secuencia más larga en el arreglo	PROG	Medio	19 de 20
Escenarios de Validación:	100 %		
Eficiencia del Código:	76 %		
7.- Encontrar los elementos que suman 10	PROG	Medio	20 de 20
Escenarios de Validación:	100 %		

	No Respondido
	Parcialmente Correcto
	Incorrecto
	Correcto

Detalles por Pregunta

1.- La combinación de comandos Java para capturar excepciones es:

Tipo:	Categoría:	Puntaje:
Pregunta de texto con respuesta de texto de eleccion individual	PRG T	10 of 10
Nivel de Dificultad:	Familia:	SubFamilia:
Fácil	Programación	Java

Texto de la Pregunta:

La combinación de comandos Java para capturar excepciones es:

Respuesta:

- ☐ commit/rollback
- ☒ try/catch **Correcto**
- ☐ on error/goto error
- ☐ handler/error
- ☐ start transaction/capture

2.- ¿Cual de las siguientes implementa correctamente el inicio de un bucle en Java?

Tipo:	Categoría:	Puntaje:
Pregunta de texto con respuesta de texto de eleccion individual	PRG T	10 of 10
Nivel de Dificultad:	Familia:	SubFamilia:
Fácil	Programación	Java

Texto de la Pregunta:

¿Cual de las siguientes implementa correctamente el inicio de un bucle en Java?

Respuesta:

- ☒ for (int i = 0; i < 10; i++) { **Correcto**
- ☐ for i=1 to 10
- ☐ Loop int i=1 to 10 do
- ☐ Function loop(
- ☐ for (1..10)

3.- Cual de las siguientes alternativas declara, construye e inicializa correctamente el arreglo myList?

Tipo:	Categoría:	Puntaje:
-------	------------	----------

Pregunta de texto con respuesta
de texto de eleccion individual

PRG T

0 of 10

Nivel de Dificultad:

Medio

Familia:

Programación

SubFamilia:

Java

Texto de la Pregunta:

Cual de las siguientes alternativas declara, construye e inicializa correctamente el arreglo myList?

Respuesta:

☐ `int [] myList = {"1", "2", "3"};`

☒ `int myList[] = (5, 8, 2);` **Incorrecto**

☐ `int myList [] [] = {4,9,7,0};`

☐ `int [] myList = {4, 3, 7};` **Alternativa Correcta**

4.- Que método se utiliza para iniciar la ejecución de un thread?

Tipo:

Pregunta de texto con respuesta
de texto de eleccion individual

Categoría:

PRG T

Puntaje:

10 of 10

Nivel de Dificultad:

Medio

Familia:

Programación

SubFamilia:

Java

Texto de la Pregunta:

Que método se utiliza para iniciar la ejecución de un thread?

Respuesta:

☐ `init();`

☒ `start();` **Correcto**

☐ `run();`

☐ `resume();`

5.- Crear Histograma en base a arreglo

Tipo:

Pregunta de programación con
codigo predefinido

Categoría:

PROG

Puntaje:

14 of 20

Nivel de Dificultad:

Medio

Familia:

Programación

SubFamilia:

Java

Texto de la Pregunta:

Escribir un programa en Java que recorra un arreglo y genere un histograma en base a los números de este. El arreglo se llama myArray y contiene 10 elementos que corresponden a números enteros del 1 al 5. Un histograma representa que tanto un elemento aparece en un conjunto de datos (Debe mostrar la frecuencia para todos los números del 1 al 5, incluso si no están presentes en el arreglo). Por ejemplo, para el arreglo:

myArray:=(1,2,1,3,3,1,2,1,5,1) el histograma se vería así:

```
1: *****
2: **
3: **
4:
5: *
```

El código para declarar y poblar myArray ya está ahí, puede editarlo para probar con otros valores y puede hacer clic en el botón de actualizar junto a él para volver al valor original que se utilizará para validar su código durante la prueba. Asegúrese de que los resultados se impriman exactamente como aparecen aquí, ya que incluso un espacio faltante o sobrante puede marcar la pregunta como incorrecta (Notar espacio entre los ":" y el primer asterisco y que los resultados deben mostrarse ordenados de forma ascendente).

Respuesta:

```
private static final int MIN_NUMBER = 1;
private static final int MAX_NUMBER = 5;
public static void main (String args[]) {

    final Map<Integer, Integer> numberKeys = new HashMap<>();

    for (int i = MIN_NUMBER; i <= MAX_NUMBER; i++) {
        numberKeys.put(i, 0);
    }

    for (int numberKey: myArray) {
        final int numTimes = numberKeys.get(numberKey);
        numberKeys.put(numberKey, numTimes + 1);
    }

    for (int key: numberKeys.keySet()) {
        final String stars = getStarsFromNumber(numberKeys.get(key));
        System.out.println(key + ": " + stars);
    }
}

private static String getStarsFromNumber(int number) {
    String stars = "";

    for (int i = 0; i < number; i++) {
        stars += "**";
    }

    return stars;
}
```

Escenarios de Validación

Descripción	Estado
Caso Extremo ignora Espacio	Correcto
Caso Básico Estricto	Correcto
Caso Básico ignora Espacio	Correcto
Caso Respuesta Aproximada	Correcto
Caso Fácil en Orden y con Todos los Números	Correcto

6.- Encuentra el número con la secuencia más larga en el arreglo**Tipo:****Categoría:****Puntaje:**

Nivel de Dificultad:

Medio

Familia:

Programación

SubFamilia:

Java

Texto de la Pregunta:

Tiene un arreglo (llamado myArray) con 10 elementos (enteros en el rango de 1 a 9). Escriba un programa en Java que imprima el numero que tiene mas ocurrencias seguidas en arreglo y también imprimir la cantidad de veces que aparece en la secuencia.

El código que llena el arreglo ya esta escrito, pero puede editarlo para probar con otros valores. Con el botón de refrescar puede recuperar el valor original que sera utilizado para evaluar la pregunta como correcta o incorrecta durante la ejecución.

Su programa debe analizar el arreglo de izquierda a derecha para que en caso de que dos números cumplan la condición, el que aparece por primera vez de izquierda a derecha será el que se imprimirá. La salida de los datos para el arreglo en el ejemplo (1,2,2,5,4,6,7,8,8,8) sería la siguiente:

Longest: 3

Number: 8

En el ejemplo, la secuencia mas larga la tiene el numero 8 con una secuencia de tres ochos seguidos. Tenga en cuenta que el código que escriba debe imprimir los resultados exactamente como se muestra con el fin de que la pregunta sea considerada válida.

El código para declarar y poblar myArray ya está ahí, puede editarlo para probar con otros valores y puede hacer clic en el botón de actualizar junto a él para volver al valor original que se utilizará para validar su código durante la prueba.

Respuesta:

```
private static final int ARRAY_MAX_LENGTH = 10;

public static void main (String args[]) {
    int longestSequence = 1;
    int longestSequenceNumber = 0;
    int tempLongestSequence = 1;

    for (int i = 1; i < ARRAY_MAX_LENGTH; i++) {
        if (myArray[i - 1] == myArray[i]) {
            tempLongestSequence += 1;

            if (tempLongestSequence > longestSequence) {
                longestSequence = tempLongestSequence;
                longestSequenceNumber = myArray[i - 1];
            }
        } else {
            tempLongestSequence = 1;
        }
    }

    System.out.println("Longest: " + longestSequence);
    System.out.println("Number: " + longestSequenceNumber);
}
```

Escenarios de Validación

Descripción	Estado
Todos iguales	Correcto
Caso Básico Estricto	Correcto
Caso doble	Correcto
Caso Básico Simple	Correcto
Caso Secuencia Duplicada	Correcto
Caso Respuesta Aproximada	Correcto

7.- Encontrar los elementos que suman 10**Tipo:**

Pregunta de programación con
codigo predefinido

Categoría:

PROG

Puntaje:

20 of 20

Nivel de Dificultad:**Familia:****SubFamilia:**

Texto de la Pregunta:

Escribir un programa utilizando Java que encuentre los dos elementos del arreglo que sumados dan 10. Se deben imprimir ambos números como resultado separados por un espacio (en el orden en que aparecen en el arreglo).

Por ejemplo, para el arreglo (1,3,4,2,7,0) el resultado seria: 3 7

Respuesta:

```
public static void main (String args[]) {  
    for (int i = 0; i < myArray.length; i++) {  
        for (int j = i + 1; j < myArray.length; j++) {  
            if ((myArray[i] + myArray[j]) == 10) {  
                System.out.println(myArray[i] + " " + myArray[j]);  
            }  
        }  
    }  
}
```

Escenarios de Validación

Descripción	Estado
Escenario Básico	Correcto
Escenario Números Duplicados	Correcto
Escenario Simple sin 5 sin espacios	Correcto
Escenario Básico Simple	Correcto