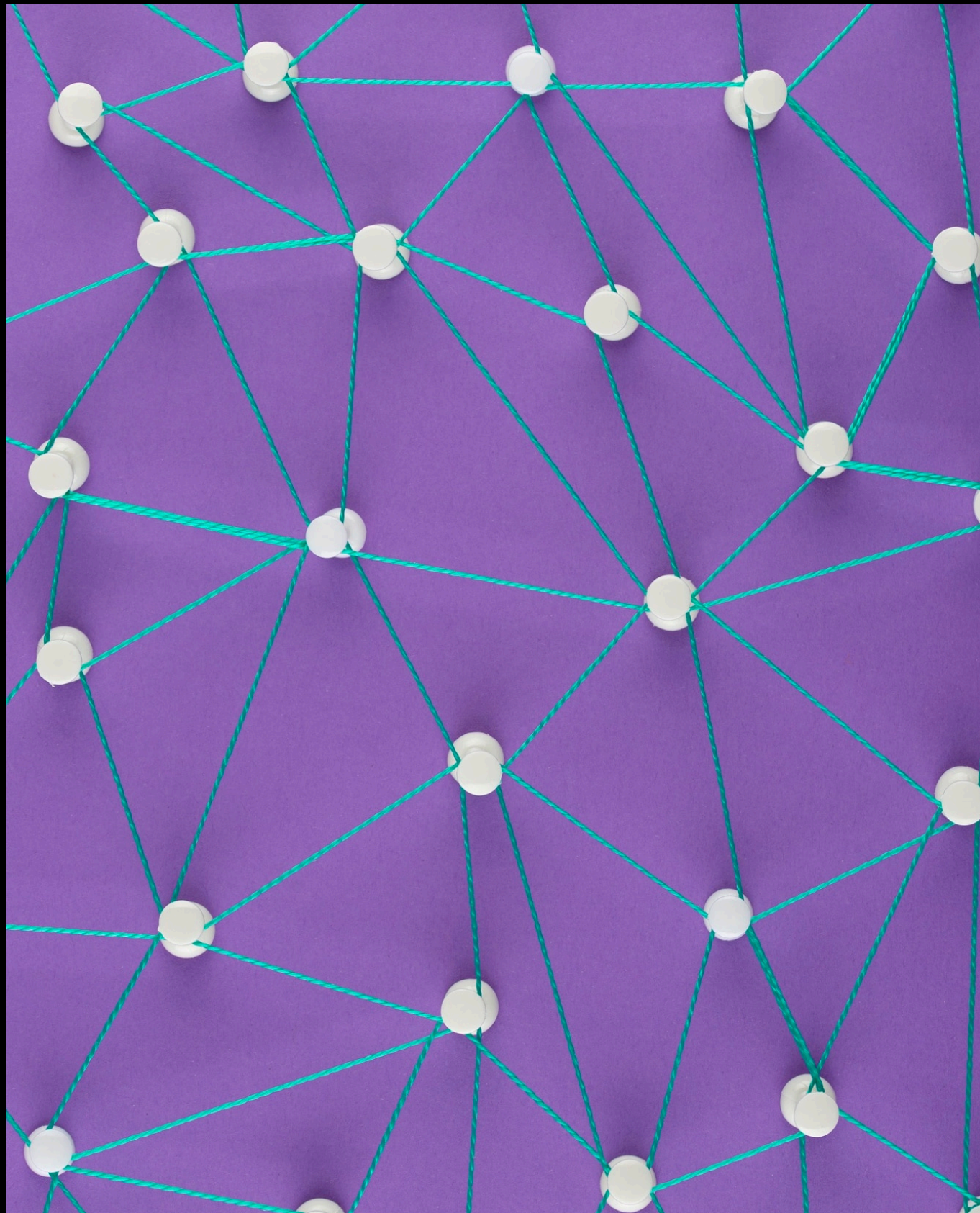




Explorando la Recursividad en Java: Una Mirada Profunda



Introducción

La **recursividad** es una técnica fundamental en programación que permite a una función **llamarse a sí misma**.

En esta presentación exploraremos su aplicación en Java y sus implicaciones en el diseño de algoritmos. ¡Prepárate para una mirada profunda!

Concepto de Recursividad

La **recursividad** es un concepto clave en programación que implica la **autollamada** de una función. Al aplicar este enfoque en Java, es crucial comprender la **condición de terminación** para evitar bucles infinitos. La recursividad puede simplificar la solución de problemas complejos.



Implementación en Java

En Java, la **recursividad** se implementa mediante la llamada **recursiva** de una función.

Es esencial comprender cómo se manejan las llamadas en la **pila de ejecución** y cómo se gestiona la memoria.

La recursividad puede mejorar la **legibilidad** del código.

```
#include <stdio.h>
```

```
int n_factor(int n)
```

```
{
```

```
    int res =1 , i;
```

```
    for (i = 1; i <= n; i++){
```

```
        res = res * i;
```

```
    }
```

```
    return res;
```

```
}
```

```
int main(int argc, char const *argv[])
```

```
{
```

```
    int n = 3, res = n_factor(n);
```

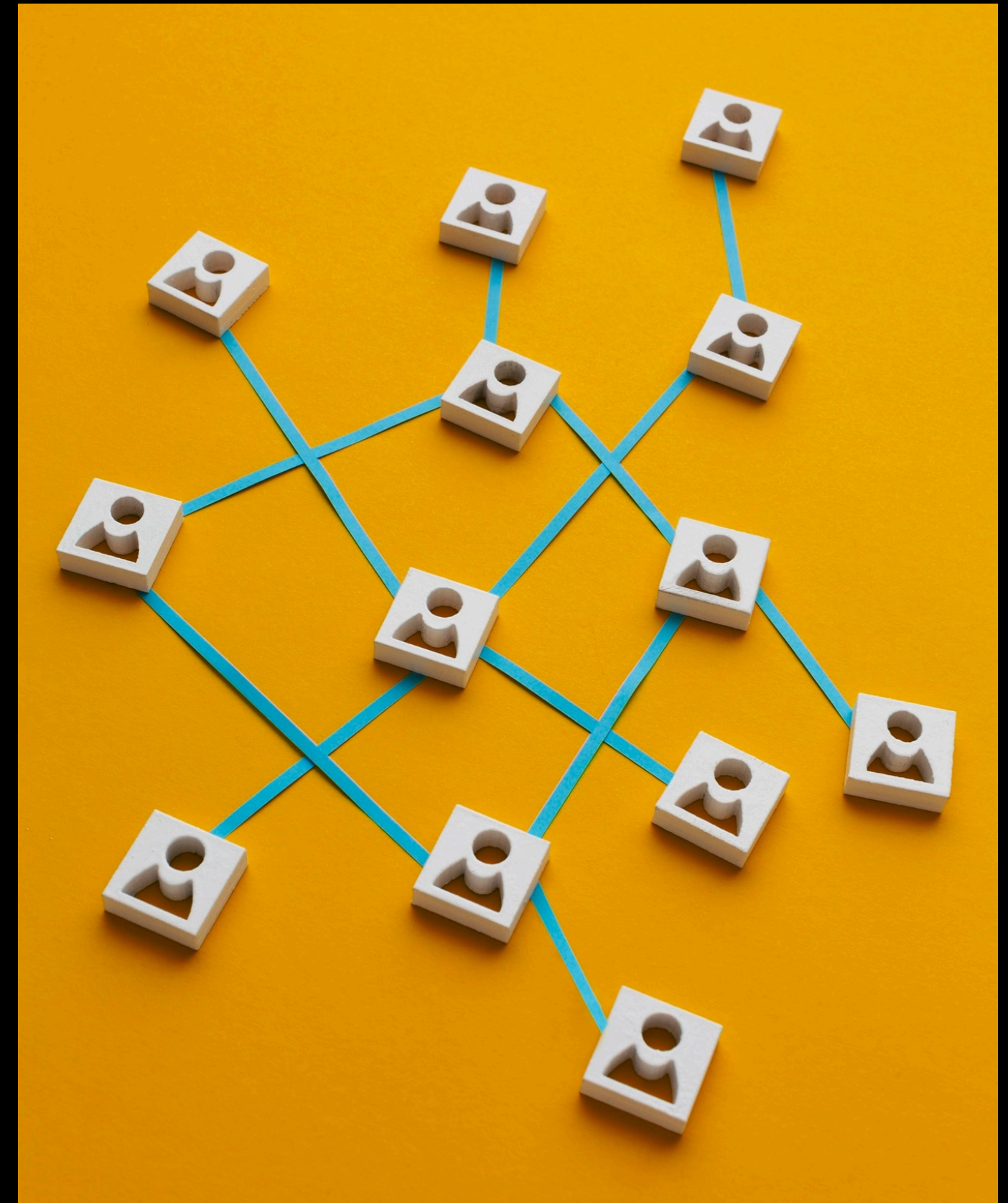
```
    printf("El resultado del factoria de %d", res);
```

```
    return 0;
```

```
}
```


Casos de Uso Comunes

La **recursividad** se utiliza en una variedad de aplicaciones, como la **búsqueda de rutas** en grafos, la **traversía de árboles** y la **división y conquista**. Al comprender estos casos de uso, podemos aplicar la recursividad de manera efectiva en Java para resolver problemas complejos.





Consideraciones y Desafíos

Al utilizar la **recursividad** en Java, es crucial considerar la **eficiencia** y la **gestión de la memoria**. Los **bucles infinitos** y la **profundidad de la pila** son desafíos comunes que deben abordarse. La recursividad también puede requerir un enfoque adicional en la **depuración**.

Conclusión

La **recursividad** en Java ofrece un enfoque poderoso para resolver problemas complejos mediante la **autollamada** de funciones. Al dominar este concepto, los programadores pueden mejorar la **estructura** y **eficiencia** de sus algoritmos. ¡Explora la recursividad con confianza y creatividad!

src > MultipliRecursividad.java > MultipliRecursividad

```
1 public class MultipliRecursividad {  
2  
3     int factor;  
4  
5     void tabla(int x) {  
6         if (x > 0) {  
7             factor = x;  
8             for (int i = 10; i >= 1; i--) {  
9                 int resultado = (i * factor);  
10                System.out.println(i + " * " + factor + " = " + resultado);  
11            }  
12            tabla(x - 1);  
13        }  
14    }  
15
```

Run | Debug

```
16 public static void main(String[] args) {  
17     MultipliRecursividad mur = new MultipliRecursividad();  
18     mur.tabla(x:8);  
19 }  
20 }  
21
```

6 * 8 = 48

5 * 8 = 40

4 * 8 = 32

3 * 8 = 24

2 * 8 = 16

1 * 8 = 8

PS C:\Users\pandr\OneDrive\Documentos\Word\JavaVS\recursividad>