

Renteria Magana Rayni Damian

Tarea 6

La recursividad es aquella propiedad que posee una función por la cual dicha función puede llamarse a si misma. Se puede utilizar la recursividad como una alternativa a la iteración .

Una solución recursiva es normalmente menos eficiente en términos de tiempo de computadora que una solución iterativa debido a las operaciones auxiliares que llevan consigo las llamadas suplementarias a las funciones; sin embargo, en muchas ocasiones el uso de recursión permite a los programadores especificar soluciones sencillas, naturales, que serían en caso contrario difíciles de resolver.

En recursión directa el código de la función F contiene una sentencia que invoca a la función F. En recursión indirecta la función F llama a otras funciones, y estas funciones llaman a F en algún momento.

#Ejemplo de recursividad en python

```
def fact(num):  
    if num==1:  
        return 1  
  
    else:  
        return num*fact(num-1)  
  
num=int(input("Numero a calcular el factorial: "))  
print(fact(num))
```

Referencias

Aguilar, L. J. (1996). *Fundamentos de programación: Algoritmos y estructura de datos*.

McGraw-Hill.

Takeyas, B. L. (2012). *Estructuras de datos orientadas a objetos: Pseudocódigo y aplicaciones en C # .NET*.