

INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO
DEPARTAMENTO ACADÉMICO DE COMPUTACIÓN.

Estructura de Datos Avanzada.

Gerardo Andres Moguel Roveló

Tarea No. 3

La mochila azul.

El problema de la mochila, en términos sencillos, es un desafío clásico en la optimización de recursos. Imagina que tienes una mochila con capacidad limitada y una lista de objetos, cada uno con un peso y un valor. El objetivo es maximizar el valor total que puedes llevar en la mochila sin exceder su capacidad. Es como decidir qué empacar para un viaje, tratando de llevar tantas cosas valiosas como sea posible sin sobrecargar la mochila.

Implementaciones en la clase.

Nos entregaron el código de solución para el problema de la mochila, pero el profesor también nos proporcionó un código para generar semillas. Luego, nos explicó la importancia de insertar la semilla tanto antes como después de ejecutar la función de la mochila, con el fin de verificar si había algún error durante las pruebas. Esta práctica nos permite asegurarnos de que los resultados obtenidos fueran consistentes y reproducibles, ayudándonos a detectar posibles problemas en la implementación del algoritmo o en la generación de números aleatorios. En este caso, se implementó la semilla dentro del main de la clase Mochila.java (*Anexo 1*).

Implementación en .bat

El uso del "for" en este caso específico puede considerarse innecesario ya que el programa no falla con los números proporcionados en los casos de prueba. Sin embargo, es importante reconocer que la inclusión de la semilla en el código puede ser una práctica valiosa para proyectos futuros. Aunque en este ejercicio particular no se presentan problemas, la semilla puede desempeñar un papel crucial en proyectos más complejos donde se trabaje con algoritmos sensibles a la inicialización de números aleatorios.

Se puede ver el código del .bat (*Anexo 2*) y lo que se muestra en la terminal (*Anexo 3*).

Anexos.

Anexo 1. Implementación de semilla en main de clase Mochila

```
public static void main(String[] args) {  
    Long SEMILLA = args.length > 0 ? Long.parseLong(args[0]) : (long) (10000.0 *  
Math.random() + 1);  
    java.util.Random rnd = new java.util.Random(SEMILLA);  
    System.out.println("Seed: " + SEMILLA);  
  
    int cuantos = 20; // artículos de donde elegir  
    int nums = (int) Math.pow(2, cuantos);  
  
    double capMochila = 10.0;  
    double pesoMochila = 0.0;  
    double valorMochila = 0.0;  
  
    double pesoArt[] = new double[cuantos];  
    double valorArt[] = new double[cuantos];  
    genArticulos(cuantos, pesoArt, valorArt);  
  
    double pesoCombinacion;  
    double valorCombinacion;  
    int combinacionMax = 0;  
    int b[];  
    for (int k = 1; k < nums; k++) {  
        b = binario(k, cuantos);  
        pesoCombinacion = 0.0;  
        valorCombinacion = 0.0;  
        for (int j = 0; j < cuantos; j++) {  
            pesoCombinacion += pesoArt[j] * b[j];  
            valorCombinacion += valorArt[j] * b[j];  
        }  
    }  
}
```

```

        if (pesoCombinacion < capMochila) {
            if (valorCombinacion > valorMochila) {
                valorMochila = valorCombinacion;
                pesoMochila = pesoCombinacion;
                combinacionMax = k;
            }
        }
    }

    System.out.println("ValorMochila:" + valorMochila + ", PesoMochila:" + pesoMochila);
    b = binario(combinacionMax, cuantos);

    String strRes = "";
    for (int k = cuantos - 1; k >= 0; k--) {
        strRes = b[k] + strRes;
    }

    System.out.println("Cuantos_obj:" + cuantos + ", Num_combinacion:" +
combinacionMax + ", Num_combinacion_binario:" + strRes);

    System.out.println("Seed: " + SEMILLA + "\n");
}

```

Anexo 2. Implementación de clase mochila en .bat, para generar las pruebas en Símbolo de Sistema.

```

@echo off
echo =====
echo Ejercicio de la mochila.
echo =====
echo "Comienzo de ejecucion"
echo .
echo .
echo .

```

```

for /L %%k in (1,1,5) do (
    echo Prueba: %%k
    java -jar "C:\Users\andre\OneDrive\Escritorio\EDA\Mochila\dist\Mochila.jar"
)
echo .
echo .
echo .
echo "Fin de ejecucion"
echo =====

```

Anexo 3. Resultado del código .bat en Símbolo de Sistema

```

C:\Users\andre\OneDrive\Escritorio\EDA>Mochila_seed.bat
=====
Ejercicio de la mochila.
=====
"Comienzo de ejecucion"
.
.
.
Prueba: 1
Seed: 6779
ValorMochila:506.2575981024431, PesoMochila:9.942478222732278
Cuantos_obj:20, Num_combinacion:851611, Num_combinacion_binario:11011001011111110011
Seed: 6779

Prueba: 2
Seed: 4189
ValorMochila:436.2905809316946, PesoMochila:9.94681168316085
Cuantos_obj:20, Num_combinacion:578365, Num_combinacion_binario:10111100110010110001
Seed: 4189

Prueba: 3
Seed: 9926
ValorMochila:506.41536705472896, PesoMochila:9.979204003057275
Cuantos_obj:20, Num_combinacion:376808, Num_combinacion_binario:0001011111111111011010
Seed: 9926

Prueba: 4
Seed: 6931
ValorMochila:481.8629183810664, PesoMochila:9.98856063933193
Cuantos_obj:20, Num_combinacion:805887, Num_combinacion_binario:11111111110100100011
Seed: 6931

Prueba: 5
Seed: 1153
ValorMochila:417.29309314055814, PesoMochila:9.86010423091739
Cuantos_obj:20, Num_combinacion:778846, Num_combinacion_binario:01111010010001111101
Seed: 1153

Prueba: 6
Seed: 8062
ValorMochila:419.75205203893626, PesoMochila:9.859118858731714
Cuantos_obj:20, Num_combinacion:1026526, Num_combinacion_binario:01111011100101011111
Seed: 8062
.
.
.
"Fin de ejecucion"
=====
C:\Users\andre\OneDrive\Escritorio\EDA>

```