

Tecnológico de Costa Rica

Tarea Programada 3

Asignatura: Lenguajes de Programación

Estudiante: Josué Suárez Campos – 2016089518

Profesor: José Araya Monge

2018

DOCUMENTACIÓN TAREA PROGRAMADA - EIFFEL

1. Introducción

La presente tarea programada consiste en dar una solución programa, utilizando el lenguaje de programación Eiffel, a un problema de empaçado. Dicho problema consiste en colocar objetos de diferentes tamaños dentro de cajas con una capacidad dada de almacenamiento, buscando minimizar el número de cajas requeridas.

Para ello, se utilizarán tres algoritmos diferentes:

- First Fit: En este algoritmo los objetos no son ordenados y se van insertando según el orden en que aparecen buscando la primer caja en que puedan ser insertados.
- First Fit Decreasing: Los objetos en este algoritmo son ordenados de forma decreciente y busca colocar cada objeto en la primera caja disponible que tenga campo suficiente.
- Best Fit: Este algoritmo no ordena los objetos y los va insertando en la caja más llena y en la que queda espacio para ser insertados.

2. Estructuras de clases usadas

En la solución programa se utilizaron las siguientes clases:

- MAIN: Es la clase principal del proyecto, en ella se llaman las otras clases y se realizan las entradas de datos.
- CAJA: Define las características que conlleva cada caja del problema a resolver.
- OBJETO: Define las característica asociadas a los objetos del problema.
- FIT: Es una clase padre que pretende asociar las características similares de dos algoritmos: FF y BF
- FF: Corresponde al algoritmo First Fit, el cual hereda las características de la clase FIT.
- BF: Corresponde al algoritmo Best Fit, el cual hereda las características de la clase FIT
- FFD: Corresponde al algoritmo First Fit Decreasing, el cual hereda las características de la clase FF.

3. Dar las instrucciones para ejecutar el programa

- ❖ Ingresar al IDE Eiffel Studio
- ❖ Ingresar al botón Add project y seleccionar la ubicación donde se encuentra almacenado el proyecto programado.
- ❖ Presionar el botón Open y esperar a que la solución programa se muestre. Puede que sea necesaria una precompilación.
- ❖ Clic en Compile y ejecutar por medio del botón Run
- ❖ Al estar abierta la consola indique los datos solicitados por el programa, si no desea especificar algún dato, indíquelo con el número -1 para que se asigne el valor por defecto para esa característica
- ❖ Los datos ingresados serán mostrados en forma resumida
- ❖ Seguidamente, los algoritmos serán ejecutados en orden: First Fit, First Fit Decreasing, Best Fit.

4. Corridas de ejemplo

- Corrida 1: usar valores default,
tamaño_cajas=10, tamaño_max_objetos=7, semilla=2018, número_objetos=20.

Datos generales:

```
- Tamaño de las cajas: 10  
- Tamaño maximo de los objetos: 7  
- Semilla: 2018  
- Cantidad de objetos generados: 20  
- Suma del tamaño de los objetos: 93
```

Listado de objetos: (Identificador, tamaño)

```
(1, 1) (2, 2) (3, 7) (4, 5) (5, 2) (6, 6) (7, 2) (8, 6) (9, 5) (10, 7)  
(11, 3) (12, 7) (13, 5) (14, 7) (15, 6) (16, 7) (17, 2) (18, 1) (19, 6) (20, 6)
```

-----Ejecutando el algoritmo First Fit-----

```
1001: [ (3, 7) (11, 3) ]  
1002: [ (10, 7) (7, 2) (1, 1) ]  
1003: [ (12, 7) (5, 2) (18, 1) ]  
1004: [ (14, 7) (17, 2) ]  
1005: [ (16, 7) (2, 2) ]  
1006: [ (6, 6) ]  
1007: [ (8, 6) ]  
1008: [ (15, 6) ]  
1009: [ (19, 6) ]  
1010: [ (20, 6) ]  
1011: [ (13, 5) (4, 5) ]  
1012: [ (9, 5) ]
```

Cantidad de cajas utilizadas: 12

Promedio de ocupacion: 0.7750000000000002

-----Ejecutando el algoritmo First Fit Decreasing-----

```
1001: [ (3, 7) (11, 3) ]  
1002: [ (10, 7) (7, 2) (1, 1) ]  
1003: [ (12, 7) (5, 2) (18, 1) ]  
1004: [ (14, 7) (17, 2) ]  
1005: [ (16, 7) (2, 2) ]  
1006: [ (6, 6) ]  
1007: [ (8, 6) ]  
1008: [ (15, 6) ]  
1009: [ (19, 6) ]  
1010: [ (20, 6) ]  
1011: [ (13, 5) (4, 5) ]  
1012: [ (9, 5) ]
```

Cantidad de cajas utilizadas: 12

Promedio de ocupacion: 0.7750000000000002

```
-----Ejecutando el algoritmo Best Fit-----  
1001: [ (3, 7) (2, 2) ]  
1002: [ (10, 7) (17, 2) ]  
1003: [ (12, 7) (5, 2) (18, 1) ]  
1004: [ (14, 7) (7, 2) (1, 1) ]  
1005: [ (16, 7) (11, 3) ]  
1006: [ (6, 6) ]  
1007: [ (8, 6) ]  
1008: [ (15, 6) ]  
1009: [ (19, 6) ]  
1010: [ (20, 6) ]  
1011: [ (13, 5) ]  
1012: [ (4, 5) ]  
1013: [ (9, 5) ]  
  
Cantidad de cajas utilizadas: 13  
Promedio de ocupacion: 0.7153846153846154
```

- Corrida 2:
tamaño_cajas=12, tamaño_max_objetos=7, semilla=2018, número_objetos=20.

```
Datos generales:
- Tamano de las cajas: 12
- Tamano maximo de los objetos: 7
- Semilla: 2018
- Cantidad de objetos generados: 20
- Suma del tamano de los objetos: 93

Listado de objetos: (Identificador, tamano)
(1, 1) (2, 2) (3, 7) (4, 5) (5, 2) (6, 6) (7, 2) (8, 6) (9, 5) (10, 7)
(11, 3) (12, 7) (13, 5) (14, 7) (15, 6) (16, 7) (17, 2) (18, 1) (19, 6) (20, 6)

-----Ejecutando el algoritmo First Fit-----
1001: [ (3, 7) (13, 5) ]
1002: [ (10, 7) (4, 5) ]
1003: [ (12, 7) (9, 5) ]
1004: [ (14, 7) (11, 3) (7, 2) ]
1005: [ (16, 7) (5, 2) (17, 2) (1, 1) ]
1006: [ (6, 6) (8, 6) ]
1007: [ (15, 6) (19, 6) ]
1008: [ (20, 6) (2, 2) (18, 1) ]

Cantidad de cajas utilizadas: 8
Promedio de ocupacion: 0.96875

-----Ejecutando el algoritmo First Fit Decreasing-----
1001: [ (3, 7) (13, 5) ]
1002: [ (10, 7) (4, 5) ]
1003: [ (12, 7) (9, 5) ]
1004: [ (14, 7) (11, 3) (7, 2) ]
1005: [ (16, 7) (5, 2) (17, 2) (1, 1) ]
1006: [ (6, 6) (8, 6) ]
1007: [ (15, 6) (19, 6) ]
1008: [ (20, 6) (2, 2) (18, 1) ]

Cantidad de cajas utilizadas: 8
Promedio de ocupacion: 0.96875
```

```
-----Ejecutando el algoritmo Best Fit-----  
1001: [ (3, 7) (5, 2) (17, 2) (1, 1) ]  
1002: [ (10, 7) (11, 3) (7, 2) ]  
1003: [ (12, 7) (9, 5) ]  
1004: [ (14, 7) (4, 5) ]  
1005: [ (16, 7) (13, 5) ]  
1006: [ (6, 6) ]  
1007: [ (8, 6) ]  
1008: [ (15, 6) ]  
1009: [ (19, 6) ]  
1010: [ (20, 6) ]  
1011: [ (2, 2) ]  
1012: [ (18, 1) ]  
  
Cantidad de cajas utilizadas: 12  
Promedio de ocupacion: 0.6458333333333337
```

- Corrida 3:
tamaño_cajas=12, tamaño_max_objetos=8, semilla=2019, número_objetos=25.

```
Datos generales:
- Tamano de las cajas: 12
- Tamano maximo de los objetos: 8
- Semilla: 2019
- Cantidad de objetos generados: 25
- Suma del tamano de los objetos: 119

Listado de objetos: (Identificador, tamano)
(1, 6) (2, 5) (3, 1) (4, 5) (5, 2) (6, 3) (7, 7) (8, 5) (9, 5) (10, 7)
(11, 7) (12, 8) (13, 2) (14, 4) (15, 5) (16, 3) (17, 1) (18, 3) (19, 7) (20, 3)
(21, 8) (22, 6) (23, 5) (24, 6) (25, 5)

-----Ejecutando el algoritmo First Fit-----
1001: [ (12, 8) (14, 4) ]
1002: [ (21, 8) (18, 3) (17, 1) ]
1003: [ (7, 7) (4, 5) ]
1004: [ (10, 7) (15, 5) ]
1005: [ (11, 7) (2, 5) ]
1006: [ (19, 7) (8, 5) ]
1007: [ (1, 6) (22, 6) ]
1008: [ (24, 6) (23, 5) (3, 1) ]
1009: [ (9, 5) (25, 5) (13, 2) ]
1010: [ (6, 3) (20, 3) (16, 3) (5, 2) ]

Cantidad de cajas utilizadas: 10
Promedio de ocupacion: 0.9916666666666667

-----Ejecutando el algoritmo First Fit Decreasing-----
1001: [ (12, 8) (14, 4) ]
1002: [ (21, 8) (18, 3) (17, 1) ]
1003: [ (7, 7) (4, 5) ]
1004: [ (10, 7) (15, 5) ]
1005: [ (11, 7) (2, 5) ]
1006: [ (19, 7) (8, 5) ]
1007: [ (1, 6) (22, 6) ]
1008: [ (24, 6) (23, 5) (3, 1) ]
1009: [ (9, 5) (25, 5) (13, 2) ]
1010: [ (6, 3) (20, 3) (16, 3) (5, 2) ]

Cantidad de cajas utilizadas: 10
Promedio de ocupacion: 0.9916666666666667
```

```
-----Ejecutando el algoritmo Best Fit-----  
1001: [ (12, 8) (18, 3) (17, 1) ]  
1002: [ (21, 8) (14, 4) ]  
1003: [ (7, 7) ]  
1004: [ (10, 7) ]  
1005: [ (11, 7) ]  
1006: [ (19, 7) ]  
1007: [ (1, 6) ]  
1008: [ (22, 6) ]  
1009: [ (24, 6) ]  
1010: [ (4, 5) ]  
1011: [ (15, 5) ]  
1012: [ (2, 5) ]  
1013: [ (8, 5) ]  
1014: [ (23, 5) ]  
1015: [ (9, 5) ]  
1016: [ (25, 5) ]  
1017: [ (6, 3) ]  
1018: [ (20, 3) ]  
1019: [ (16, 3) ]  
1020: [ (13, 2) ]  
1021: [ (5, 2) ]  
1022: [ (3, 1) ]  
  
Cantidad de cajas utilizadas: 22  
Promedio de ocupacion: 0.450757575757575
```

5. Comentarios finales (estado del programa)

El problema dado se resolvió en su totalidad.