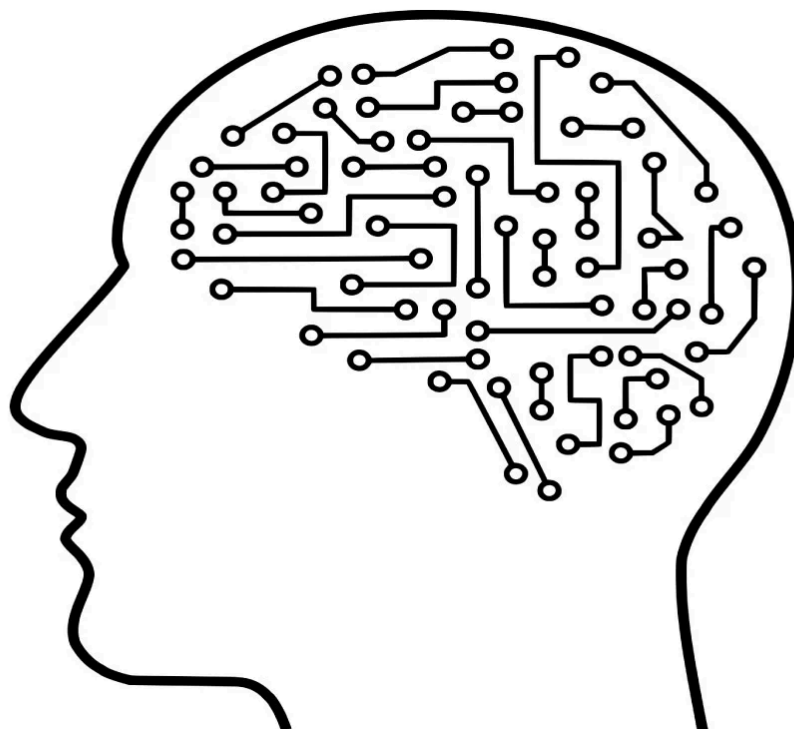


ALGORITMIA BÁSICA



PRÁCTICA 4

Jesús López Ansón - 839922
Javier Sin Pelayo - 843442

Análisis de los resultados

Como bien se indica en *LEEME.md*, en el script *ejecutar.sh*, para las pruebas, hemos hecho que por defecto aplique el algoritmo a 6 ficheros de interés:

- *1_prueba.txt*: es un fichero cuyo contenido es idéntico al del ejemplo presentado en el guión.
- *2_singleArticle.txt*: contiene un bloque con un único artículo.
- *3_moreArticles.txt*: es una versión del fichero inicial *1_prueba.txt*, pero al cual se le han añadido algunos artículos a cada uno de los dos bloques.
- *4_tricky.txt*: caso de prueba con trampas para el algoritmo.
- *5_evenMoreArticles.txt*: caso de estudio con un bloque conteniendo un número elevado de artículos
- *6_greaterBlocks.txt*: caso de estudio con 2 bloques, uno de 14 artículos y otro de 16.

A continuación se muestra un ejemplo de la salida en pantalla de *ejecutar.sh*, pero sin incluir los ficheros 5 y 6 ya que el algoritmo de ramificación y poda no es capaz de soportar sus dimensiones:

```
##### Ejecutando pruebas con 1_prueba.txt #####
111600 0.489390
 10 10 0 0
 20 10 25 15
 10 10 10 10
104000 0.348362
 10 20 30 40
 20 30 40 50
 90 80 70 60
##### Ejecutando pruebas con 2_singleArticle.txt #####
111900 0.060714
 10 10 15 15
##### Ejecutando pruebas con 3_moreArticles.txt #####
111575 1.525851
 10 10 0 0
 20 10 25 15
 10 10 10 10
 5 5 15 25
103700 7.461215
 10 20 30 40
 20 10 20 20
 10 10 10 10
 20 30 40 50
 90 80 70 60
##### Ejecutando pruebas con 4_tricky.txt #####
9872 0.164705
 8 8 0 0
 8 8 8 8
9872 0.110984
 8 8 0 0
 8 8 8 8
```

A su vez, como bien se indica en la tarea 3, se ha realizado una experimentación comparativa con las soluciones de las otras prácticas. Los resultados obtenidos han sido los siguientes:

Tests adicionales:

```
Block: n: 9, W: 200, H: 200, articles:
Article with values--> w: 10, h: 10, x: 30, y: 30
Article with values--> w: 10, h: 10, x: 0, y: 0
Article with values--> w: 10, h: 10, x: 30, y: 30
Article with values--> w: 10, h: 10, x: 45, y: 45
```

Article with values--> w: 10, h: 10, x: 60, y: 60
Article with values--> w: 10, h: 10, x: 5, y: 5
Article with values--> w: 10, h: 10, x: 75, y: 75
Article with values--> w: 10, h: 10, x: 15, y: 15
Article with values--> w: 10, h: 10, x: 100, y: 100

Backtracking: 33.512044ms
Iterative dp: 7.054564ms
Recursive dp: 1.564157ms
Branch&Bound: 741.727445ms
Brute force: 5.218095ms
Nodes generated in backtracking: 4859
Nodes generated in iterative dp: 508
Nodes generated in recursive dp: 447
Nodes generated in Branch&Bound: 7
Nodes generated in brute force: 511

Block: n: 5, W: 280, H: 400, articles:
Article with values--> w: 10, h: 10, x: 0, y: 0
Article with values--> w: 20, h: 10, x: 20, y: 20
Article with values--> w: 20, h: 10, x: 25, y: 15
Article with values--> w: 10, h: 10, x: 10, y: 10
Article with values--> w: 10, h: 10, x: 15, y: 15

Backtracking: 0.174182ms
Iterative dp: 0.197285ms
Recursive dp: 0.078340ms
Branch&Bound: 0.336259ms
Brute force: 0.096936ms
Nodes generated in backtracking: 28
Nodes generated in iterative dp: 28
Nodes generated in recursive dp: 23
Nodes generated in Branch&Bound: 3
Nodes generated in brute force: 31

Block: n: 6, W: 280, H: 400, articles:
Article with values--> w: 90, h: 80, x: 70, y: 60
Article with values--> w: 20, h: 30, x: 40, y: 50
Article with values--> w: 50, h: 50, x: 40, y: 40
Article with values--> w: 80, h: 70, x: 60, y: 50
Article with values--> w: 50, h: 60, x: 70, y: 80
Article with values--> w: 10, h: 20, x: 30, y: 40

Backtracking: 0.101601ms
Iterative dp: 0.240285ms
Recursive dp: 0.047707ms
Branch&Bound: 0.294322ms
Brute force: 0.168755ms
Nodes generated in backtracking: 33
Nodes generated in iterative dp: 55
Nodes generated in recursive dp: 37
Nodes generated in Branch&Bound: 3
Nodes generated in brute force: 63

Block: n: 1, W: 280, H: 400, articles:
Article with values--> w: 10, h: 10, x: 15, y: 15

Backtracking: 0.006501ms
Iterative dp: 0.006821ms
Recursive dp: 0.004563ms
Branch&Bound: 0.026096ms
Brute force: 0.005489ms
Nodes generated in backtracking: 1
Nodes generated in iterative dp: 1
Nodes generated in recursive dp: 1
Nodes generated in Branch&Bound: 1
Nodes generated in brute force: 1

Block: n: 7, W: 280, H: 400, articles:
Article with values--> w: 20, h: 10, x: 25, y: 15
Article with values--> w: 20, h: 10, x: 20, y: 20
Article with values--> w: 10, h: 10, x: 0, y: 0
Article with values--> w: 5, h: 5, x: 15, y: 25
Article with values--> w: 10, h: 10, x: 15, y: 15
Article with values--> w: 7, h: 7, x: 12, y: 12
Article with values--> w: 10, h: 10, x: 10, y: 10

Backtracking: 0.451747ms
Iterative dp: 0.524089ms
Recursive dp: 0.100775ms
Branch&Bound: 2.016551ms
Brute force: 0.318433ms
Nodes generated in backtracking: 134
Nodes generated in iterative dp: 121
Nodes generated in recursive dp: 75
Nodes generated in Branch&Bound: 4
Nodes generated in brute force: 127

Block: n: 9, W: 280, H: 400, articles:
Article with values--> w: 90, h: 80, x: 70, y: 60
Article with values--> w: 80, h: 70, x: 60, y: 50
Article with values--> w: 10, h: 10, x: 10, y: 10
Article with values--> w: 20, h: 10, x: 20, y: 20
Article with values--> w: 20, h: 30, x: 40, y: 50
Article with values--> w: 10, h: 10, x: 15, y: 15
Article with values--> w: 50, h: 50, x: 40, y: 40
Article with values--> w: 50, h: 60, x: 70, y: 80
Article with values--> w: 10, h: 20, x: 30, y: 40

Backtracking: 2.501622ms
Iterative dp: 2.545225ms
Recursive dp: 0.252977ms
Branch&Bound: 10.242432ms
Brute force: 1.065415ms
Nodes generated in backtracking: 581
Nodes generated in iterative dp: 485
Nodes generated in recursive dp: 191
Nodes generated in Branch&Bound: 5
Nodes generated in brute force: 511

Block: n: 3, W: 100, H: 100, articles:
Article with values--> w: 8, h: 8, x: 8, y: 8
Article with values--> w: 10, h: 10, x: 0, y: 0
Article with values--> w: 8, h: 8, x: 0, y: 0

Backtracking: 0.031562ms
Iterative dp: 0.025931ms
Recursive dp: 0.010563ms
Branch&Bound: 0.044393ms
Brute force: 0.016916ms
Nodes generated in backtracking: 6
Nodes generated in iterative dp: 5
Nodes generated in recursive dp: 7
Nodes generated in Branch&Bound: 2
Nodes generated in brute force: 7

Block: n: 3, W: 100, H: 100, articles:
Article with values--> w: 10, h: 10, x: 7, y: 7
Article with values--> w: 8, h: 8, x: 0, y: 0
Article with values--> w: 8, h: 8, x: 8, y: 8

Backtracking: 0.017074ms
Iterative dp: 0.017950ms
Recursive dp: 0.011067ms

Branch&Bound:	0.043005ms	
Brute force:	0.015278ms	
Nodes generated in backtracking:		6
Nodes generated in iterative dp:		5
Nodes generated in recursive dp:		7
Nodes generated in Branch&Bound:		2
Nodes generated in brute force:		7

BOLA EXTRA

En el fichero buscaLP.py se ha implementado un algoritmo de programación lineal para este mismo problema de los artículos. En él se detalla en la cabecera la formalización del problema, y posteriormente se procede a su implementación.