

# ADDA - Practica Individual 2

## Problema 3

Juan Arteaga Carmona  
(juan.arteaga41567@gmail.com)  
2º Curso - TI-2

23 de mayo de 2018

### 1. Completar la ficha de descripción del problema

- Tipos:
  - S - SolucionAlimentos
  - A - Integer  $\in [0,1000]$  (El numero de gramos que se seleccionan de cada ingrediente)
- Propiedades compartidas:
  - nutrientes - List<Nutriente>- Lista con los ingredientes del problema
  - ingredientesActivos - List<IngredienteActivo>- Lista con los ingredientes activos del problema
- Solucion:

Determinar cuantos gramos de cada ingrediente activo es necesario incluir por kilo de alimento para minimizar costes, teniendo en cuenta que cada kilo de alimento debe contener una cantidad mínima de nutrientes concreta.
- Propiedades:
- Restricciones:
- Solución óptima:

## 2. Resolver el problema por PD, para ello:

- 2.1. Completar la ficha de descripción de la solución mediante programación dinámica.
- 2.2. Escriba un archivo denominado 'alimentos.txt' con los datos del escenario de entrada de forma similar a como se ha realizado en las clases de prácticas para otros problemas.

A continuación se incluye el archivo con los datos iniciales del problema propuesto en el enunciado de la práctica individual. Cabe destacar que todas las cantidades Mínimas- PorKilo de cada nutriente se escriben todas en una línea y que la cantidad Nutrientes y coste se escriben en una línea distinta por cada ingrediente activo.

```
#La cantidad minima de cada nutriente en esta linea#  
Nutrientes,90,50,20,2
```

```
#Cada ingrediente activo en una linea distinta#  
IngredienteActivo,0.1-0.08-0.04-0.01,0.04  
IngredienteActivo,0.2-0.15-0.02-0,0.06
```

- 2.3. Desarrolle un proyecto que resuelva el problema especificado por la técnica sin hacer uso en principio de una función de cota. Dicho proyecto debe incluir un test de prueba que genere la solución para el escenario previamente descrito en el enunciado.
- 2.4. El test de prueba debe generar el archivo "GrafoAlimentos.gv"(grafo and/or relacionado de la búsqueda llevada a cabo) que debe entregar incluido en el proyecto y en la memoria.
- 2.5. Modifique la solución anterior para realizar una función de cota adecuada al problema a resolver. Ejecute de nuevo el test de prueba sobre el proyecto modificado e indique qué solución obtiene para el problema del escenario indicado previamente en el enunciado. el test de prueba debe generar el archivo "GrafoAlimentosFiltro.gv"que debe entregar incluido en el proyecto y en la memoria.

## 3. Resolver el problema mediante BT, para ello:

- 3.1. Completar la ficha de descripción de la solución mediante BT.

- Problema:
  - Problema de los alimentos

- Descripción:
  - Determinar cuantos gramos de cada ingrediente activo es necesario incluir por kilo de alimento para minimizar costes, teniendo en cuenta que cada kilo de alimento debe contener una cantidad mínima de nutrientes concreta.
- Técnica:
  - Vuelta Atrás
- Tipos:
  - S - SolucionAlimentos
  - A - Integer  $\in [0,1000]$  (El numero de gramos que se seleccionan de cada ingrediente)
- Propiedades compartidas:
  - nutrientes - List<Nutriente>- Lista con los ingredientes del problema
  - ingredientesActivos - List<IngredienteActivo>- Lista con los ingredientes activos del problema
  - numeroIngredientes - Derivada - ingredientesActivos.size()
- Propiedades individuales del estado:
  - index - Integer - Indice indicador del alimento por el que vamos
  - costeIngrediente - Derivada - ingredientesActivos.get(index).getCoste()
- Solución:
  - List<Integer>s (Lista de integers que indica la cantidad de gramos que se han seleccionado de cada ingrediente)
- Tamaño:
  - numeroIngredientes - index
- Alternativas:
  - A = a  $\in [0,1000]$
- Instanciación:
  - (0,0)
- Estado final:
 

index == numeroIngredientes  
Devuelve true si hemos llegado al último ingrediente
- Avanza:
 

(index, coste) -> (index+1, coste+a\*costeIngrediente)
- Retrocede:
 

(index, coste) -> (index-1, coste-a\*costeIngrediente)
- Solución (para calcular el estado final):
  - $s() = sr(0, null)$
  - $sr(i, coste) = \begin{cases} coste & \text{index} = \text{numeroIngredientes} \\ sr(index + 1, coste + a * costeIngrediente) & \text{index} < \text{numeroIngredientes} \end{cases}$
- Objetivo:
  - Min coste

- 3.2. Desarrolle un proyecto que resuelva el problema especificado por técnica indicada sin hacer uso en principio de una función de cota. Dicho proyecto debe incluir un test de prueba que genere la solución para el escenario previamente descrito en el enunciado.
- 3.3. Modifique la solución anterior para realizar una función de cota adecuada al problema a resolver. Ejecute de nuevo el test de prueba sobre el proyecto modificado e indique qué solución obtiene para el problema del escenario indicado previamente en el enunciado.

## 4. Anexos

- 4.1. Código completo
- 4.2. Volcado de pantalla de los resultados obtenidos por cada prueba realizada
- 4.3. Código fuente y licencia

Todo el código fuente del trabajo se podrá encontrar en [www.github.com/Andalu30/US-ADDA-PracticaIndividual2/](http://www.github.com/Andalu30/US-ADDA-PracticaIndividual2/) a partir del lunes 4 de junio de 2018, fecha límite de entrega de este trabajo. No se recomienda el uso de la totalidad o de parte de este trabajo en caso de que no se cambie la actividad en años posteriores debido a que la universidad usa software de detección de copias. El contenido de este trabajo es libre y se encuentra licenciado bajo una licencia GPL v3.

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see [<http://www.gnu.org/licenses/>](http://www.gnu.org/licenses/).