



MEMORIA ENTREGA 3

Universidad Carlos III
Grado Ingeniería Informática 2022-23
Desarrollo Software (Grupo 81)

Práctica realizada por:

- **Jaime Vaquero Rabahieh** (NIA: 100472248, Email: 100472248@alumnos.uc3m.es)
- **Alejandro Díaz Cuéllar** (NIA: 100472173, Email: 100472173@alumnos.uc3m.es)

ÍNDICE

1.- EJERCICIO 1	3
1.1 - Casos de prueba: clases de equivalencia	3
1.2- Casos de prueba: valores límite	4
2.- EJERCICIO 2	5
2.1 - Gramática	5
2.2 - Árbol de derivación	6
2.3 - Clases de equivalencia y valores límite	6
3.- EJERCICIO 3	7
3.1- Diagrama de control de flujo	7
3.2 - Rutas básicas	8
3.3 - Casos de prueba de la función	8

1.- EJERCICIO 1

1.1 - Casos de prueba: clases de equivalencia

Hemos listado todas estas clases organizadas por su tipo:

-Product-id:

- Válido 1: Product-Id correcto.
- No Válido 1: ID con más de 13 cifras.
- No Válido 2: ID con menos de 13 cifras.
- No Válido 3: ID que no tiene números.
- No Válido 4: ID que su última cifra no cumple el estándar EAN-13.

-Order-type:

- Válido 2: Pedido REGULAR correcto.
- Válido 3: Pedido PREMIUM correcto
- No Válido 5: Order-type que no es STRING.
- No Válido 6: Order-type que no es ni REGULAR ni PREMIUM

-Address:

- Válido 4: Address correcta.
- No Válido 7: Address de menos de 20 caracteres.
- No Válido 8: Address de más de 20 caracteres.
- No Válido 9: Address que no es string.
- No Válido 10: Address con menos de un espacio.

-Phone_number:

- Válido 5: Teléfono válido.
- No Válido 11: Teléfono con más de 9 cifras.
- No Válido 12: Teléfono con menos de 9 cifras.
- No Válido 13: Teléfono que no tiene números.

-Zip_code:

- Válido 6: Zip correcto.
- No Válido 14: Zip con más de 5 cifras.
- No Válido 15: Zip con menos de 5 cifras.
- No Válido 16: Zip que no tiene números.
- No Válido 17: Zip con las dos primeras cifras fuera del rango 01-52.

-Salida MD5 hexadecimal:

- Válido 7: Se devuelve una cadena hexadecimal MD5.
- No Válido 18: No genera esta cadena.

-Excepciones:

- Válido 8: No salen excepciones.
- No Válido 19: Sale una excepción del ID .
- No Válido 20: Sale una excepción del Order-type.
- No Válido 21: Sale una excepción del address.
- No Válido 22: Sale una excepción del teléfono.
- No Válido 23: Sale una excepción del zip.

-Ficheros:

- Válido 9: Fichero creado.
- No Válido 24: Fichero no creado.

Hemos simplificado las clases que se solapan y nos salen en total unos 2 tests de clases válidas (Uno por tipo de order) y 17 inválidas. Las relaciones de solapamiento las hemos representado en el excel.

1.2- Casos de prueba: valores límite

Hemos listado los valores límite por su tipo:

-Product-id:

- VL 1: ID de 12 cifras.
- VL 2: ID de 13 cifras.
- VL 3: ID de 14 cifras.

-Address:

- VL 4: Address de 19 caracteres.
- VL 5: Address de 20 caracteres.
- VL 6: Address de 21 caracteres.
- VL 7: Address de 99 caracteres.
- VL 8: Address de 100 caracteres.
- VL 9: Address de 101 caracteres.

-Phone_number:

- VL 10: Teléfono de 8 cifras.
- VL 11: Teléfono de 9 cifras.
- VL 12: Teléfono de 10 cifras.

-Zip_code:

- VL 13: Zip de 4 cifras.
- VL 14: Zip de 5 cifras.
- VL 15: Zip de 6 cifras.

Como con las clases, hemos simplificado las que se solapan con las de las clases y nos salen en total unos 4 tests (para VL5, VL6, VL7 y VL8).

2.- EJERCICIO 2

2.1 - Gramática

La gramática aborda este texto:

```
{“OrderID”:"caf7eace516dced5512b338105303c83",  
  “ContactEmail”:"ejercicioguiado3@gmail.com"}
```

La definición formal de la gramática es:

Nivel 1: Fichero ::= LlaveIz Datos LlaveDr

Nivel 2: LlaveIz ::= {
 LlaveDr ::= }
 Datos ::= Campo1 Coma Campo2

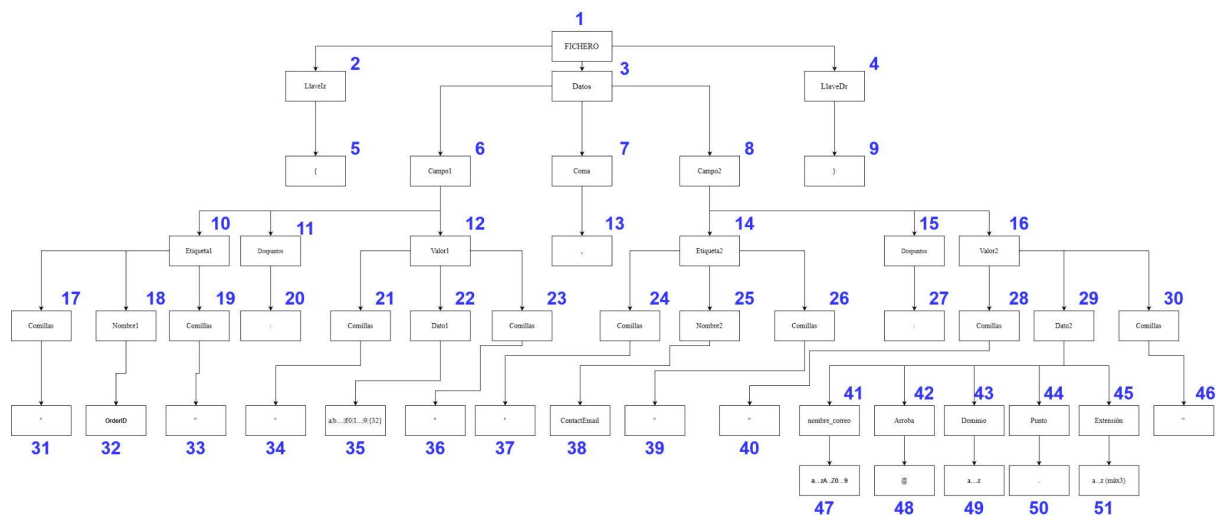
Nivel 3: Coma ::= ,
 Campo1 ::= Etiqueta1 Igual Valor1
 Campo2 ::= Etiqueta2 Igual Valor2

Nivel 4: Dospuntos ::= :
 Etiqueta1 ::= Comillas Nombre1 Comillas
 Etiqueta2 ::= Comillas Nombre2 Comillas
 Valor1 ::= Comillas Dato1 Comillas
 Valor2 ::= Comillas Dato2 Comillas

Nivel 5: Comillas ::= “”
 Nombre1 ::= OrderID
 Nombre2 ::= ContactEmail
 Dato1 ::= a|b....|f|0|1....|9| {32}
 Dato2 ::= Nombre_correo Arroba Dominio Punto Extensión

Nivel 6: Nombre_correo ::= a..zA..Z0..9
 Arroba ::= @
 Dominio ::= a..z
 Punto ::= .
 Extensión ::= a..z (máximo 3)

2.2 - Árbol de derivación



Como podemos observar, hemos creado un total de 51 nodos: 16 terminales y el resto no terminales. En total nos salen un total de 50 tests o casos de prueba distintos debido a que hay numerosos nodos que se repiten varias veces (como en el caso de las comillas o los dos puntos). Además bastantes de los nodos contienen la misma estructura de árbol, por lo que esos no hace falta comprobarlos igualmente.

Para los que si hace falta, se han realizado las pruebas de eliminación y duplicación de nodos para los no terminales y de modificación para los terminales (puesto que la duplicación y eliminación de estos nodos ya se prueba en el nodo no terminal padre).

Cada uno de estos casos de prueba están detallados con mayor profundidad en el excel

2.3 - Clases de equivalencia y valores límite

Para el OrderID, hemos creado estas clases y valores límites:

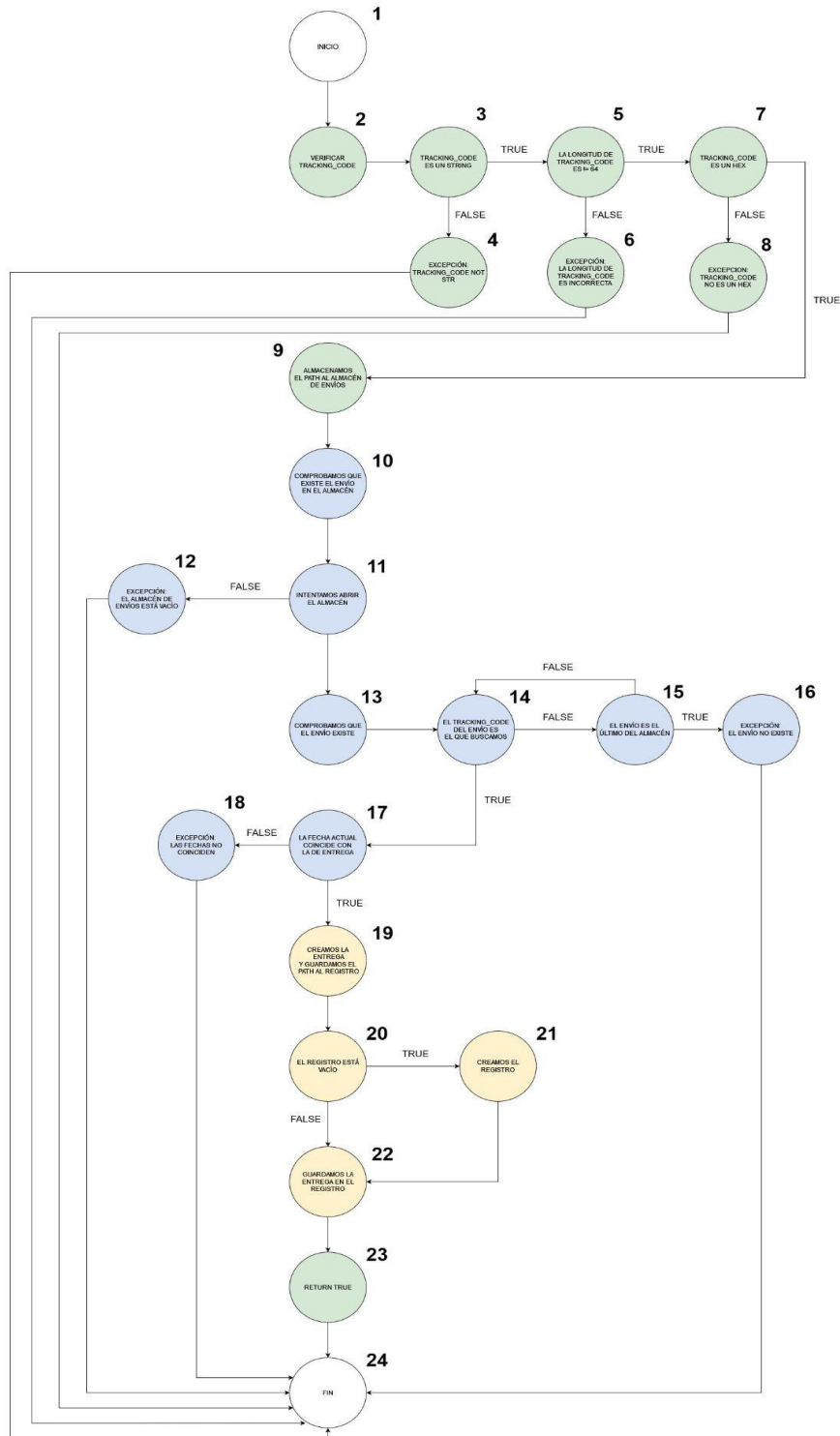
- Válida: OrderID correcto.
- No válida 1: OrderID de más de 32 caracteres.
- No válida 2: OrderID de menos de 32 caracteres.
- No válida 3: OrderID que no es un string.
- No válida 4: OrderID que no cumple el hex.
- VL 1: OrderID de 31 caracteres.
- VL 2: OrderID de 32 caracteres.
- VL 3: OrderID de 33 caracteres.

Debido a los solapamientos con los tests de los nodos, solo nos ha hecho falta representar 3 tests más: para no válida 3, VL1 y VL3.

3.- EJERCICIO 3

3.1- Diagrama de control de flujo

El diagrama de control de flujo de la función RF3 es el siguiente:



(Como la foto insertada no se visualiza correctamente, hay una copia de la imagen en la carpeta docs del github)

Para realizar la función RF3, se llaman a dos métodos de la clase OrderManager, por lo que para representar el flujo del código en esos métodos hemos usado distintos colores para cada uno.

3.2 - Rutas básicas

Existen 8 rutas básicas en nuestro diagrama de flujo:

RUTA BÁSICA	VALOR DE TERMINACIÓN
1-2-3-4	Exception: Tracking_code no es un string
1-2-3-5-6	Exception: Tracking_code no tiene la longitud adecuada
1-2-3-5-7-8	Exception: Tracking_code no es un hex
1-2-3-5-7-9 -10-11-12	Exception: El almacén esta vacío
1-2-3-5-7-9 -10-11-13-14 -15-16	Exception: Envío no encontrado en el almacén
1-2-3-5-7-9 -10-11-13-14 -15-17-18	Exception: La fecha de entrega no es correcta
1-2-3-5-7-9 -10-11-13-14 -15-17-19-20- 21-22-23-24	VERDADERO
1-2-3-5-7-9 -10-11-13-14 -15-17-19-20- 22-23-24	VERDADERO

3.3 - Casos de prueba de la función

Los casos de prueba de esta función los definen los caminos básicos, y habrá uno por cada uno de ellos.

RUTA BÁSICA	CASO DE PRUEBA
1-2-3-4	Tracking_code no es un string

1-2-3-5-6	Tracking_code no tiene la longitud adecuada
1-2-3-5-7-8	Tracking_code no es un hex
1-2-3-5-7-9 -10-11-12	El almacén esta vacío
1-2-3-5-7-9 -10-11-13-14 -15-16	Envío no encontrado en el almacén
1-2-3-5-7-9 -10-11-13-14 -15-17-18	La fecha de entrega no es correcta
1-2-3-5-7-9 -10-11-13-14 -15-17-19-20- 21-22-23-24	Todo válido pero el registro de entregas no existe
1-2-3-5-7-9 -10-11-13-14 -15-17-19-20- 22-23-24	Todo válido y el registro de entregas existe