**Ejercicio 2**

Encontramos 5 entradas y 1 salida.

Agrupamos 2 entradas “fechaInicioCurso”, “fechaFinCurso” y “diaSemana ” ya que dependen entre ellas.

**Entrada 1:** **(E1)** (asignatura)

* Clase valida (V1) el nombre de la asignatura

**Entrada 2: (E2)** (fechaInicioCurso , fechaFinCurso y diaSemana) agrupación ya que las tres dependen mutuamente y un cambio en una afecta a la ejecución del programa (#6)

* Clase valida (V2): fechaInicioCurso menor que fechaFinCurso, diaSemana entre 1 y 6 y diaSemana entre las fechas de inicio y fin
* Clase invalida (N2): fechaInicioCurso mayor que fechaFinCurso
* Clase invalida (N3): día no es un valor entre 1 y 6
* Clase invalida (N4): día no está entre las fechas de inicio y fin

**Entrada 4: (E4)** (horaInicioClase) horario que debe ser de formato hh:mm

* Clase valida (V3): formato hh:mm
* Clase valida (V4): NULL
* Clase invalida (N4): formato o valor no disponible

**Salida:** (numero)

* Clase valida (S1): Lista de eventos
* Clase valida (S2): duración -1
* Clase invalida (NS1): Lista vacía
* Clase invalida (NS2): ParseException

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Camino | Datos de entrada | | | Salida | |
| **HorarioAsignatura** | | |
| **E1** | **E2** | **E3** | |  | |
| V1-V2-V3-S1 | “STR” | 11/1/22 - 15/1/23  3 | “10:00” | | [(“STR”,¿?,“10:00”,120)] | |
| V1-V2-V3 -S2 | “STR” | 11/1/22 - 15/1/23  3 | NULL | | [(“STR”,¿?,“10:00”,120)] | |
| V1-N2-V3-NS1 | “STR” | 11/1/24 - 15/1/23  3 | “10:00” | | [] | |
| V1-N4-V3-NS1 | “STR” | 11/1/22 - 15/1/23  2 | “10:00” | | [] | |
| V1-N3- V3-NS2 | “STR” | 11/1/22 - 15/1/23  0 | “10:00” | | ParseException | |
| V1-V2-N4-NS3 | “STR” | 11/1/22 - 15/1/23  3 | “10:0” | | ParseException | |