**Test de cobertura**

***Integrantes G1:***

***Pablos, Braulio.***

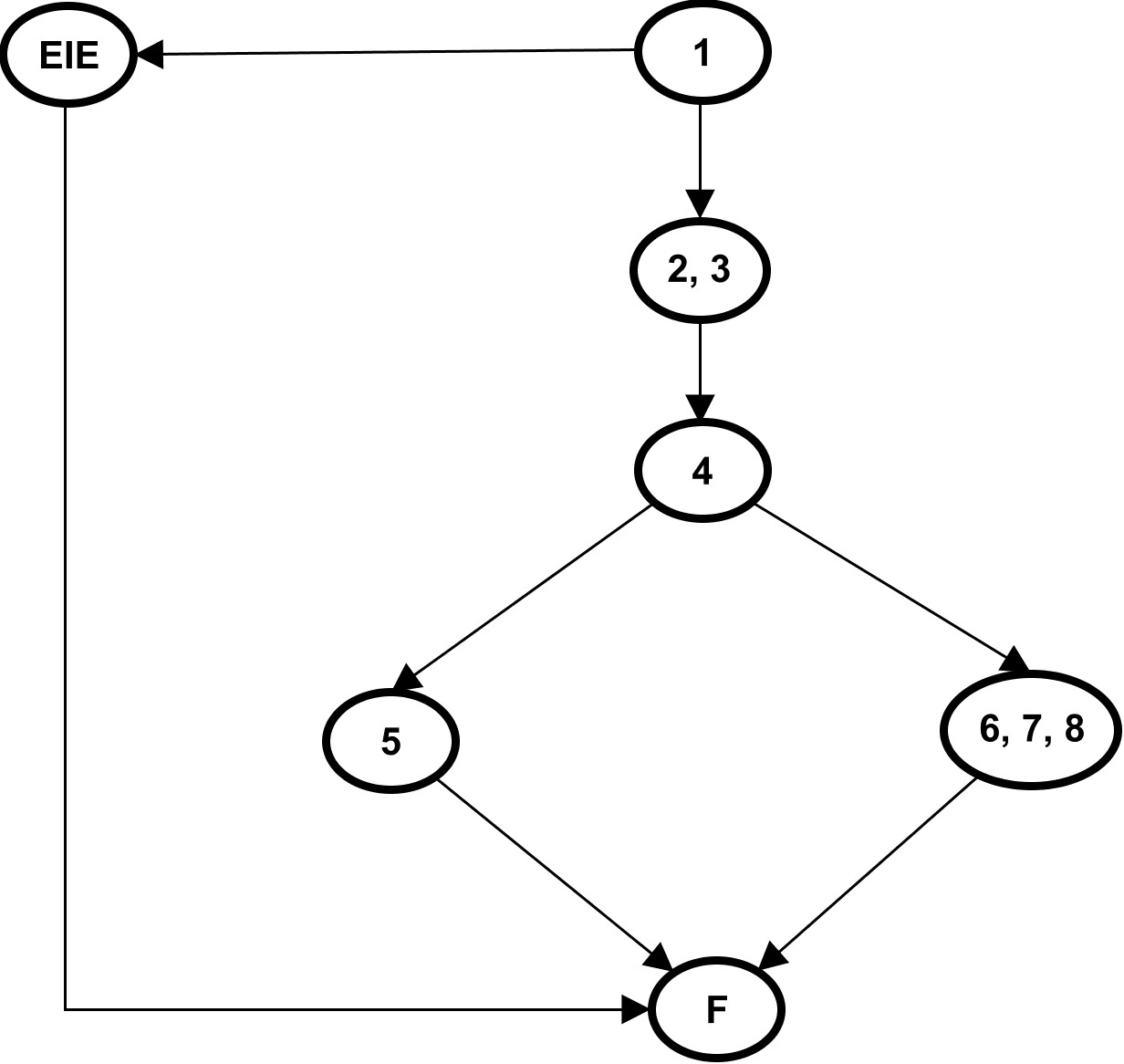
***Mateos, Wenceslao.***

***Grupo testeado G4:***

***Canónaco, Sebastián.***

***Nucci, Manuel.***

**altaAlumno(String nombre, String apellido, String domicilio, String telefono, String mail)**

****

CC = número de arcos (8) – número de nodos (7) + 2 = 3

CC = nodos predicados (2) + 1 = 3

CC = número de regiones (2) + 1 = 3

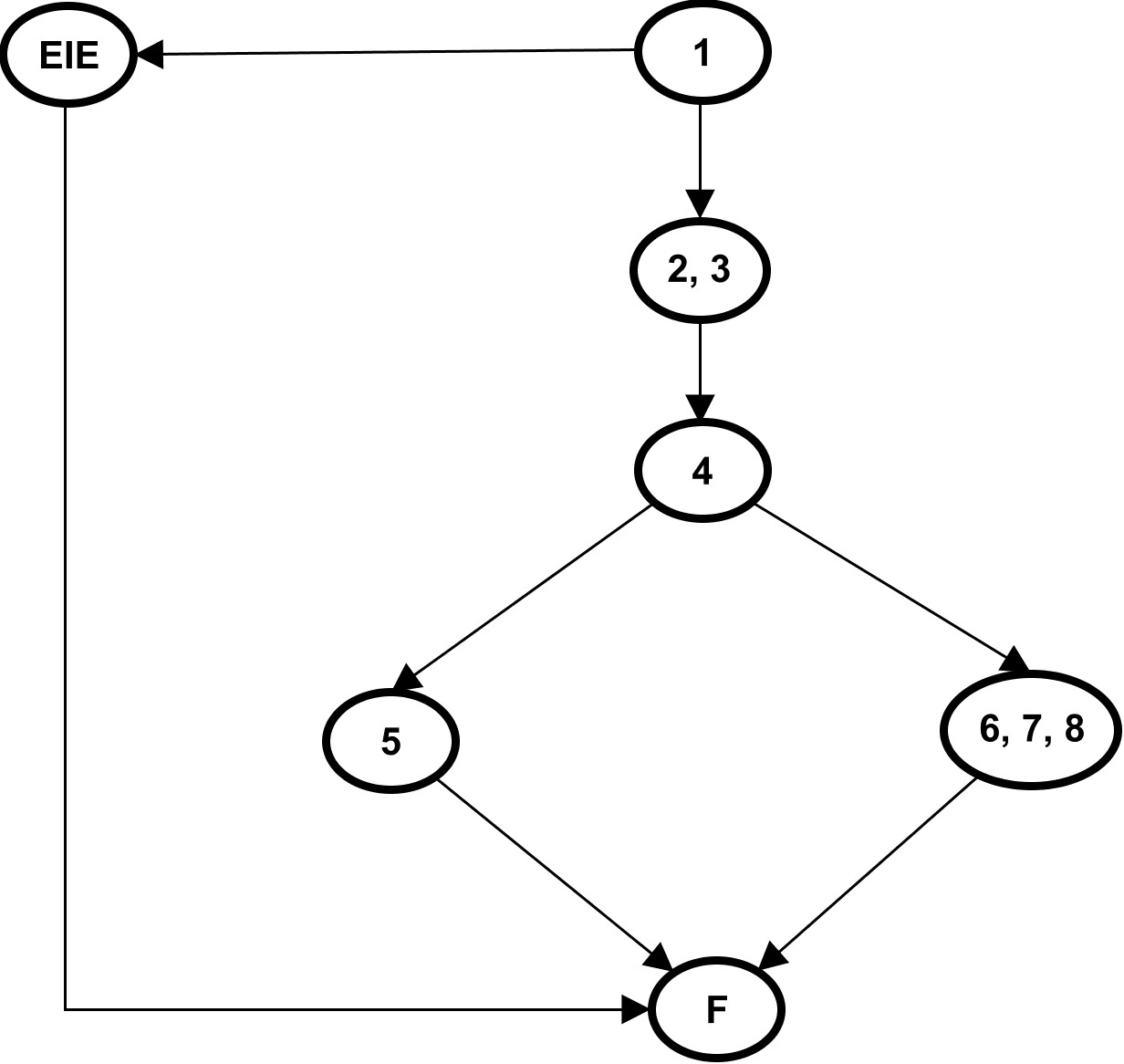
**camino 1: 1, EIE, F**

**camino 2: 1, 2 - 3, 4, 5, F**

**camino 3: 1, 2 - 3, 4, 6 - 7 - 8, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **AAl1** | Verificar camino 1. | Valores de los parámetros nombre, apellido, domicilio, telefono y mail.  Estado de Manager.alumnos. | * Dejar Manager.alumnos vacío pero no nulo. * Invocar con (“Manuel”, “Nucci”, “Jujuy 345”, “123456789”, “mailmal”). | EmailInvalidoException |  |
| **AAl2** | Verificar camino 2. | Valores de los parámetros nombre, apellido, domicilio, telefono y mail.  Estado de Manager.alumnos. | * Agregar un alumno a Manager.alumnos con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (“Manuel”, “Nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | Manager.alumnos tiene un alumno más cuyos atributos coinciden con los parámetros. | PASS |
| **AAl3** | Verificar camino 3. | Valores de los parámetros nombre, apellido, domicilio, telefono y mail.  Estado de Manager.alumnos. | * Dejar Manager.alumnos vacío pero no nulo. * Invocar con (“Manuel”, “Nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | Manager.alumnos tiene un alumno más cuyos atributos coinciden con los parámetros. |  |

**altaProfesor(String nombre, String apellido, String domicilio, String telefono, String mail)**

****

CC = número de arcos (8) – número de nodos (7) + 2 = 3

CC = nodos predicados (2) + 1 = 3

CC = número de regiones (2) + 1 = 3

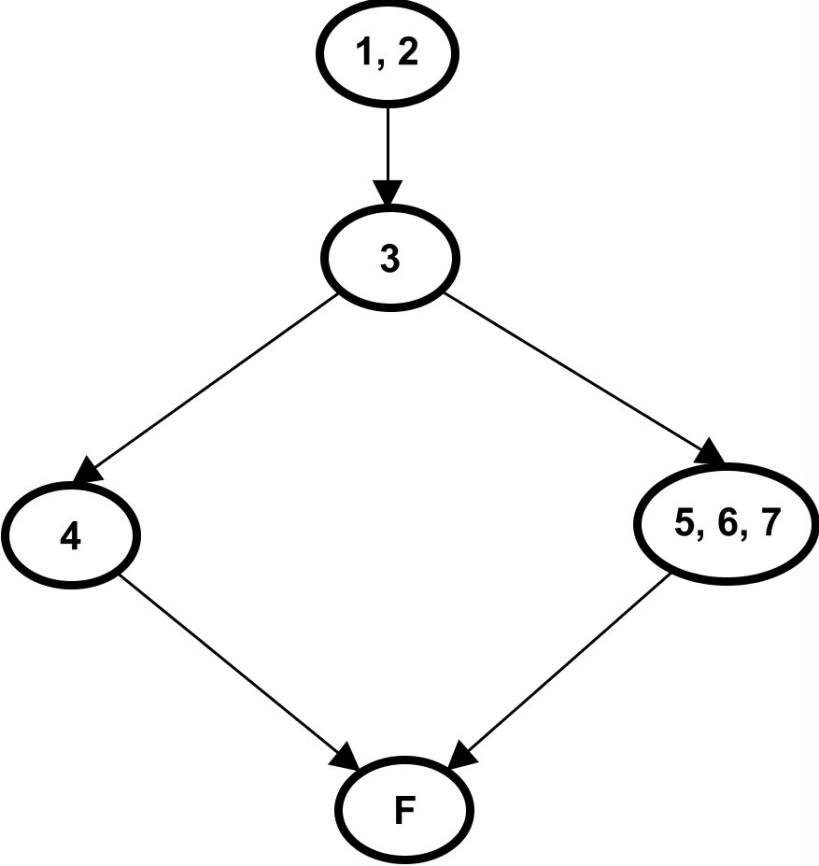
**camino 1: 1, EIE, F**

**camino 2: 1, 2 - 3, 4, 5, F**

**camino 3: 1, 2 - 3, 4, 6 - 7 - 8, F**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| **AP1** | Verificar camino 1. | Valores de los parámetros nombre, apellido, domicilio, telefono y mail.  Estado de Manager.profesores. | * Dejar Manager.profesores vacío pero no nulo. * Invocar con (“Manuel”, “Nucci”, “Jujuy 345”, “123456789”, “mailmal”). | EmailInvalidoException |  |
| **AP2** | Verificar camino 2. | Valores de los parámetros nombre, apellido, domicilio, telefono y mail.  Estado de Manager.profesores. | * Agregar un profesor a Manager.profesores con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (“Manuel”, “Nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | Manager.profesores tiene un profesor más cuyos atributos coinciden con los parámetros. |  |
| **AP3** | Verificar camino 3. | Valores de los parámetros nombre, apellido, domicilio, telefono y mail.  Estado de Manager.profesores. | * Dejar Manager.profesores vacío pero no nulo. * Invocar con (“Manuel”, “Nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | Manager.profesores tiene un profesor más cuyos atributos coinciden con los parámetros. |  |

**altaAsignatura(String nombre)**

****

CC = número de arcos (5) – número de nodos (5) + 2 = 2

CC = nodos predicados (1) + 1 = 2

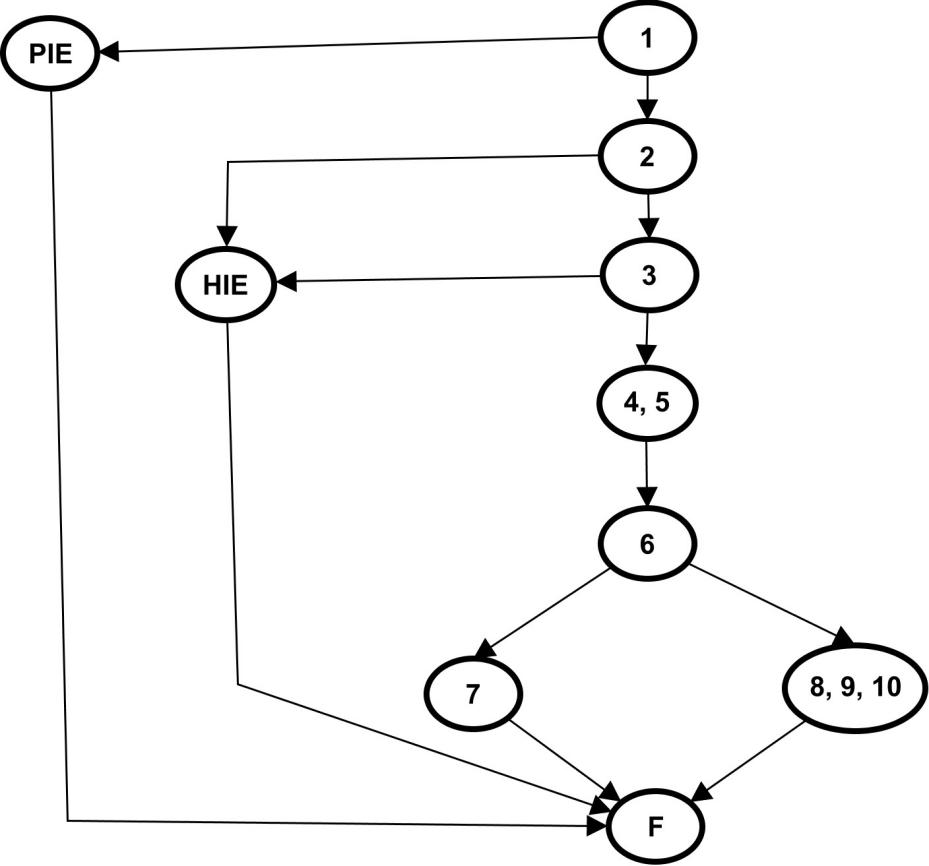
CC = número de regiones (1) + 1 = 2

**camino 1: 1 - 2, 3, 4, F**

**camino 2: 1 - 2, 3, 5 - 6 - 7, F**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| **AAs1** | Verificar camino 1. | Valor del parámetro nombre.  Estado de Manager.asignaturas. | * Agregar una asignatura a Manager.asignaturas cuyo nombre sea “Programación”. * Invocar con (“Programación”). | Manager.asignaturas tiene una asignatura más cuyo nombre coincide con el parámetro. | PASS |
| **AAs2** | Verificar camino 2. | Valor del parámetro nombre.  Estado de Manager.asignaturas. | * Dejar Manager.asignaturas vacío pero no nulo. * Invocar con (“Programaciónl”) | Manager.asignaturas tiene una asignatura más cuyo nombre coincide con el parámetro. |  |

**altaCursada(String nombre, Asignatura asignatura, String periodo, String dia, String horaInicio, String horaFin)**

****

CC = número de arcos (13) – número de nodos (10) + 2 = 5

CC = nodos predicados (4) + 1 = 5

CC = número de regiones (4) + 1 = 5

**camino 1: 1, PIE, F**

**camino 2: 1, 2, HIE, F**

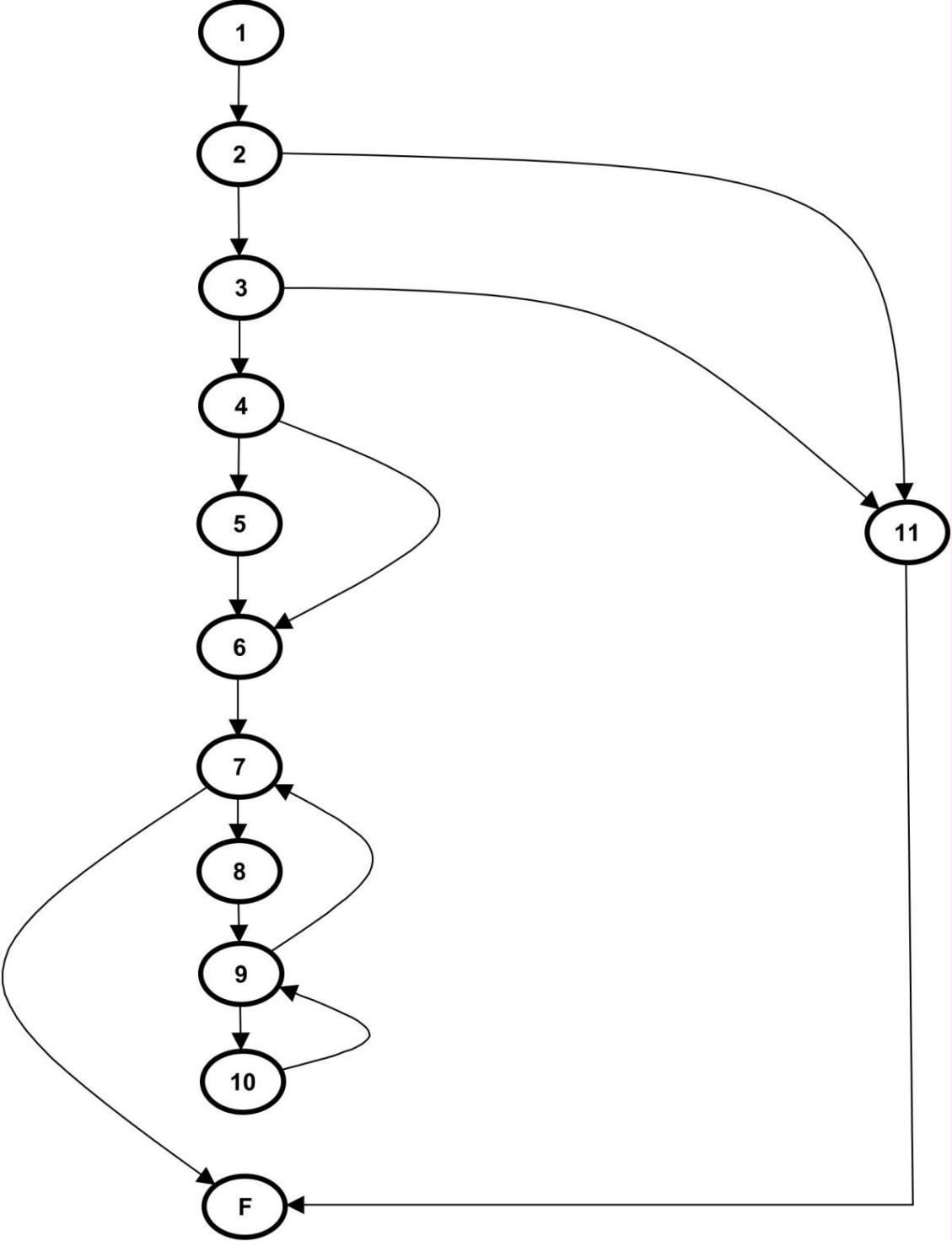
**camino 3: 1, 2, 3, HIE, F**

**camino 4: 1, 2, 3, 4 - 5, 6, 7, F**

**camino 5: 1, 2, 3, 4 - 5, 6, 8 - 9 - 10, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **AC1** | Verificar camino 1. | Valor de los parámetros nombre, asignatura, periodo, dia, horaInicio y horaFin.  Estado de Manager.cursadas. | * Dejar Manager.cursadas vacío pero no nulo. * Crear una asignatura ASIde nombre “Programación”. * Invocar con (“Mi cursada”, ASI, “bla”, “Lunes”, “12:00”, “14:00”). | PeriodoInvalidoException |  |
| **AC2** | Verificar camino 2. | Valor de los parámetros nombre, asignatura, periodo, dia, horaInicio y horaFin.  Estado de Manager.cursadas. | * Dejar Manager.cursadas vacío pero no nulo. * Crear una asignatura ASIde nombre “Programación”. * Invocar con (“Mi cursada”, ASI, “02-2017”, “Lunes”, “bla”, “14:00”). | HoraInvalidaException |  |
| **AC3** | Verificar camino 3. | Valor de los parámetros nombre, asignatura, periodo, dia, horaInicio y horaFin.  Estado de Manager.cursadas. | * Dejar Manager.cursadas vacío pero no nulo. * Crear una asignatura ASIde nombre “Programación”. * Invocar con (“Mi cursada”, ASI, “02-2017”, “Lunes”, “12:00”, “bla”). | HoraInvalidaException |  |
| **AC4** | Verificar camino 4. | Valor de los parámetros nombre, asignatura, periodo, dia, horaInicio y horaFin.  Estado de Manager.cursadas. | * Crear una asignatura ASIde nombre “Programación”. * Agregar a Manager.cursadas una cursada con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (“Mi cursada”, ASI, “02-2017”, “Lunes”, “12:00”, “14:00”). | Manager.cursadas tiene un profesor más cuyos atributos coinciden con los parámetros. | PASS |
| **AC5** | Verificar camino 5. | Valor de los parámetros nombre, asignatura, periodo, dia, horaInicio y horaFin.  Estado de Manager.cursadas. | * Dejar Manager.cursadas vacío pero no nulo. * Crear una asignatura ASIde nombre “Programación”. * Invocar con (“Mi cursada”, ASI, “02-2017”, “Lunes”, “12:00”, “14:00”). | Manager.cursadas tiene un profesor más cuyos atributos coinciden con los parámetros. |  |

**bajaAlumno(Alumno alumno)**

****

CC = número de arcos (16) – número de nodos (12) + 2 = 6

CC = nodos predicados (5) + 1 = 6

CC = número de regiones (5) + 1 = 6

**camino 1: 1, 2, 11, F**

**camino 2: 1, 2, 3, 11, F**

**camino 3: 1, 2, 3, 4, 6, 7, F**

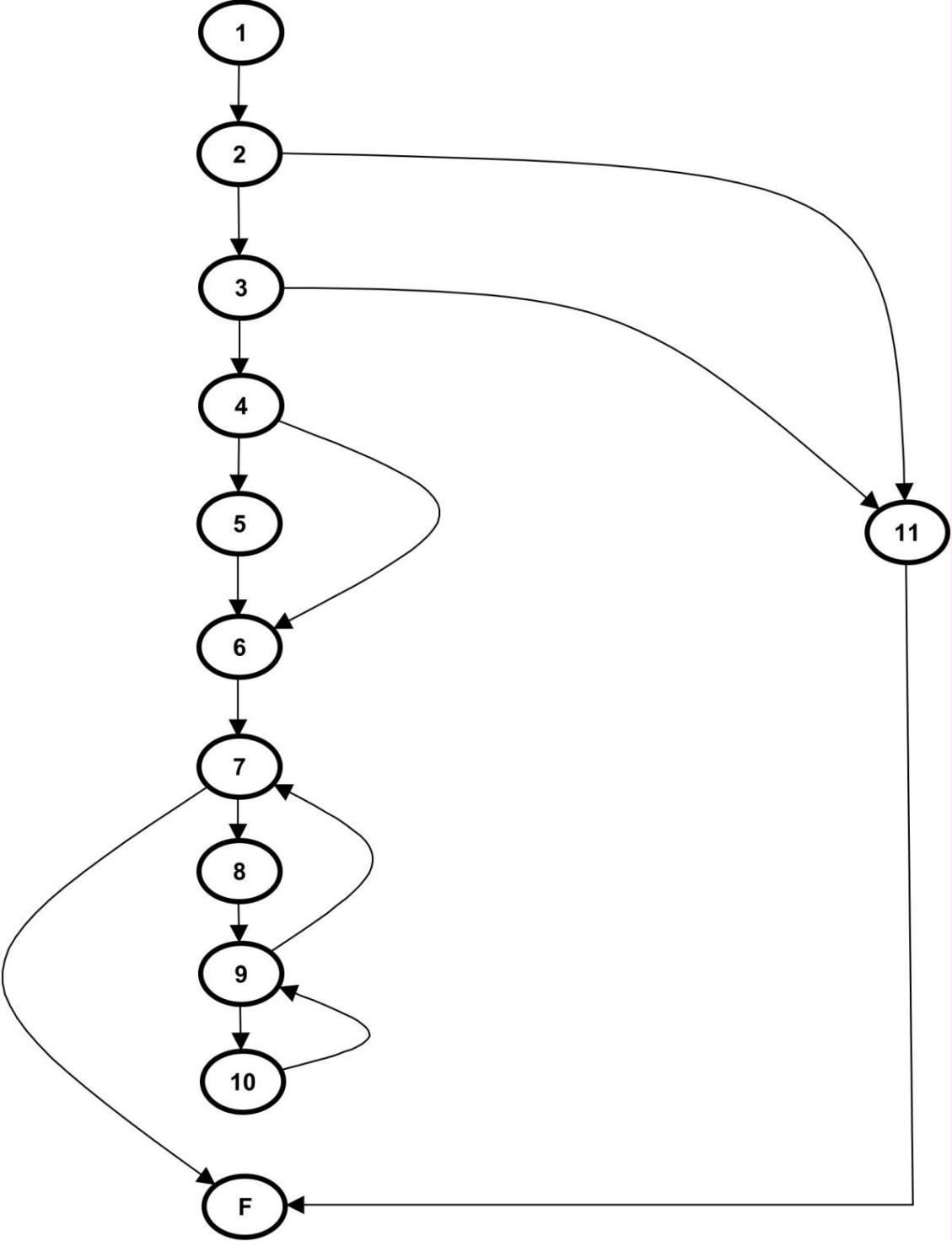
**camino 4: 1, 2, 3, 4, 5, 6, 7, F**

**camino 5: 1, 2, 3, 4, 6, 7, 8, 9, 7, F**

**camino 6: 1, 2, 3, 4, 6, 7, 8, 9, 10, 9, 7, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **BAl1** | Verificar camino 1. | Estado del parámetro alumno.  Estados de Manager.alumnos y Manager.cursadas. | * Dejar Manager.alumnos y Manager.cursada vacíos pero no nulos. * Crear un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. No incorporarlo a Manager.alumnos. * Invocar con (ALU). | NoEstaEntidadException | PASS |
| **BAl2** | Verificar camino 2. | Estado del parámetro alumno.  Estados de Manager.alumnos y Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Agregar a Manager.alumnos un alumno con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Otro 2345”, telefono = “123456789” y mail = “otro@gmail.com”. * Crear un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. No incorporarlo a Manager.alumnos. * Invocar con (ALU). | NoEstaEntidadException | PASS |
| **BAl3** | Verificar camino 3. | Estado del parámetro alumno.  Estados de Manager.alumnos y Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Agregar a Manager.alumnos un alumno con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Otro 2345”, telefono = “123456789” y mail = “otro@gmail.com”. * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (ALU). | Manager.alumnos tiene un alumno menos.  Ya no existe ALU en dicha colección. | PASS |
| **BAl4** | Verificar camino 4. | Estado del parámetro alumno.  Estados de Manager.alumnos y Manager.cursadas. | * Dejar Manager.alumnos y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (ALU). | Manager.alumnos tiene un alumno menos.  Ya no existe ALU en dicha colección. |  |
| **BAl5** | Verificar camino 5. | Estado del parámetro alumno.  Estados de Manager.alumnos y Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una cubeta vacía dentro Manager.cursadas de clave “Mi cursada”. * Invocar con (ALU). | Manager.alumnos tiene un alumno menos.  Ya no existe ALU en dicha colección. |  |
| **BAl6** | Verificar camino 6. | Estado del parámetro alumno.  Estados de Manager.alumnos y Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una asignatura ASIde nombre “Programación”. * Agregar a Manager.cursadas una cursada con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. Incorporar en esta cursada a ALU. * Invocar con (ALU). | Manager.alumnos tiene un alumno menos.  Ya no existe ALU en dicha colección.  ALU ya no figura en ninguna cursada. |  |

**bajaProfesor(Profesor profesor)**

****

CC = número de arcos (16) – número de nodos (12) + 2 = 6

CC = nodos predicados (5) + 1 = 6

CC = número de regiones (5) + 1 = 6

**camino 1: 1, 2, 11, F**

**camino 2: 1, 2, 3, 11, F**

**camino 3: 1, 2, 3, 4, 6, 7, F**

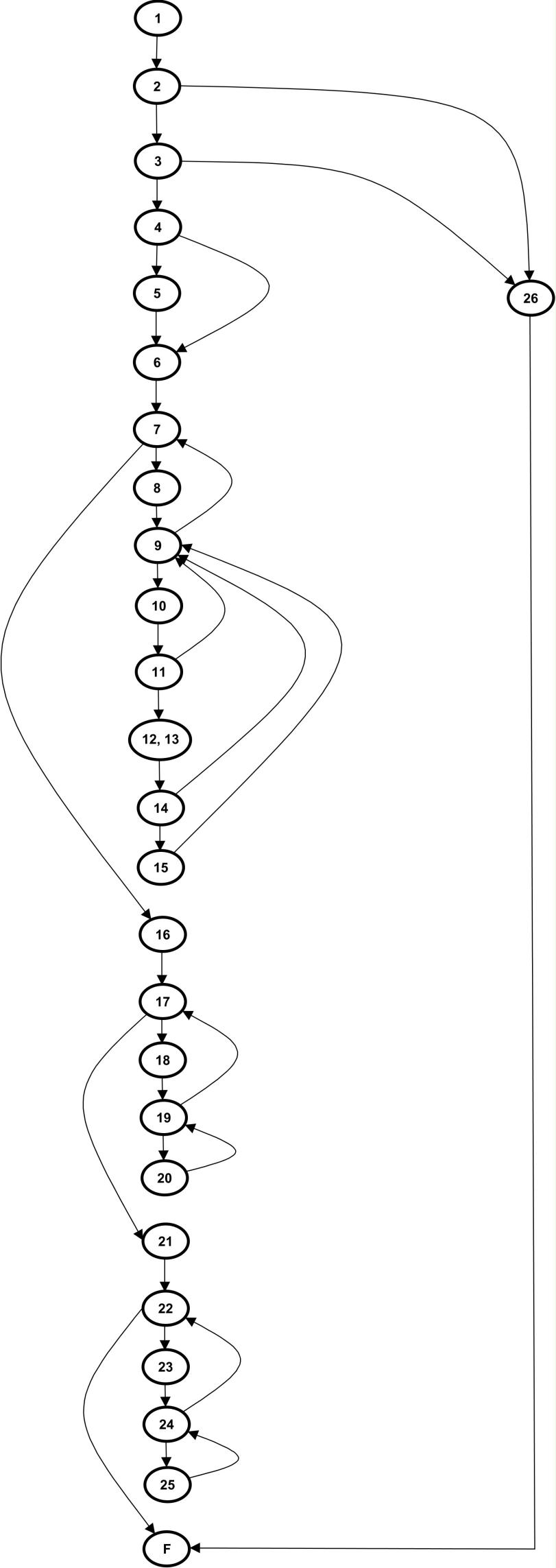
**camino 4: 1, 2, 3, 4, 5, 6, 7, F**

**camino 5: 1, 2, 3, 4, 6, 7, 8, 9, 7, F**

**camino 6: 1, 2, 3, 4, 6, 7, 8, 9, 10, 9, 7, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **BP1** | Verificar camino 1. | Estado del parámetro profesor.  Estados de Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Crear un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. No incorporarlo a Manager.profesores. * Invocar con (PRO). | NoEstaEntidadException | PASS |
| **BP2** | Verificar camino 2. | Estado del parámetro profesor.  Estados de Manager.profesores y Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Agregar a Manager.profesores un alumno con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Otro 2345”, telefono = “123456789” y mail = “otro@gmail.com”. * Crear un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. No incorporarlo a Manager.profesores. * Invocar con (PRO). | NoEstaEntidadException | PASS |
| **BP3** | Verificar camino 3. | Estado del parámetro profesor.  Estados de Manager.profesores y Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Agregar a Manager.profesores un alumno con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Otro 2345”, telefono = “123456789” y mail = “otro@gmail.com”. * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (PRO). | Manager.profesores tiene un profesor menos.  Ya no existe PRO en dicha colección. | PASS |
| **BP4** | Verificar camino 4. | Estado del parámetro profesor.  Estados de Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (PRO). | Manager.profesores tiene un profesor menos.  Ya no existe PRO en dicha colección. |  |
| **BP5** | Verificar camino 5. | Estado del parámetro profesor.  Estados de Manager.profesores y Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una cubeta vacía dentro Manager.cursadas de clave “Mi cursada”. * Invocar con (PRO). | Manager.profesores tiene un profesor menos.  Ya no existe PRO en dicha colección. |  |
| **BP6** | Verificar camino 6. | Estado del parámetro profesor.  Estados de Manager.profesores y Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una asignatura ASIde nombre “Programación”. * Agregar a Manager.cursadas una cursada con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. Incorporar en esta cursada a PRO. * Invocar con (PRO). | Manager.profesores tiene un profesor menos.  Ya no existe PRO en dicha colección.  PRO ya no figura en ninguna cursada. |  |

**bajaAsignatura(Asignatura asignatura)**

****

CC = número de arcos (36) – número de nodos (26) + 2 = 12

CC = nodos predicados (11) + 1 = 12

CC = número de regiones (11) + 1 = 12

**camino 1: 1, 2, 26, F**

**camino 2: 1, 2, 3, 26, F**

**camino 3: 1, 2, 3, 4, 6, 7, 16, 17, 21, 22, F**

**camino 4: 1, 2, 3, 4, 5, 6, 7, 16, 17, 21, 22, F**

**camino 5: 1, 2, 3, 4, 6, 7, 8, 9, 7, 16, 17, 21, 22, F**

**camino 6: 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 9, 7, 16, 17, 21, 22, F**

**camino 7: 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12 - 13, 14, 9, 7, 16, 17, 21, 22, F** (no viable)

**camino 8: 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12 - 13, 14, 15, 9, 7, 16, 17, 21, 22, F**

**camino 9: 1, 2, 3, 4, 6, 7, 16, 17, 18, 19, 17, 21, 22, F**

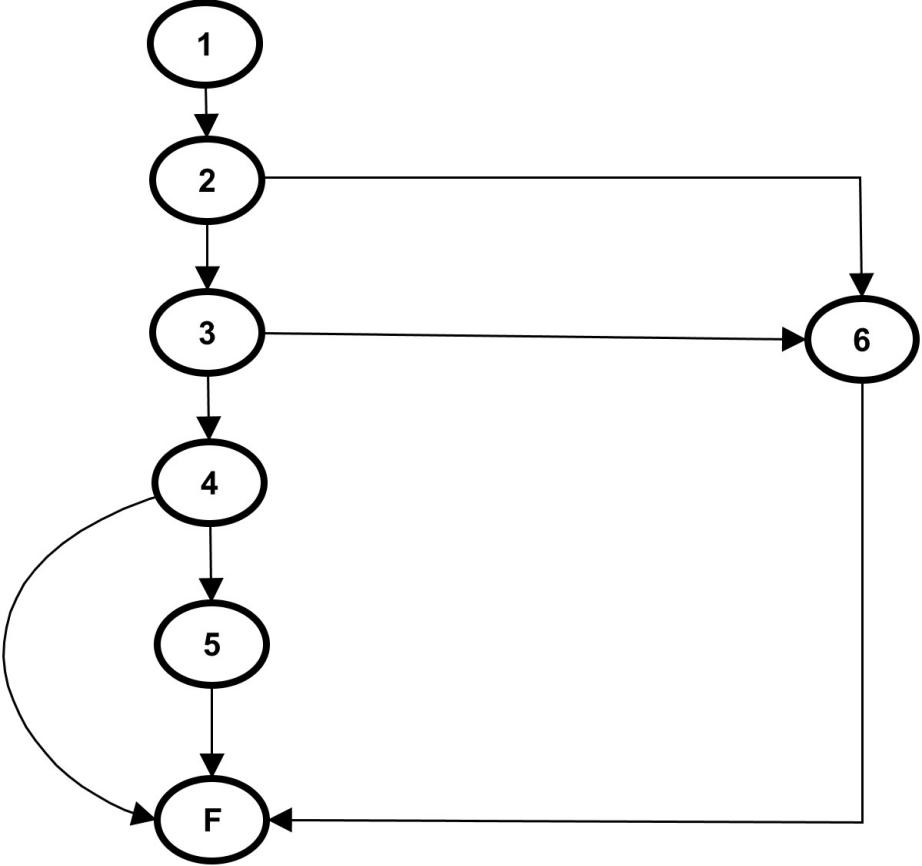
**camino 10: 1, 2, 3, 4, 6, 7, 16, 17, 18, 19, 20, 19, 17, 21, 22, F**

**camino 11: 1, 2, 3, 4, 6, 7, 16, 17, 21, 22, 23, 24, 22, F**

**camino 12: 1, 2, 3, 4, 6, 7, 16, 17, 21, 22, 23, 24, 25, 24, 22, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **BAs1** | Verificar camino 1. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.alumnos, Manager.profesores y Manager.cursada vacíos pero no nulos. * Crear una asignatura ASI de nombre “Programación”. No incorporarla a Manager.asignaturas. * Invocar con (ASI). | NoEstaEntidadException |  |
| **BAs2** | Verificar camino 2. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Agregar a Manager.asignaturas una asignatura de nombre “Programación”. * Crear otra asignatura ASI de nombre “Programación”. No incorporarla a Manager.asignaturas. * Invocar con (ASI). | NoEstaEntidadException | PASS |
| **BAs3** | Verificar camino 3. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Agregar a Manager.asignaturas una asignatura de nombre “Programación”. * Incorporar a Manager.asignaturas otra asignatura ASI de nombre “Programación”. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección. |  |
| **BAs4** | Verificar camino 4. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.asignaturas, Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.asignaturas una asignatura ASI de nombre “Programación”. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección. |  |
| **BAs5** | Verificar camino 5. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.asignaturas una asignatura ASI de nombre “Programación”. * Crear una cubeta vacía dentro Manager.cursadas de clave “Mi cursada”. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección. |  |
| **BAs6** | Verificar camino 6. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.asignaturas una asignatura ASI de nombre “Programación”. * Crear una asignatura ASIC de nombre “Álgebra A”. * Agregar a Manager.cursadas una cursada con atributos nombre = “Mi cursada”, asignatura = ASIC, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección.  Las cursadas existentes no se ven afectadas por la eliminación de ASI. |  |
| **BAs7** | Verificar camino 7 adaptado al código dado. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.asignaturas una asignatura ASI de nombre “Programación”. * Crear una asignatura ASIC de nombre “Álgebra A”. * Agregar a Manager.cursadas una cursada con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Agregar a Manager.cursadas una cursada con atributos nombre = “Mi cursada”, asignatura = ASIC, periodo = “02-2018”, dia = “Lunes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección.  No quedan cursadas cuya asignatura sea ASI. | PASS |
| **BAs8** | Verificar camino 8. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.asignaturas una asignatura ASI de nombre “Programación”. * Agregar a Manager.cursadas una cursada con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección.  No quedan cursadas cuya asignatura sea ASI. | PASS |
| **BAs9** | Verificar camino 9. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.asignaturas una asignatura ASI de nombre “Programación”. * Agregar a Manager.asignaturas una cubeta de clave “Programación 2”. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección. |  |
| **BAs10** | Verificar camino 10. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.asignaturas una asignatura ASI de nombre “Programación”. * Agregar a Manager.asignaturas una asignatura de nombre “Programación 2”. Poner a ASI como correlativa de la nueva asignatura. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección.  ASI ya no es correlativa de ninguna asignatura. |  |
| **BAs11** | Verificar camino 11. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.asignaturas una asignatura ASI de nombre “Programación”. * Agregar a Manager.profesores una cubeta cuya clave es producto de la combinación del nombre y “Manuel” y del apellido “Nucci”. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección. |  |
| **BAs12** | Verificar camino 12. | Estado del parámetro asignatura.  Estados de Manager.asignaturas, Manager.profesores y Manager.cursadas. | * Dejar Manager.profesores y Manager.cursada vacíos pero no nulos. * Incorporar a Manager.asignaturas una asignatura ASI de nombre “Programación”. * Agregar un profesor a Manager.profesores con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. Agregarle a este nuevo profesor ASI a sus competencias. * Invocar con (ASI). | Manager.asignaturas tiene una asignatura menos.  Ya no existe ASI en dicha colección.  No hay ningún profesor en el sistema que tenga a ASI como competencia. |  |

**bajaCursada(Cursada cursada)**

****

CC = número de arcos (9) – número de nodos (7) + 2 = 4

CC = nodos predicados (3) + 1 = 4

CC = número de regiones (4) + 1 = 4

**camino 1: 1, 2, 6, F**

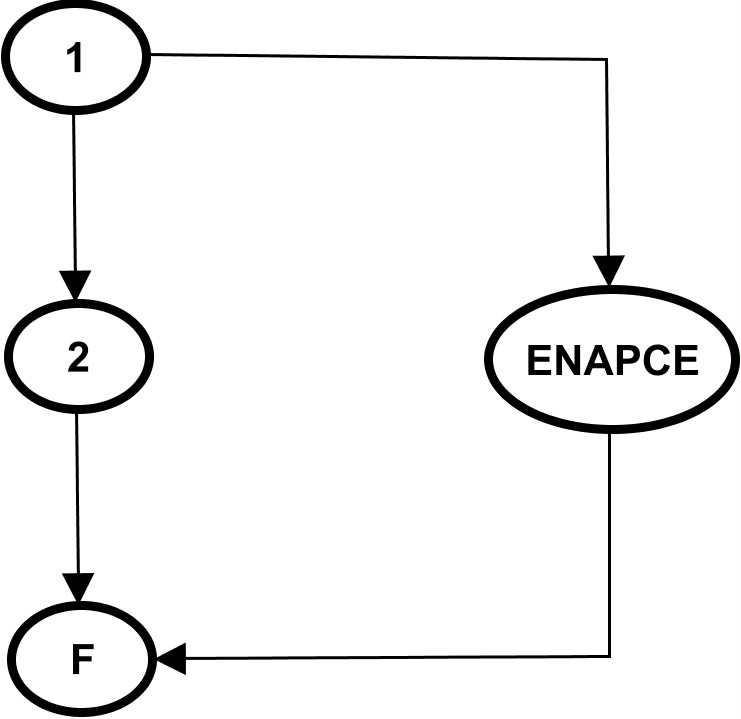
**camino 2: 1, 2, 3, 6, F**

**camino 3: 1, 2, 3, 4, F**

**camino 4: 1, 2, 3, 4, 5, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **BC1** | Verificar camino 1. | Estado del parámetro cursada.  Estado de Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Crear una asignatura ASIde nombre “Programación”. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. No incorporarla a Manager.cursada. * Invocar con (CUR). | NoEstaEntidadException | PASS |
| **BC2** | Verificar camino 2. | Estado del parámetro cursada.  Estado de Manager.cursadas. | * Dejar Manager.cursada vacío pero no nulo. * Crear una asignatura ASIde nombre “Programación”. * Agregar a Manager.cursadas una cursada con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “02-2018”, dia = “Lunes”, horaInicio = “12:00” y horaFin = “14:00”. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. No incorporarla a Manager.cursada. * Invocar con (CUR). | NoEstaEntidadException | PASS |
| **BC3** | Verificar camino 3. | Estado del parámetro cursada.  Estado de Manager.cursadas. | * Agregar a Manager.cursadas una cursada con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “02-2018”, dia = “Lunes”, horaInicio = “12:00” y horaFin = “14:00”. * Incorporar a Manager.cursadas una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (CUR). | Manager.cursadastiene una cursada menos.  Ya no existe CUR en dicha colección. | PASS |
| **BC4** | Verificar camino 4. | Estado del parámetro cursada.  Estado de Manager.cursadas. | * Dejar Manager.cursadas vacío pero no nulo. * Incorporar a Manager.cursadas una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (CUR). | Manager.cursadastiene una cursada menos.  Ya no existe CUR en dicha colección. |  |

**altaAlumnoACursada(Alumno alumno, Cursada cursada)**

****

CC = número de arcos (4) – número de nodos (4) + 2 = 2

CC = nodos predicados (1) + 1 = 2

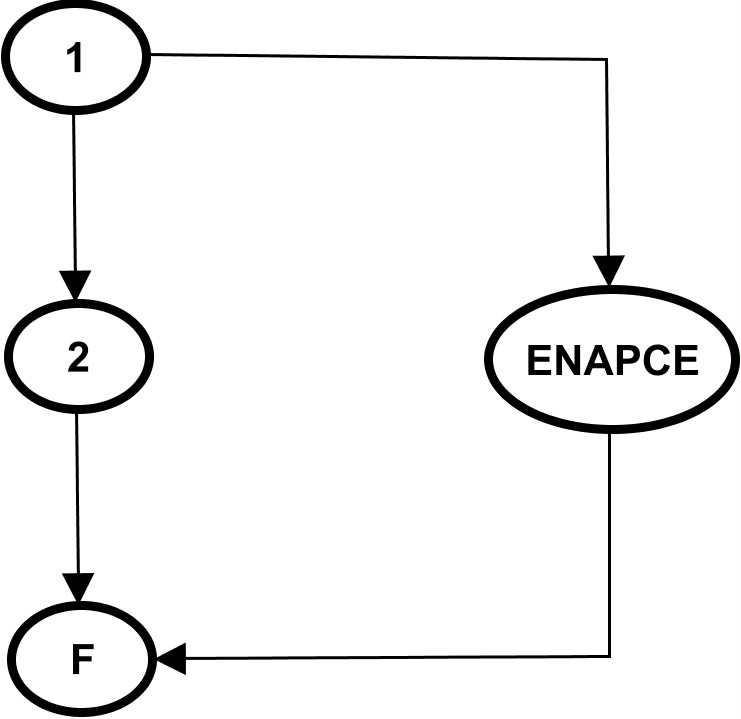
CC = número de regiones (1) + 1 = 2

**camino 1: 1, ENAPCE, F**

**camino 2: 1, 2, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **AAC1** | Verificar camino 1. | Estado de los parámetros alumno y cursada. | * Crear una asignatura ASIde nombre “Programación”. * Crear una asignatura ASI2de nombre “Programación 2”. Agregarle ASI como correlativa. * Crear un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI2, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (ALU, CUR). | EntidadNoAptaParaCursadaException |  |
| **AAC2** | Verificar camino 2. | Estado de los parámetros alumno y cursada. | * Crear una asignatura ASIde nombre “Programación”. * Crear un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (ALU, CUR). | La cursada CUR tiene un alumno más.  ALU está entre los alumnos de CUR. |  |

**altaProfesorACursada(Profesor profesor, Cursada cursada)**

****

CC = número de arcos (4) – número de nodos (4) + 2 = 2

CC = nodos predicados (1) + 1 = 2

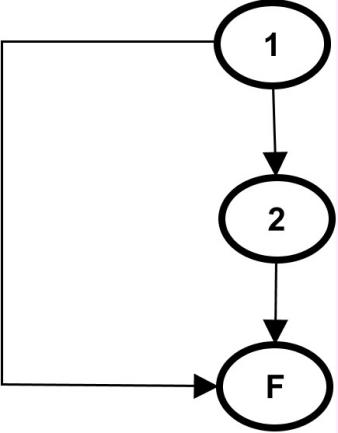
CC = número de regiones (1) + 1 = 2

**camino 1: 1, ENAPCE, F**

**camino 2: 1, 2, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **APC1** | Verificar camino 1. | Estado de los parámetros profesor y cursada. | * Crear una asignatura ASIde nombre “Programación”. * Crear un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (PRO, CUR). | EntidadNoAptaParaCursadaException |  |
| **APC2** | Verificar camino 2. | Estado de los parámetros profesor y cursada. | * Crear una asignatura ASIde nombre “Programación”. * Crear un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. Agregarle ASI como una competencia. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (ALU, CUR). | La cursada CUR tiene un profesor más.  PRO está entre los profesores de CUR. |  |

**bajaAlumnoDeCursada(Alumno alumno, Cursada cursada)**

****

CC = número de arcos (3) – número de nodos (3) + 2 = 2

CC = nodos predicados (1) + 1 = 2

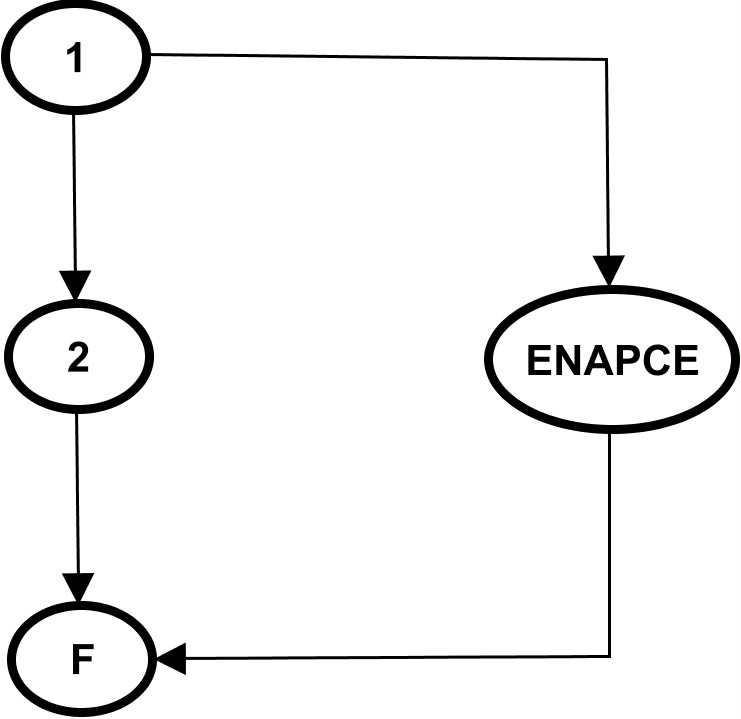
CC = número de regiones (1) + 1 = 2

**camino 1: 1, F**

**camino 2: 1, 2, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **BAC1** | Verificar camino 1. | Estado de los parámetros alumno y cursada. | * Crear una asignatura ASIde nombre “Programación”. * Crear un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. Incorporar a ALU a la nueva cursada. * Invocar con (ALU, CUR). | La cursada CUR tiene un alumno menos.  ALU ya no figura entre los alumnos de CUR. |  |
| **BAC2** | Verificar camino 2. | Estado de los parámetros alumno y cursada. | * Crear una asignatura ASIde nombre “Programación”. * Crear un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (ALU, CUR). | NoEstaEntidadException |  |

**bajaProfesorDeCursada(Profesor profesor, Cursada cursada)**

****

CC = número de arcos (3) – número de nodos (3) + 2 = 2

CC = nodos predicados (1) + 1 = 2

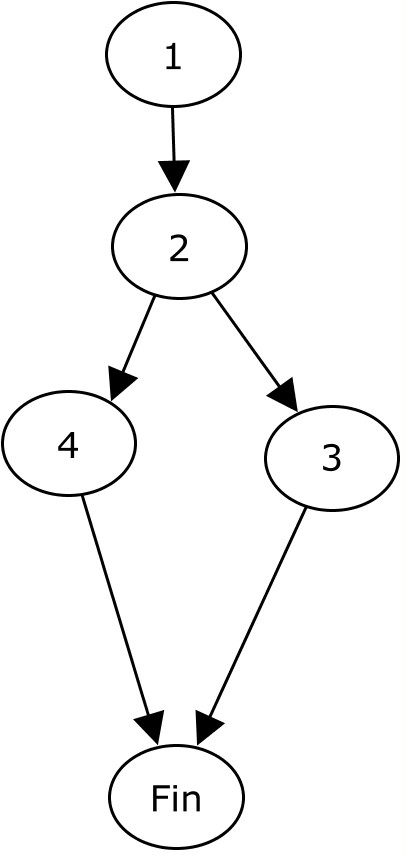
CC = número de regiones (1) + 1 = 2

**camino 1: 1, F**

**camino 2: 1, 2, F**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **BPC1** | Verificar camino 1. | Estado de los parámetros profesor y cursada. | * Crear una asignatura ASIde nombre “Programación”. * Crear un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. Incorporar a ASI como una competencia del nuevo profesor. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. Incorporar a ALU a la nueva cursada. * Invocar con (PRO, CUR). | La cursada CUR tiene un profesor menos.  PRO ya no figura entre los profesores de CUR. |  |
| **BPC2** | Verificar camino 2. | Estado de los parámetros profesor y cursada. | * Crear una asignatura ASIde nombre “Programación”. * Crear un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Crear una cursada CUR con atributos nombre = “Mi cursada”, asignatura = ASI, periodo = “01-2018”, dia = “Martes”, horaInicio = “12:00” y horaFin = “14:00”. * Invocar con (PRO, CUR). | NoEstaEntidadException |  |

**ubicarAlumno(String nombre, String apellido)**

****

CC = número de arcos (5) – número de nodos (5) + 2 = 2

CC = nodos predicados (1) + 1 = 2

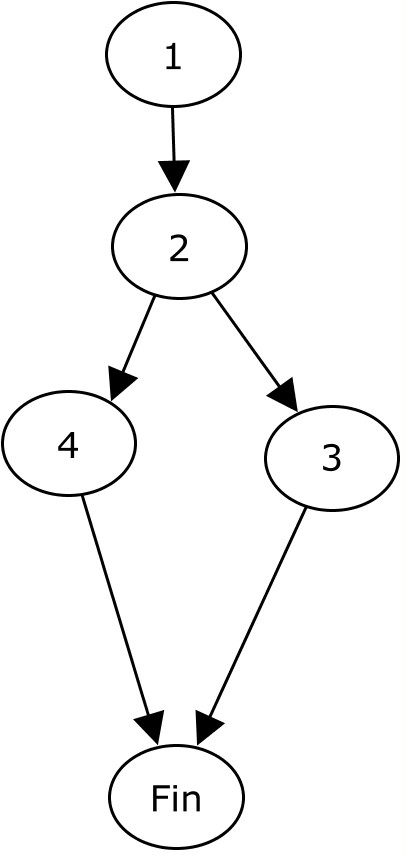
CC = número de regiones (1) + 1 = 2

**camino 1: 1, 2, 4, Fin**

**camino 2: 1, 2, 3, Fin**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **UAl1** | Verificar camino 1. | Nombre del alumno a buscar.  Estado de Manager.alumnos. | * Agregar a Manager.alumnos un alumno con nombre: “Braulio” y apellido “Pablos”. * Invocar con (“Braulio”,”Pablos”). | Todos los alumnos que coinciden con el nombre y apellido ingresados. |  |
| **UAl2** | Verificar camino 2. | Nombre del alumno a buscar.  Estado de Manager.alumnos. | * Mantener Manager.alumnos vacío pero no nulo. * Invocar con (“Braulio”,”Pablos”). | NoEstaEntidadException |  |

**ubicarProfesor(String nombre, String apellido)**

****

CC = número de arcos (5) – número de nodos (5) + 2 = 2

CC = nodos predicados (1) + 1 = 2

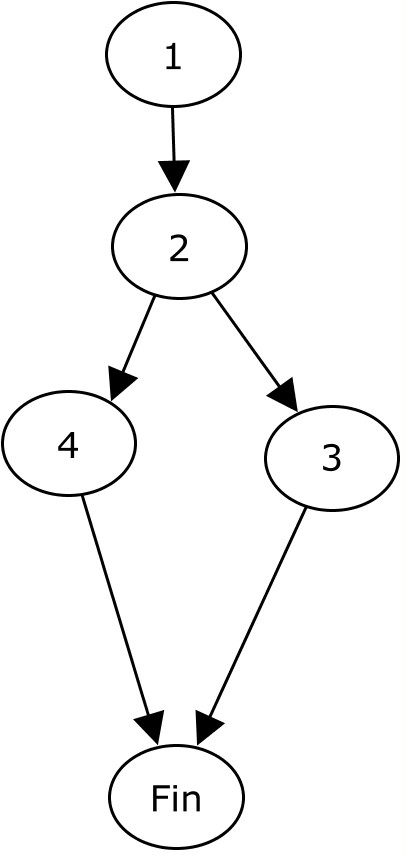
CC = número de regiones (1) + 1 = 2

**camino 1: 1, 2, 4, Fin**

**camino 2: 1, 2, 3, Fin**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **UP1** | Verificar camino 1. | Nombre del profesor a buscar.  Estado de Manager.profesores. | * Agregar a Manager.profesores un profesor con nombre: “Guillermo” y apellido “Lazurri”. * Invocar con (“Guillermo”,”Lazurri”). | Todos los profesores que coinciden con el nombre y apellido ingresados. |  |
| **UP2** | Verificar camino 2. | Nombre del profesor a buscar.  Estado de Manager.profesores. | * Mantener Manager.profesores vacío pero no nulo. * Invocar con (“Guillermo”,”Lazurri”). | NoEstaEntidadException |  |

**ubicarAsignatura(String nombre)**

****

CC = número de arcos (5) – número de nodos (5) + 2 = 2

CC = nodos predicados (1) + 1 = 2

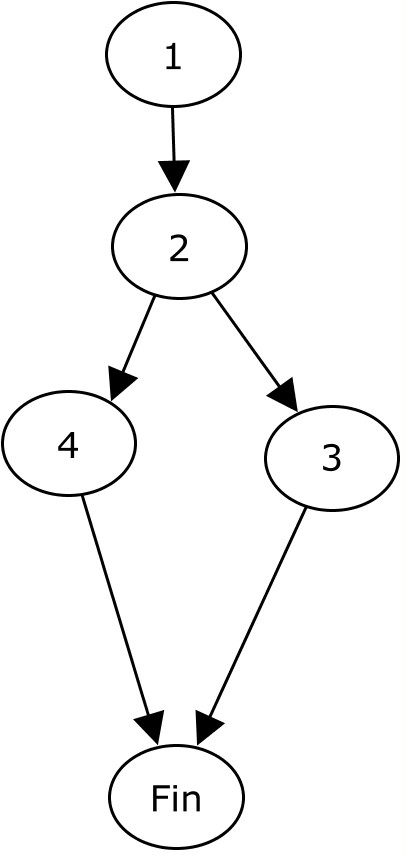
CC = número de regiones (1) + 1 = 2

**camino 1: 1, 2, 4, Fin**

**camino 2: 1, 2, 3, Fin**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **UAs1** | Verificar camino 1. | Nombre de la asignatura a buscar.  Estado de Manager.asignaturas. | * Agregar a Manager.asignaturas una asignatura con nombre: “Programación 3”. * Invocar con (“Programación 3”). | Todas las asignaturas que coinciden con el nombre y apellido ingresados. |  |
| **UAs2** | Verificar camino 2. | Nombre de la asignatura a buscar.  Estado de Manager.asignaturas. | * Mantener Manager.asignaturas vacío pero no nulo. * Invocar con (“Programación 3”). | NoEstaEntidadException |  |

**ubicarCursada(String nombre)**

****

CC = número de arcos (5) – número de nodos (5) + 2 = 2

CC = nodos predicados (1) + 1 = 2

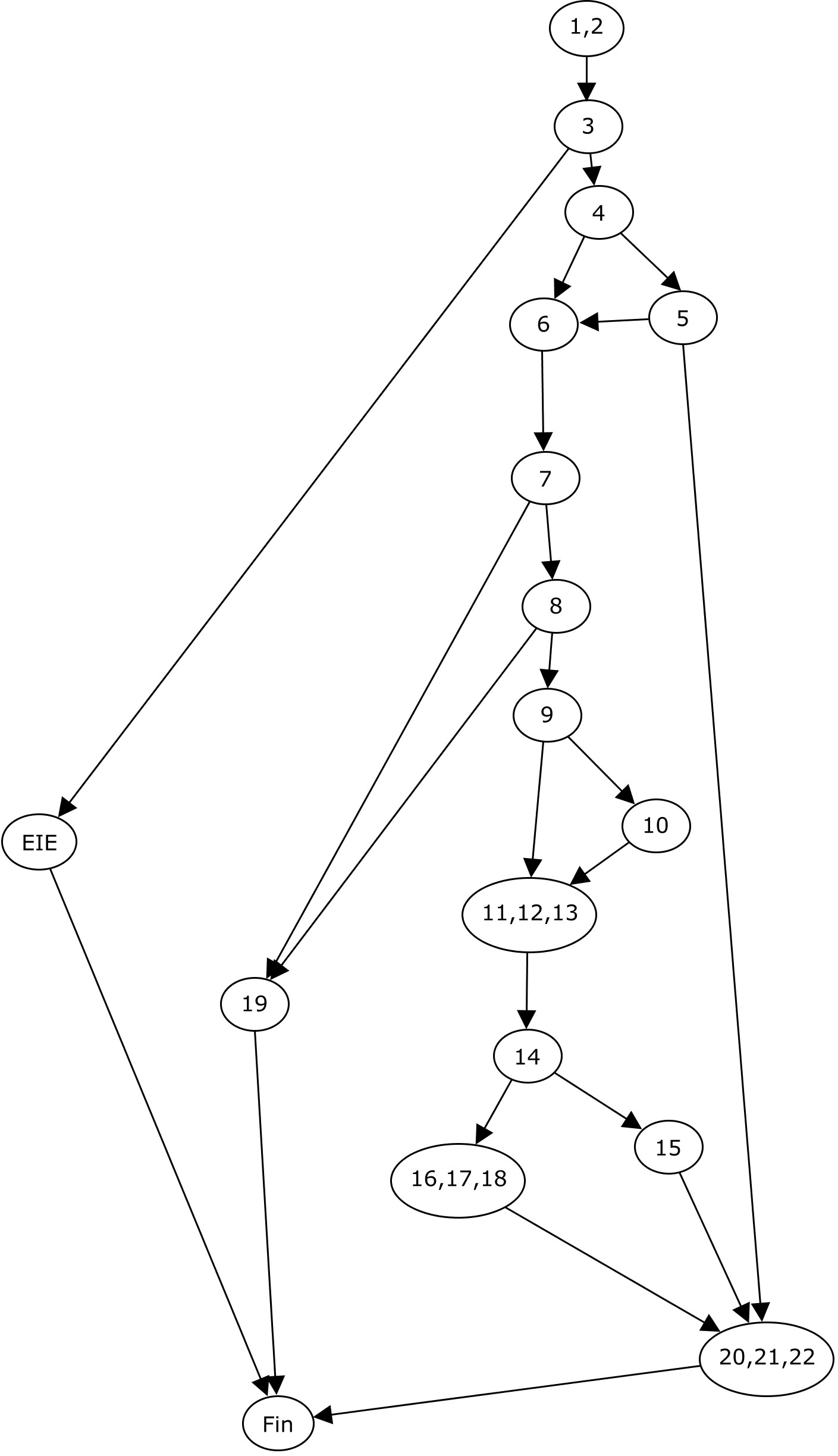
CC = número de regiones (1) + 1 = 2

**camino 1: 1, 2, 4, Fin**

**camino 2: 1, 2, 3, Fin**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **UC1** | Verificar camino 1. | Nombre de la cursada a buscar.  Estado de Manager.cursadas. | * Agregar a Manager.cursadas una cursada con nombre: “Mi Cursada”. * Invocar con (“Mi Cursada”). | Todas las cursadas que coinciden con el nombre ingresado. |  |
| **UC2** | Verificar camino 2. | Nombre de la cursada a buscar.  Estado de Manager.cursadas. | * Mantener Manager.cursadas vacío pero no nulo. * Invocar con (“Mi Cursada”). | NoEstaEntidadException |  |

**modificaAlumno(Alumno alumno, String nombre, String apellido, String domicilio, String telefono, String mail)**

****

CC = número de arcos (23) – número de nodos (17) + 2 = 8

CC = nodos predicados (7) + 1 = 8

CC = número de regiones (7) + 1 = 8

**camino 1: 1, 2, 3, EIE, Fin**

**camino 2: 1, 2, 3, 4, 5, 20, 21, 22, Fin**

**camino 3: 1, 2, 3, 4, 5, 6, 7, 19, Fin**

**camino 4: 1, 2, 3, 4, 6, 7, 19, Fin**

**camino 5: 1, 2, 3, 4, 6, 7, 8, 19, Fin**

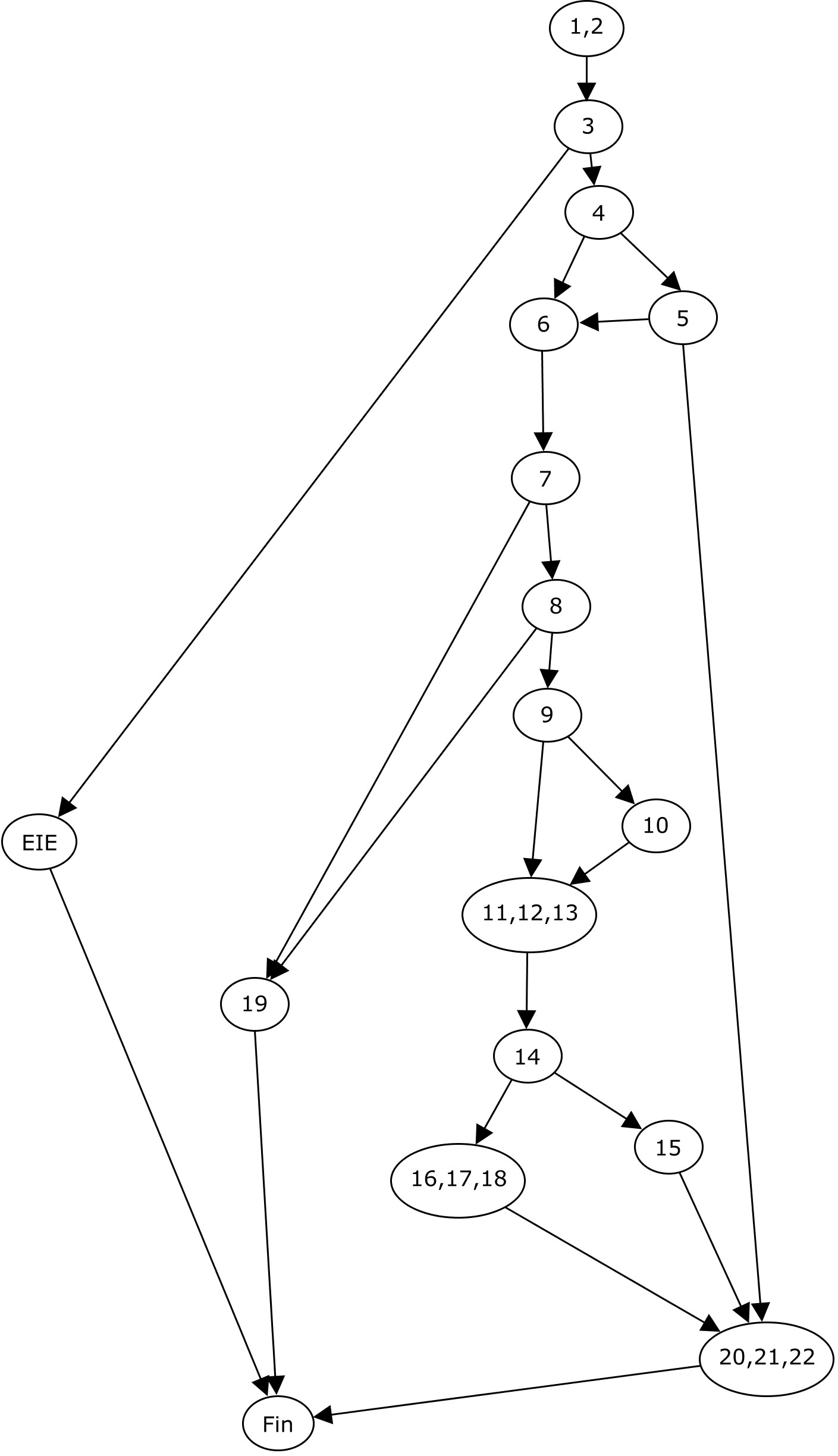
**camino 6: 1, 2, 3, 4, 6, 7, 8, 9, 11, 12, 13, 14, 15, 20, 21, 22, Fin**

**camino 7: 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 20, 21, 22, Fin**

**camino 8: 1, 2, 3, 4, 6, 7, 8, 9, 11, 12, 13, 14, 16, 17, 18, 20, 21, 22, Fin**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **MAl1** | Verificar camino 1. | Alumno al cual se le modificaran los atributos.  Nombre del alumno.  Apellido del alumno.  Domicilio del alumno.  Teléfono del alumno.  Mail del alumno.  Estado de Manager.alumnos. | * Agregar un alumno (lo llamaremos MANUEL) a Manager.alumnos con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANUEL, “Manuel”, “Nucci”, “Jujuy 345”, “123456789”, “nuccigmail.com”). | EmailInvalidoException |  |
| **MAl2** | Verificar camino 2. | Alumno al cual se le modificaran los atributos.  Nombre del alumno.  Apellido del alumno.  Domicilio del alumno.  Teléfono del alumno.  Mail del alumno.  Estado de Manager.alumnos. | * Agregar un alumno ALU a Manager.alumnos con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANUEL, “Manuel”, “Nucci”, “Ju 345”, “16789”, “nu@gmail.com”). | Al alumno ALU se le cambian los atributos de la siguiente manera:  Domicilio = “Ju 345”.  Telefono = “16789”.  Mail = “nu@gmail.com”. | PASS |
| **MAl3** | Verificar camino 3. | Alumno al cual se le modificaran los atributos.  Nombre del alumno.  Apellido del alumno.  Domicilio del alumno.  Teléfono del alumno.  Mail del alumno.  Estado de Manager.alumnos. | * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Generar un alumno MANU con atributos nombre = “Manuel”, apellido = “Nuccia”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | NoEstaEntidadException |  |
| **MAl4** | Verificar camino 4. | Alumno al cual se le modificaran los atributos.  Nombre del alumno.  Apellido del alumno.  Domicilio del alumno.  Teléfono del alumno.  Mail del alumno.  Estado de Manager.alumnos. | * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Generar un alumno MANU con atributos nombre = “Manuelita”, apellido = “Nuccia”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | NoEstaEntidadException | PASS |
| **MAl5** | Verificar camino 5. | Alumno al cual se le modificaran los atributos.  Nombre del alumno.  Apellido del alumno.  Domicilio del alumno.  Teléfono del alumno.  Mail del alumno.  Estado de Manager.alumnos. | * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Generar un alumno MANU con atributos nombre = “Manuel”, apellido = “Nuccia”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | NoEstaEntidadException | PASS |
| **MAl6** | Verificar camino 6. | Alumno al cual se le modificaran los atributos.  Nombre del alumno.  Apellido del alumno.  Domicilio del alumno.  Teléfono del alumno.  Mail del alumno.  Estado de Manager.alumnos. | * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | El apellido del alumno ALU pasa de ser “Nucci” a ser “nucci”. | PASS |
| **MAl7** | Verificar camino 7. | Alumno al cual se le modificaran los atributos.  Nombre del alumno.  Apellido del alumno.  Domicilio del alumno.  Teléfono del alumno.  Mail del alumno.  Estado de Manager.alumnos. | * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | El apellido del alumno ALU pasa de ser “Nucci” a ser “nucci”.  El nombre del alumno ALU pasa de ser “Manuel” a ser “manuel”. |  |
| **MAl8** | Verificar camino 8. | Alumno al cual se le modificaran los atributos.  Nombre del alumno.  Apellido del alumno.  Domicilio del alumno.  Teléfono del alumno.  Mail del alumno.  Estado de Manager.alumnos. | * Incorporar a Manager.alumnos un alumno ALU con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | El apellido del alumno ALU pasa de ser “Nucci” a ser “nucci”.  El nombre del alumno ALU pasa de ser “Manuel” a ser “manuel”. |  |

**modificaProfesor(Profesor profesor, String nombre, String apellido, String domicilio, String telefono, String mail)**

****

CC = número de arcos (23) – número de nodos (17) + 2 = 8

CC = nodos predicados (7) + 1 = 8

CC = número de regiones (7) + 1 = 8

**camino 1: 1, 2, 3, EIE, Fin**

**camino 2: 1, 2, 3, 4, 5, 20, 21, 22, Fin**

**camino 3: 1, 2, 3, 4, 5, 6, 7, 19, Fin**

**camino 4: 1, 2, 3, 4, 6, 7, 19, Fin**

**camino 5: 1, 2, 3, 4, 6, 7, 8, 19, Fin**

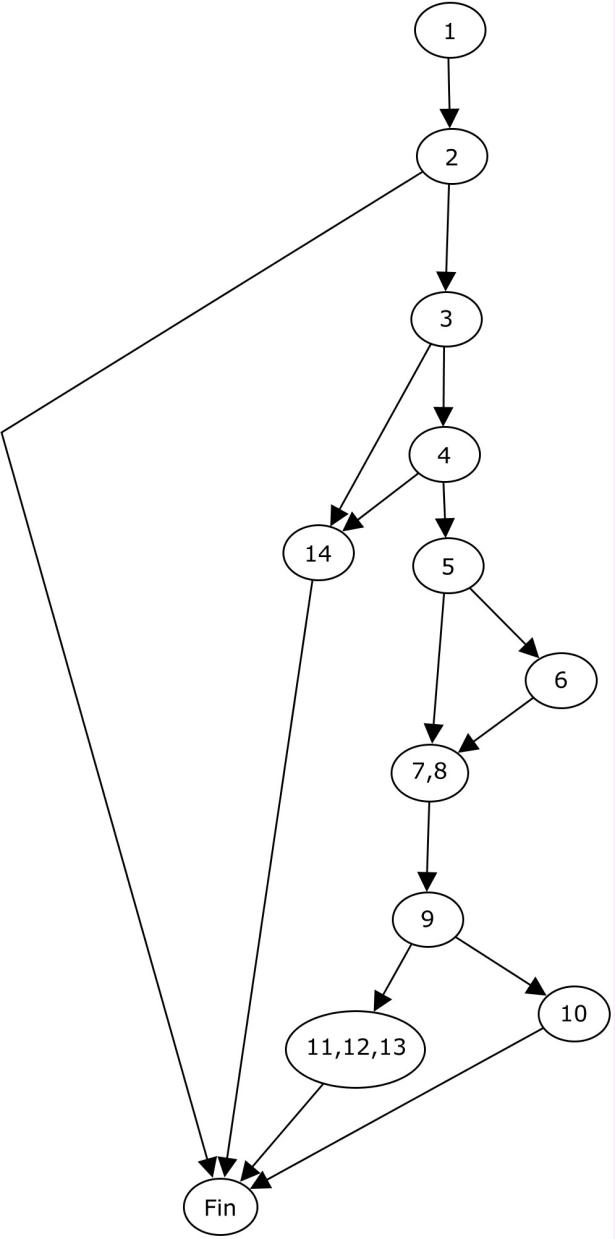
**camino 6: 1, 2, 3, 4, 6, 7, 8, 9, 11, 12, 13, 14, 15, 20, 21, 22, Fin**

**camino 7: 1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 20, 21, 22, Fin**

**camino 8: 1, 2, 3, 4, 6, 7, 8, 9, 11, 12, 13, 14, 16, 17, 18, 20, 21, 22, Fin**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **MP1** | Verificar camino 1. | Profesor al cual se le modificaran los atributos.  Nombre del profesor.  Apellido del profesor.  Domicilio del profesor.  Teléfono del profesor.  Mail del profesor.  Estado de Manager.profesores. | * Agregar un profesor (lo llamaremos MANUEL) a Manager.profesores con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANUEL, “Manuel”, “Nucci”, “Jujuy 345”, “123456789”, “nuccigmail.com”). | EmailInvalidoException |  |
| **MP2** | Verificar camino 2. | Profesor al cual se le modificaran los atributos.  Nombre del profesor.  Apellido del profesor.  Domicilio del profesor.  Teléfono del profesor.  Mail del profesor.  Estado de Manager.profesores. | * Agregar un profesor PRO a Manager.profesores con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANUEL, “Manuel”, “Nucci”, “Ju 345”, “16789”, “nu@gmail.com”). | Al profesor PRO se le cambian los atributos de la siguiente manera:  Domicilio = “Ju 345”.  Telefono = “16789”.  Mail = “nu@gmail.com”. | PASS |
| **MP3** | Verificar camino 3. | Profesor al cual se le modificaran los atributos.  Nombre del profesor.  Apellido del profesor.  Domicilio del profesor.  Teléfono del profesor.  Mail del profesor.  Estado de Manager.profesores. | * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Generar un profesor MANU con atributos nombre = “Manuel”, apellido = “Nuccia”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | NoEstaEntidadException |  |
| **MP4** | Verificar camino 4. | Profesor al cual se le modificaran los atributos.  Nombre del profesor.  Apellido del profesor.  Domicilio del profesor.  Teléfono del profesor.  Mail del profesor.  Estado de Manager.profesores. | * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Generar un profesor MANU con atributos nombre = “Manuelita”, apellido = “Nuccia”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | NoEstaEntidadException | PASS |
| **MP5** | Verificar camino 5. | Profesor al cual se le modificaran los atributos.  Nombre del profesor.  Apellido del profesor.  Domicilio del profesor.  Teléfono del profesor.  Mail del profesor.  Estado de Manager.profesores. | * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Generar un profesor MANU con atributos nombre = “Manuel”, apellido = “Nuccia”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | NoEstaEntidadException | PASS |
| **MP6** | Verificar camino 6. | Profesor al cual se le modificaran los atributos.  Nombre del profesor.  Apellido del profesor.  Domicilio del profesor.  Teléfono del profesor.  Mail del profesor.  Estado de Manager.profesores. | * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | El apellido del profesor PRO pasa de ser “Nucci” a ser “nucci”. | PASS |
| **MP7** | Verificar camino 7. | Profesor al cual se le modificaran los atributos.  Nombre del profesor.  Apellido del profesor.  Domicilio del profesor.  Teléfono del profesor.  Mail del profesor.  Estado de Manager.profesores. | * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | El apellido del profesor PRO pasa de ser “Nucci” a ser “nucci”.  El nombre del profesor PRO pasa de ser “Manuel” a ser “manuel”. |  |
| **MP8** | Verificar camino 8. | Profesor al cual se le modificaran los atributos.  Nombre del profesor.  Apellido del profesor.  Domicilio del profesor.  Teléfono del profesor.  Mail del profesor.  Estado de Manager.profesores. | * Incorporar a Manager.profesores un profesor PRO con atributos nombre = “Manuel”, apellido = “Nucci”, domicilio = “Alberti 2345”, telefono = “987654321” y mail = “manu@gmail.com”. * Invocar con (MANU, “Manuel”, “nucci”, “Jujuy 345”, “123456789”, “nucci@gmail.com”). | El apellido del profesor PRO pasa de ser “Nucci” a ser “nucci”.  El nombre del profesor PRO pasa de ser “Manuel” a ser “manuel”. |  |

**modificaAsignatura(Asignatura asignatura, String nombre)**

****

CC = número de arcos (16) – número de nodos (12) + 2 = 6

CC = nodos predicados (5) + 1 = 6

CC = número de regiones (5) + 1 = 6

**camino 1: 1, 2, Fin**

**camino 2: 1, 2, 3, 14, Fin**

**camino 3: 1, 2, 3, 4, 14, Fin**

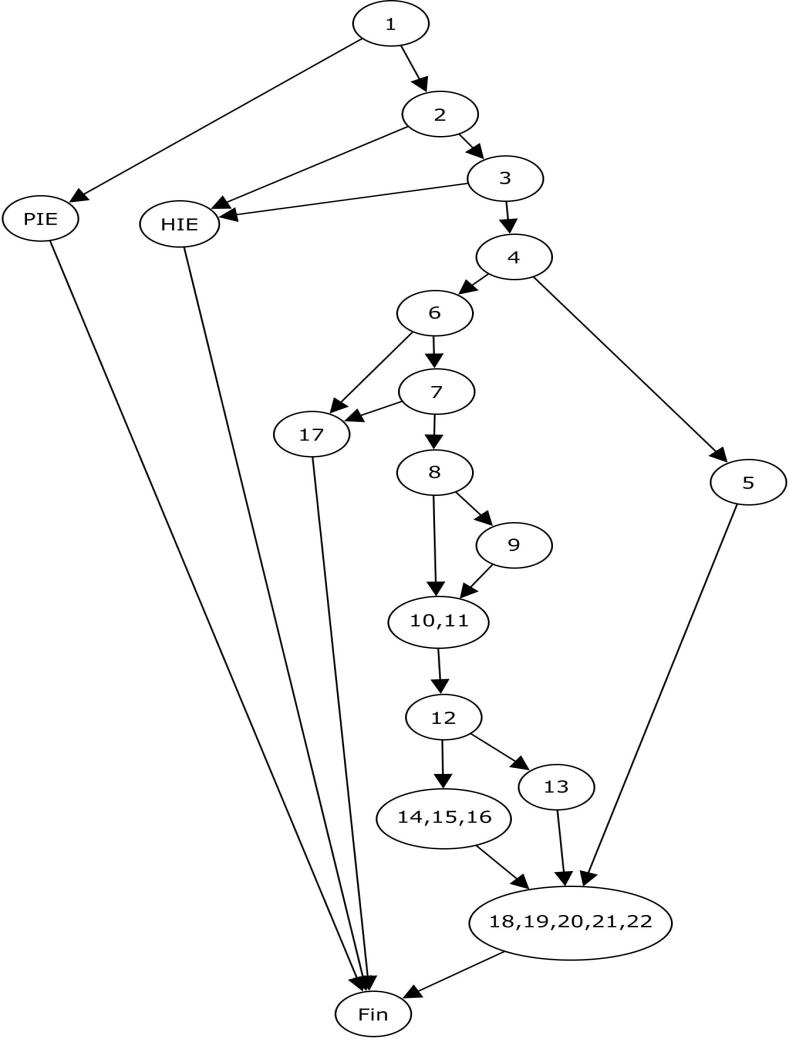
**camino 4: 1, 2, 3, 4, 5, 7, 8, 9, 10, Fin**

**camino 5: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, Fin**

**camino 6: 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, Fin**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **MAs1** | Verificar camino 1. | Asignatura al cual se le modificará el nombre.  Nombre de la asignatura.  Estado de Manager.asignaturas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Invocar con (ASI, “Programación 3”). | No se alteró nada | PASS |
| **MAs2** | Verificar camino 2. | Asignatura al cual se le modificará el nombre.  Nombre de la asignatura.  Estado de Manager.asignaturas. | * Manager.asignaturas debe estar vacío. * Generar una asignatura ASI con nombre = “Programación 3”. * Invocar con (ASI, “Programación 2”). | NoEstaEntidadException | PASS |
| **MAs3** | Verificar camino 3. | Asignatura al cual se le modificará el nombre.  Nombre de la asignatura.  Estado de Manager.asignaturas. | * Agregar una asignatura ASI1 a Manager.asignaturas con nombre = “Programación 3”. * Generar una asignatura ASI2 con nombre = “ProgramacióN 3”. * Invocar con (ASI, “Programación 2”). | NoEstaEntidadException | PASS |
| **MAs4** | Verificar camino 4. | Asignatura al cual se le modificará el nombre.  Nombre de la asignatura.  Estado de Manager.asignaturas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una asignatura ASI2 a Manager.asignaturas con nombre = “ProgramacióN 3”. * Agregar una asignatura ASI3 a Manager.asignaturas con nombre = “ProgramacióN 2”. * Invocar con (ASI, “Programación 2”). | A la asignatura se le cambió el nombre.  Se generó una nueva cubeta para la asignatura y se agregó a la misma.. | PASS |
| **MAs5** | Verificar camino 5. | Asignatura al cual se le modificará el nombre.  Nombre de la asignatura.  Estado de Manager.asignaturas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una asignatura ASI3 a Manager.asignaturas con nombre = “ProgramacióN 2”. * Invocar con (ASI, “Programación 2”). | A la asignatura se le cambió el nombre.  Se agregó a una nueva cubeta la asignatura y se eliminó de la cubeta anterior. |  |
| **MAs6** | Verificar camino 6. | Asignatura al cual se le modificará el nombre.  Nombre de la asignatura.  Estado de Manager.asignaturas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una asignatura ASI2 a Manager.asignaturas con nombre = “ProgramacióN 3”. * Invocar con (ASI, “Programación 2”). | A la asignatura se le cambió el nombre.  Se generó una nueva cubeta para la asignatura y se eliminó de la cubeta anterior. |  |

**modificaCursada(Cursada cursada, String nombre, Asignatura asignatura, String periodo, String dia, String horaInicio, String horaFin)**

****

CC = número de arcos (25) – número de nodos (18) + 2 = 9

CC = nodos predicados (8) + 1 = 9

CC = número de regiones (8) + 1 = 9

**camino 1: 1, PIE, Fin**

**camino 2: 1, 2, HIE, Fin**

**camino 3: 1, 2, 3, HIE, Fin**

**camino 4: 1, 2, 3, 4, 5, 18, 19, 20, 21, 22, Fin**

**camino 5: 1, 2, 3, 4, 5, 6, 17, Fin**

**camino 6: 1, 2, 3, 4, 5, 6, 7, 17, Fin**

**camino 7: 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 14, 15, 16, 18, 19, 20, 21, 22, Fin**

**camino 8: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 18, 19, 20, 21, 22, Fin**

**camino 9: 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20, 21, 22, Fin**

| **Id** | **Objetivo de la prueba** | **Datos de entrada** | **Procedimiento** | **Salida esperada** | **Resultado** |
| --- | --- | --- | --- | --- | --- |
| **MC1** | Verificar camino 1. | Cursada a la cual se le modificaran los atributos.  Nombre de la cursada.  Asignatura de la cursada.  Periodo de la cursada.  Día de la cursada.  Hora de inicio de la cursada.  Hora de finalización de la cursada.  Estado de Manager.asignaturas.  Estado de Manager.cursadas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una cursada CUR a Manager.cursadas con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Invocar con (CUR, “cursada1”, ASI, “7”, “Lunes”, “14:00”, “16:00”). | PeriodoInvalidoException |  |
| **MC2** | Verificar camino 2. | Cursada a la cual se le modificaran los atributos.  Nombre de la cursada.  Asignatura de la cursada.  Periodo de la cursada.  Día de la cursada.  Hora de inicio de la cursada.  Hora de finalización de la cursada.  Estado de Manager.asignaturas.  Estado de Manager.cursadas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una cursada CUR a Manager.cursadas con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Invocar con (CUR, “cursada1”, ASI, “01-2017”, “Lunes”, “1”, “16:00”). | HoraInvalidaException |  |
| **MC3** | Verificar camino 3. | Cursada a la cual se le modificaran los atributos.  Nombre de la cursada.  Asignatura de la cursada.  Periodo de la cursada.  Día de la cursada.  Hora de inicio de la cursada.  Hora de finalización de la cursada.  Estado de Manager.asignaturas.  Estado de Manager.cursadas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una cursada CUR a Manager.cursadas con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Invocar con (CUR, “cursada1”, ASI, “01-2017”, “Lunes”, “14:00”, “1”). | HoraInvalidaException |  |
| **MC4** | Verificar camino 4. | Cursada a la cual se le modificaran los atributos.  Nombre de la cursada.  Asignatura de la cursada.  Periodo de la cursada.  Día de la cursada.  Hora de inicio de la cursada.  Hora de finalización de la cursada.  Estado de Manager.asignaturas.  Estado de Manager.cursadas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una cursada CUR a Manager.cursadas con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Invocar con (CUR, “cursada1”, ASI, “01-2018”, “Lunes”, “14:00”, “16:00”). | Se cambió el atributo de periodo en cursada por el nuevo. | PASS |
| **MC5** | Verificar camino 5. | Cursada a la cual se le modificaran los atributos.  Nombre de la cursada.  Asignatura de la cursada.  Periodo de la cursada.  Día de la cursada.  Hora de inicio de la cursada.  Hora de finalización de la cursada.  Estado de Manager.asignaturas.  Estado de Manager.cursadas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una cursada CUR1 a Manager.cursadas con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Crear una cursada CUR2 con:   Nombre=”cursada2”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Invocar con (CUR2, “CURSADA1”, ASI, “01-2018”, “Lunes”, “14:00”, “16:00”). | NoEstaEntidadException | PASS |
| **MC6** | Verificar camino 6. | Cursada a la cual se le modificaran los atributos.  Nombre de la cursada.  Asignatura de la cursada.  Periodo de la cursada.  Día de la cursada.  Hora de inicio de la cursada.  Hora de finalización de la cursada.  Estado de Manager.asignaturas.  Estado de Manager.cursadas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una cursada CUR1 a Manager.cursadas con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Crear una cursada CUR2 con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Invocar con (CUR2, “CURSADA1”, ASI, “01-2018”, “Lunes”, “14:00”, “16:00”). | NoEstaEntidadException | PASS |
| **MC7** | Verificar camino 7. | Cursada a la cual se le modificaran los atributos.  Nombre de la cursada.  Asignatura de la cursada.  Periodo de la cursada.  Día de la cursada.  Hora de inicio de la cursada.  Hora de finalización de la cursada.  Estado de Manager.asignaturas.  Estado de Manager.cursadas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una cursada CUR1 a Manager.cursadas con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Crear una cursada CUR2 con:   Nombre=”CURSADA1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Invocar con (CUR1, “cursada2”, ASI, “01-2018”, “Lunes”, “14:00”, “16:00”). | Se eliminó la cursada de las cubetas y se creó una nueva con la cursada dentro.  Se modificó el nombre de la cursada. |  |
| **MC8** | Verificar camino 8. | Cursada a la cual se le modificaran los atributos.  Nombre de la cursada.  Asignatura de la cursada.  Periodo de la cursada.  Día de la cursada.  Hora de inicio de la cursada.  Hora de finalización de la cursada.  Estado de Manager.asignaturas.  Estado de Manager.cursadas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una cursada CUR1 a Manager.cursadas con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Invocar con (CUR1, “cursada2”, ASI, “01-2018”, “Lunes”, “14:00”, “16:00”). | Se eliminó la cursada de la cubeta, se eliminó la cubeta y se creó una nueva con la cursada dentro.  Se modificó el nombre de la cursada. |  |
| **MC9** | Verificar camino 9. | Cursada a la cual se le modificaran los atributos.  Nombre de la cursada.  Asignatura de la cursada.  Periodo de la cursada.  Día de la cursada.  Hora de inicio de la cursada.  Hora de finalización de la cursada.  Estado de Manager.asignaturas.  Estado de Manager.cursadas. | * Agregar una asignatura ASI a Manager.asignaturas con nombre = “Programación 3”. * Agregar una cursada CUR1 a Manager.cursadas con:   Nombre=”cursada1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Crear una cursada CUR2 con:   Nombre=”CURSADA1”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Crear una cursada CUR3 con:   Nombre=”CURSADA2”  Asignatura= ASI  Periodo= “01-2017”  Dia= “Lunes”  HoraInicio= “14:00”  HoraFin= “16:00”.   * Invocar con (CUR1, “cursada2”, ASI, “01-2018”, “Lunes”, “14:00”, “16:00”). | Se eliminó la cursada de las cubetas y se agregó a la correspondiente.  Se modificó el nombre de la cursada. | PASS |