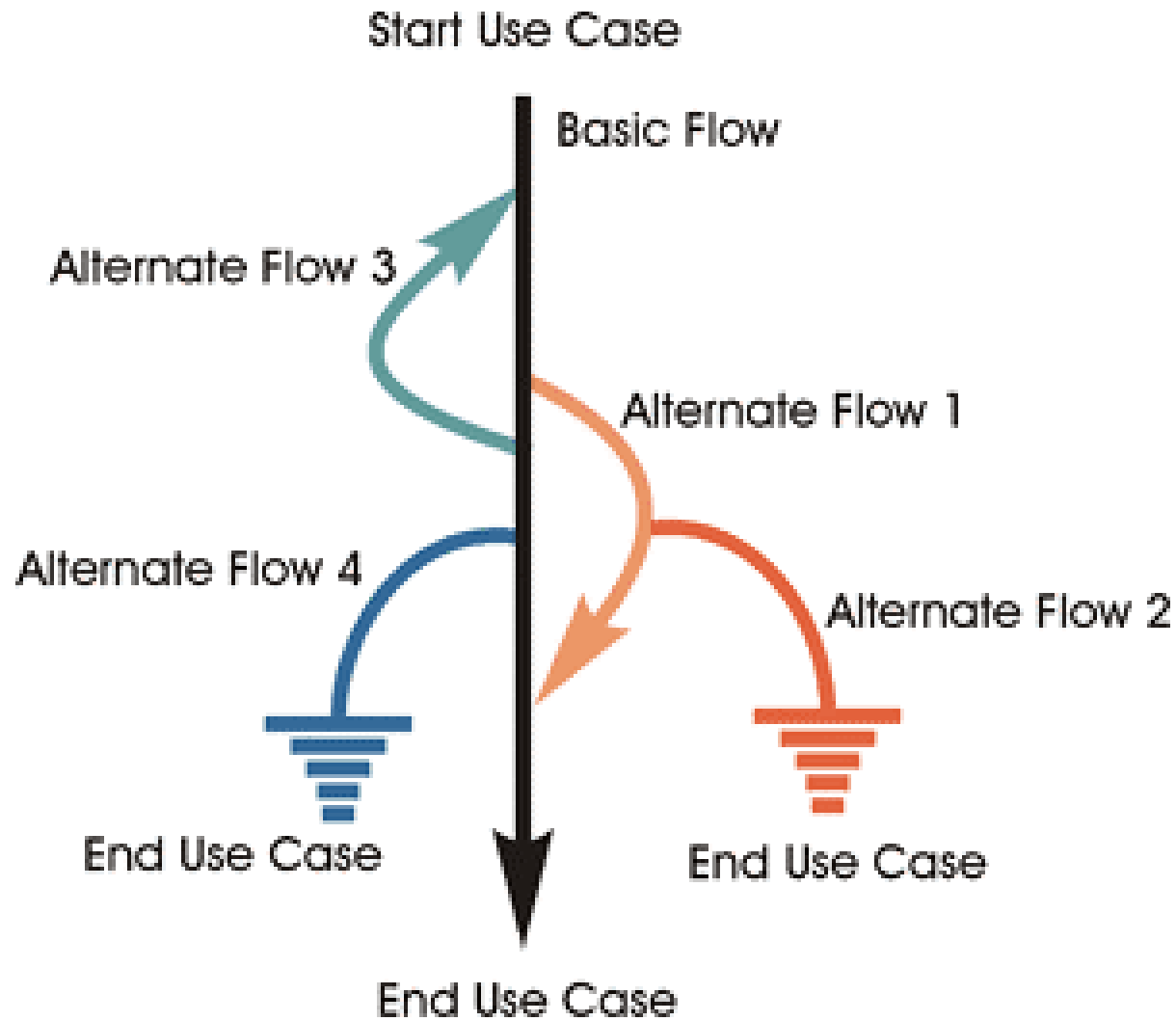


CP a partir de CU: Procedimiento

- Pasos para la generación:
 - 1) Para cada caso de uso, generar el conjunto de todos los escenarios posibles.
 - 2) Para cada escenario, identificar por lo menos un caso de prueba y las condiciones que lo harán ejecutable.
 - 3) Para cada caso de prueba, identificar los datos de prueba a utilizar.

Flujo de Eventos principal y alternativos



Ejercicio: caso de uso “Retiro”

Nombre:	Retiro
Precondiciones: El cliente ingresó una tarjeta válida e ingresó nro. de PIN	
Flujo Principal: <ol style="list-style-type: none">1.El CA despliega las distintas alternativas disponibles,el Cliente elige Retiro2.El CA pide cuenta e importe3.El Cliente elige (o ingresa) la cuenta de la cual retirar y el importe4.CA envía Id. Tarjeta, PIN, cuenta e importe al SC (Servicio de Cajeros)5.El SC contesta: Continuar (OK) o No Continuar6.CA dispensa el dinero7.CA devuelve la tarjeta8.CA imprime el recibo	
Flujos Alternativos: <p>4.a. El importe indicado por el cliente no puede obtenerse a partir de los billetes de que dispone el CA</p> <ol style="list-style-type: none">4.a.1. El CA despliega un mensaje de advertencia y le solicita que ingrese nuevamente el importe.4.a.2. El Sigue en el punto 3 del FP. <p>5.a. El sistema indica que no se han completado el texto obligatorio.</p> <ol style="list-style-type: none">5.a.1. El sistema informa al cliente que falta completar algunos campos obligatorios.5.a.2. Sigue en el punto 4 del flujo principal.	

Ejercicio (cont.)

- Algunos escenarios de un Casos de Uso
- 1. Flujo Principal
- 2. Flujo Principal- Importe inválido (n)
- 3. Flujo Principal- No hay suficiente saldo en cuenta
- 4. Flujo Principal- No se ha completado texto obligatorio

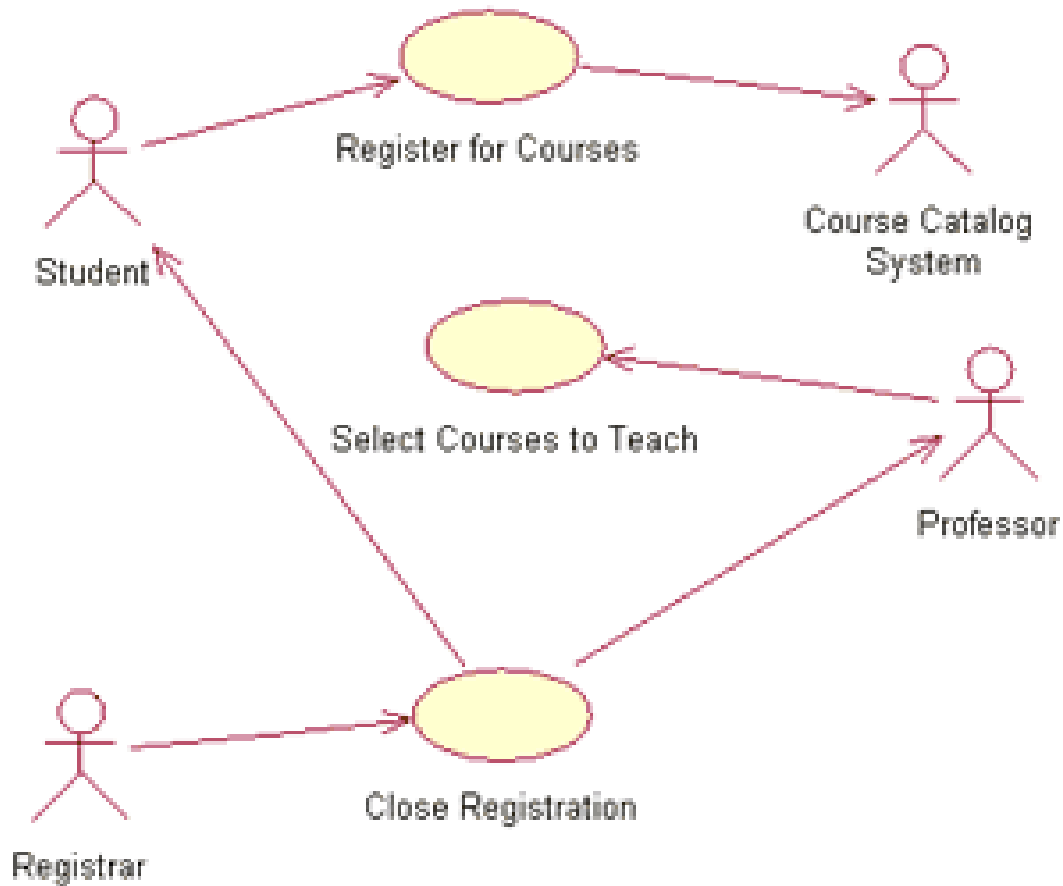
Ejercicio (cont.)

Condiciones	Tipo de Resultado	Caso
1. Normal	Retiro exitoso	
1.1 Normal Caja de Ahorros		
1.2 Normal Cuenta Corriente		
2. Normal-Importe inválido (n)	Retiro exitoso	
3. Normal- sin saldo	No exitoso	

Ejercicio (cont.)

\Caso	A1	A2	A3		A4
Entradas					
Tarjeta	13131	13131	13131		23232
PIN	9001	9001	9001		9002
Cuenta	CA128	CC321	CC321		CA228
Monto	100	500	12	1200	100
Salidas			Importe inválido		Sin saldo
Dispensa	\$100	\$500		\$1200	
Dev.tarjeta	Sí	Sí		Sí	Sí
Recibo	13131...	13131...		13131...	
Condics.	1.1	1.2		2	3

Ejercicio: Inscripción a Curso en la Universidad



- El estudiante se inscribe a cursos.
- El profesor selecciona cursos para los cuales dar clase.
- Bedelía cierra las inscripciones a un curso.

Nombre:	Inscripción a cursos
Precondiciones: El estudiante se encuentra logueado en el sistema.	
<p>Flujo Principal:</p> <ol style="list-style-type: none"> 1.El Estudiante selecciona la opción “Crear una agenda”. 2.El sistema devuelve la lista de cursos disponibles que ofrece el Sistema de Catálogo de Cursos y despliega una lista al estudiante. 3.El estudiante selecciona cuatro cursos primarios ofrecidos y 2 cursos alternativos de la lista de cursos disponibles. 4.El estudiante indica que la agenda está completa. 5.El sistema verifica que el estudiante tiene los pre-requisitos necesarios para la inscripción a cada curso seleccionado en la agenda. 6.El sistema despliega la agenda conteniendo los cursos ofrecidos y un mensaje de confirmación de que se creó la agenda de forma exitosa junto con un número de confirmación . 	
<p>Flujos Alternativos:</p> <p>G1. Salir sin confirmar agenda</p> <ol style="list-style-type: none"> G1.1. El estudiante elige guardar la agenda de forma parcial antes de salir. G1.2. El sistema guarda la agenda. G1.3. Fin del CU. <p>2.a. Sistema de Catálogo de Cursos no disponible</p> <ol style="list-style-type: none"> 2.a.1. El sistema despliega un mensaje de error. 2.a.2. Fin del CU. <p>2.b. Inscripciones cerradas</p> <ol style="list-style-type: none"> 2.a.1. El sistema despliega un mensaje indicando que las inscripciones están cerradas. 2.a.2. Fin del CU. <p>6.a. El estudiante no completa los requisitos, Curso lleno o Conflicto en la Agenda</p> <ol style="list-style-type: none"> 6.a.1.El sistema determina que el estudiante no cumple con los pre-requisitos, el curso está lleno o hay conflictos de agenda. 6.a.2. El sistema despliega un mensaje al estudiante indicando que debe seleccionar un curso distinto. 6.a.3. Sigue en el punto 3 del FP. 	

Consideraciones importantes

- Para un mismo escenario puedo tener varios escenarios condición.
- Es necesario incorporar las condiciones borde de cada escenario.
- Si tengo varios escenarios condición con la misma combinación de I's y V's significa que olvidé poner alguna condición.



Técnicas de caja blanca

Técnicas de caja blanca

- La derivación de los casos de prueba se basa en la estructura inherente o diseño del objeto de prueba.
- Son también llamadas técnicas basadas en la estructura o en el código.
- Tipos de técnicas de caja blanca:
 - Basadas en el **flujo de control** del programa
 - Expresan los cubrimientos del testing en términos del grafo de flujo de control del programa
 - Basadas en el **flujo de datos** del programa
 - Expresan los cubrimientos del testing en términos de las asociaciones definición-uso del programa
- Vamos a ver brevemente las de flujo de control.

Criterios de cubrimiento

- El foco de las técnicas de caja blanca es ejecutar (cubrir) cada parte del código al menos una vez. El cubrimiento de ese código puede tener diferentes criterios:
 - Cubrimiento de sentencias
 - Cubrimiento de decisión
 - Cubrimiento de condición
 - Cubrimiento de decisión/condición
 - Cubrimiento de condición múltiple
 - Cubrimiento de arcos
 - Cubrimiento de caminos
 - Cubrimiento de trayectorias independientes

Cubrimiento de sentencias

- Asegura que el conjunto de casos de pruebas (CCP) ejecuta al menos una vez cada instrucción del código

```
If (a > 1) and (b = 0) {  
    x = x / a  
}
```

```
If (a = 2) or (x > 1) {  
    x = x + 1  
}
```

CP : Entrada a=2, b=0,
x=3

- ¿Que pasa si tenía un or en lugar del and de la primera decisión?
- ¿Que pasa si tenía $x > 0$ en lugar de $x > 1$ en la segunda decisión?
- Hay una secuencia en la cual x se mantiene sin cambios y esta no es probada

Este criterio es tan débil que normalmente se lo considera inútil. Es necesario pero no suficiente (Myers)

Criterio de cubrimiento de decisión

- Cada decisión dentro del código toma al menos una vez el valor true y otra vez el valor false para el CCP

```
If (a > 1) and (b = 0) {  
    x = x / a  
}
```

```
If (a = 2) or (x > 1) {  
    x = x + 1  
}
```

```
CP1  a=3, b=0, x=3  
CP2  a=2, b=1, x=1
```

- ¿Qué pasa si en la segunda decisión tuviera $x < 1$ en lugar de $x > 1$?

- Hay una secuencia en la cual x se mantiene sin cambios y esta no es probada

- Hay que extender el criterio para sentencias del tipo CASE

Es más fino que el criterio de sentencias

Otros criterios

- **Criterio de cubrimiento de condición**
 - Cada condición dentro de una decisión debe tomar al menos una vez el valor true y otra el valor false para el CCP
- **Criterio de cubrimiento de decisión/condición**
 - Combinación de los dos criterios anteriores
- **Criterio de cubrimiento de condición múltiple**
 - Todas las combinaciones posibles de resultados de condición dentro de una decisión se ejecuten al menos una vez
- **Criterio de cubrimiento de caminos**
 - Se ejecutan al menos una vez todos los caminos posibles (combinaciones de trayectorias)

