



2007-08

# Práctica 8: Particiones equivalentes

Tipos de pruebas y objetivos

Pruebas del sistema

Pruebas de componentes

Diseño de casos de prueba

Automatización de las pruebas



## Ejemplo: cálculo de holguras

- método **calcularHolguras**
- **PARÁMETROS DE ENTRADA:**
  - id**: identificador de la actividades, es una cadena de 1 hasta 8 caracteres alfanuméricos,
  - d**: duración de la actividad , es un entero positivo mayor o igual que cero
  - eSi**, y **ISi**: tiempo early y last del suceso inicio de la actividad id
  - eSf**, y **ISf**: tiempo early y last del suceso fin de la actividad id
  - Los parámetros **eSi**, **ISi**, **eSf**, y **ISf** son valores enteros positivos.
- **PARÁMETROS DE SALIDA:**
  - hl** la holgura libre, es un entero positivo,
  - ht** la holgura total, es un entero positivo,
  - fte** es la fecha de terminación más temprana, es un entero positivo
  - ftl** es la fecha de terminación más tardía, es un entero positivo
- **PRECONDICIONES DE ENTRADA:**
  - eSi**, **ISi**, **eSf** y **ISf** son enteros positivos mayores o iguales a cero
  - $eSi \leq \{ ISi, eSf, ISf \}; ISi \leq \{ eSf, ISf \};$

## Ejemplo: pasos a seguir

---

1. Creamos las clases de equivalencias válidas y no válidas para los datos de entrada y de salida. Las clases válidas serán V1, V2, V3, ... y las no válidas N1, N2, N3,...
2. Combinamos dichas clases de equivalencia para crear datos de prueba de forma que queden probadas todas las clases válidas y todas las clases no válidas
3. Creamos los datos de entrada
4. Indicamos el resultado esperado



## Ejemplo: clases de entrada

---

### id:

- V1: cadena alfanumérico entre 1 y 8 caracteres
- N1: cadena alfanumérica mayor de 8 caracteres
- N2: cadena vacía

### d:

- V2: valor cero (actividad ficticia)
- V3: valor entero positivo mayor que cero
- N3: valor numérico negativo
- N5: valor no numérico

### eSi:

- V4: valor entero  $> 0$  y menor o igual que  $\{lSi, eSf, lSf\}$ , con  $lSi < eSf < lSf$
- V14: valor entero  $> 0$  y menor o igual que  $\{lSi, eSf, lSf\}$ , con  $lSi < lSf < eSf$

### lSi:

- V5: valor entero positivo mayor que cero y menor o igual que  $\{eSf, lSf\}$

### eSf:

- V6: valor entero positivo mayor que cero mayor o igual que  $\{eSi, lSi\}$

### lSf:

- V7: valor entero positivo mayor que cero mayor o igual que  $\{eSi, lSi\}$



## Ejemplo: clases de salida

---

### hl:

- V8: valor 0
- V9: valor entero positivo mayor que cero y menor o igual que la holgura total
- N4: valor mayor que la holgura total
- (en este caso podemos considerar que V8 y V9 serían subclases de la clase correspondiente a "valor menor o igual que la holgura total")

### ht:

- V10: valor 0 (actividad crítica)
- V11: valor entero positivo mayor que cero

### fte:

- V12: valor entero positivo mayor que cero

### ftl:

- V13: valor entero positivo mayor que cero

## Ejemplo: clases a probar

---

### ENTRADAS

- (A) V1-V2-V4-V5-V6-V7
- (B) V1-V3-V4-V5-V6-V7
- (D) V1-V3-V14-V5-V6-V7
- (F) N1-V3-V4-V5-V6-V7
- (G) N2-V3-V4-V5-V6-V7
- (H) V1-N3-V4-V5-V6-V7
- (E) V1-N5-V4-V5-V6-V7

### SALIDAS

- (I) V8-V10-V12-V13  
(actividad crítica)
- (J) V8-V11-V12-V13 (hl =0)
- (K) V9-V11-V12-V13 (hl >0)

(L) N4-V11-V12-V13

## Ej: tabla de casos de prueba

Clases probadas	Dato de entrada	Resultado esperado	Resultado real
(A) + (I)	"id1", 0, 3, 7, 3, 7	0, 0, 7, 7	
(B) + (J)	"id1", 5, 3, 7, 8, 15	0, 7, 8, 15	
(B) + (K)	"id1", 5, 3, 7, 10, 15	2, 7, 8, 15	
(D) + (L)	"id1", 5, 3, 7, 10, 8	2, 0, 8, 8 (*)	
(F) + (K)	"cadenamal", 5, 3, 7, 10, 15	2, 7, 8, 15 (**)	
(G) + (K)	", 5, 3, 7, 10, 15	2, 7, 8, 15 (***)	
(H) + (K)	"id1", -5, 3, 7, 10, 8	12, 17, -2, 10 (****)	
(E) + (K)	"id1", "pepe", 3, 7, 10, 8	?, ?, ?, ? (\$)	

- (\*) Es incorrecto que la holgura libre sea mayor que la holgura total, el programa debería dar como resultado un ERROR
- (\*\*) El programa no detecta como error el que la cadena tenga más de ocho caracteres
- (\*\*\*) El programa no detecta como error el que la cadena sea vacía
- (\*\*\*\*) El programa no detecta que se ha introducido entero negativo como duración
- (\$) El programa no detecta que se ha introducido un valor no numérico como duración