

Módulo 5: Diseño de casos de prueba

5.4 Técnica de caminos básicos



Técnica de Caminos Básicos

- Se utiliza cuando existe un "grafo"
 - También llamado "diagrama de estados finitos"
 - Formado por nodos y arcos
 - Reglas:
 - Arcos siempre con <u>una</u> sola dirección (flechas)
 - Arcos siempre con un nodo en cada extremo
 - Un solo inicio y un solo final
- Algunos ejemplos donde existen grafos
 - Eventos en una pantalla (caja negra)
 - Transacciones en un módulo (caja negra)
 - Código fuente (caja blanca)



Técnica de Caminos Básicos

- Objetivo: diseñar casos de prueba que ejerciten cada posible camino
 - Problema: existe una enorme multitud de posibles caminos
 - Necesitamos: encontrar la cantidad <u>mínima</u> de caminos que ejerciten todo el grafo
- Solución: utilizar los "caminos básicos"
 - Son caminos <u>independientes</u>
 - Son pocos
 - Pasan al menos una vez por todos los nodos del grafo



Proceso para el diseño de casos de prueba

Dibujar grafo

Calcular cantidad de caminos básicos

Encontrar caminos básicos

Diseñar un caso de prueba para cada camino básico



Paso 1: Dibujar el grafo

Curso: TC3045 Calidad y Pruebas del Software

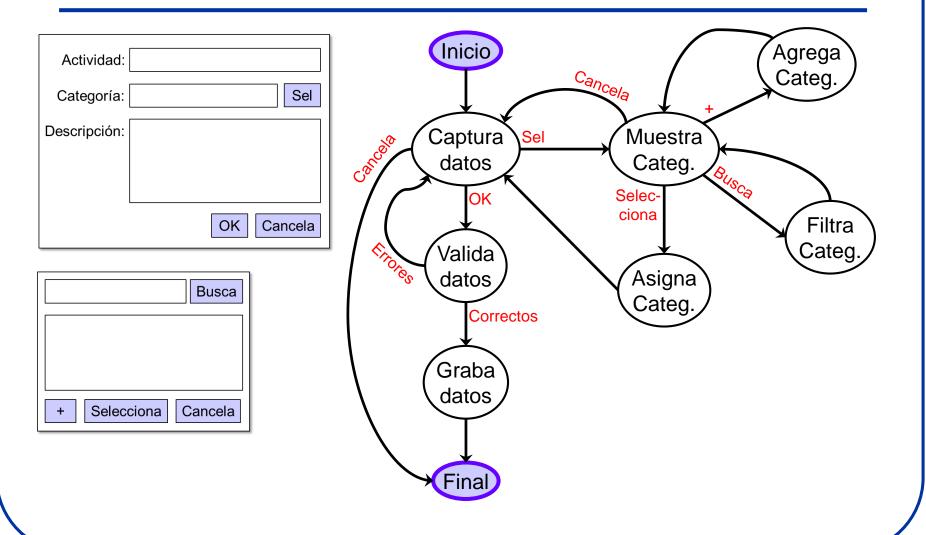


Ejemplo 1: eventos en una pantalla

Actividad:		
Categoría:	Sel	Busca
Descripción:		
	OK Cancela	+ Selecciona Cancela

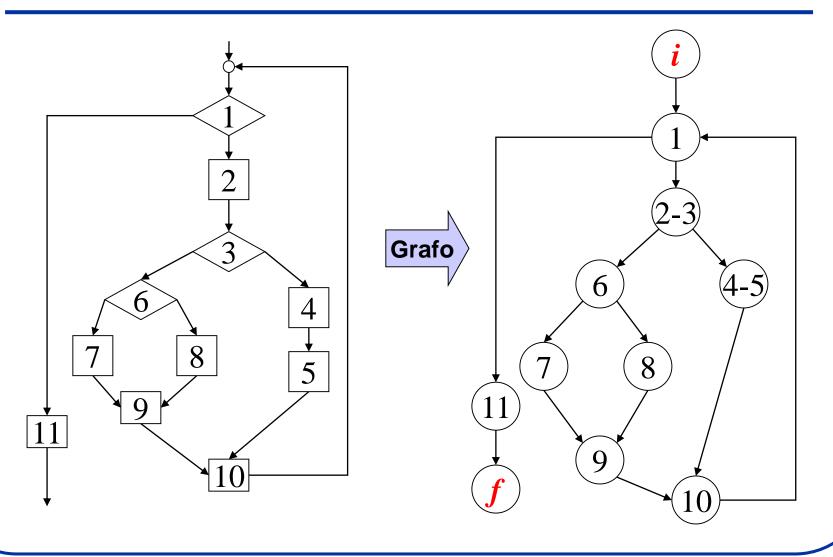


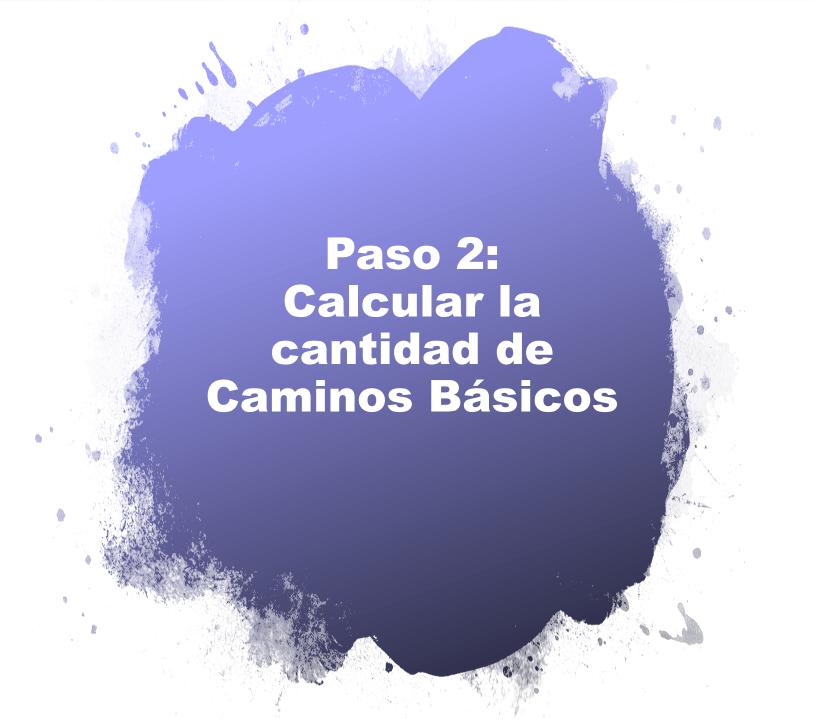
Grafo





Ejemplo 2: diagrama de flujo de código





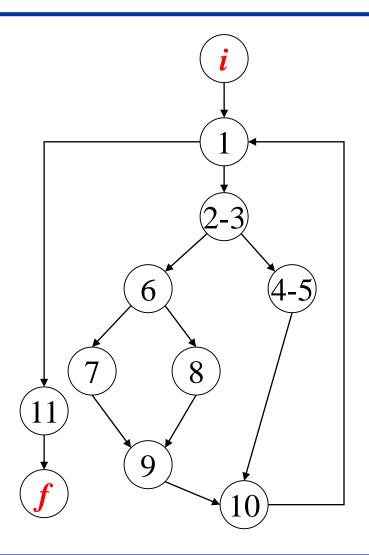


Cantidad de caminos básicos

- Verificar que se cumplen las reglas de los grafos
- Contar la cantidad de:
 - Arcos (*nArcos*)
 - Nodos (*nNodos*)
- Calcular la cantidad de caminos básicos (nCB)
 - Fórmula → nCB = nArcos nNodos + 2



Ejemplo

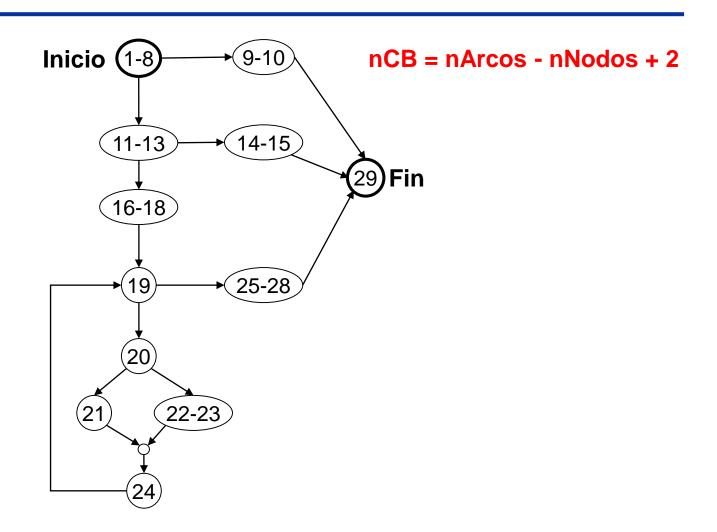


- Contar
 - nArcos = 13
 - nNodos = 11

- Cant. Cam. Básicos
 - nArcos nNodos + 2
 - 13 11 + 2
 - 4 Caminos Básicos



Ejercicio en parejas: Calcula cant. de caminos básicos

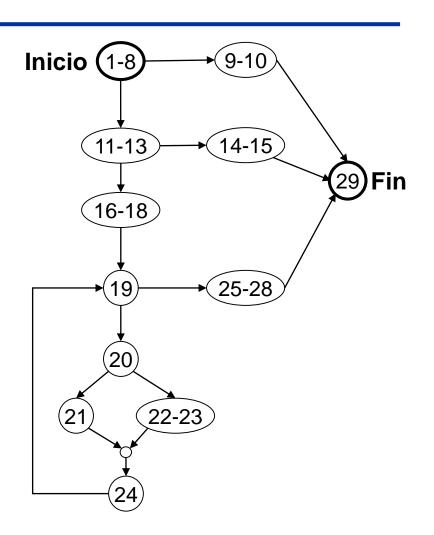




Respuesta

• nArcos = 16nNodos = 13

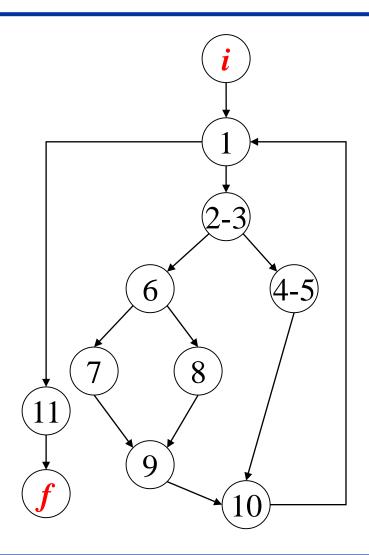
- nCB = 16 13 + 2 = 5
- 5 caminos básicos



Paso 3: Encontrar los Caminos Básicos



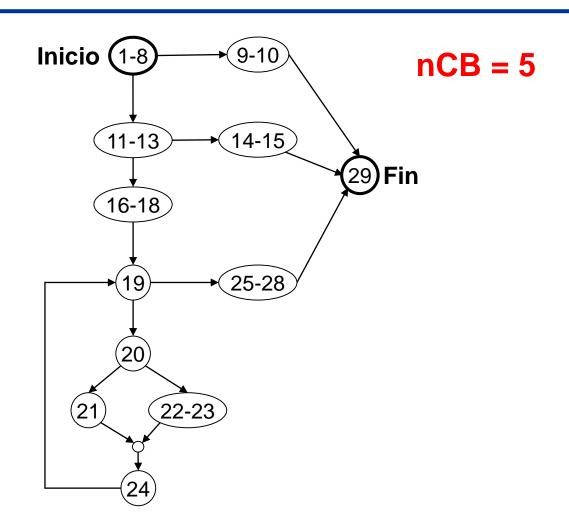
Ejemplo



- nCB = 4
- Caminos básicos:
 - 1, 11
 - 1, 2-3, 6, 7, 9, 10, 1, 11
 - 1, 2-3, 6, 8, 9, 10, 1, 11
 - 1, 2-3, 4-5, 10, 1, 11



Ejercicio en parejas: Encontrar los caminos básicos

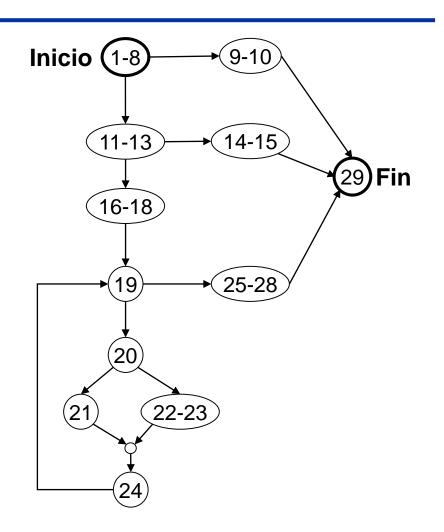




Respuesta

Caminos básicos:

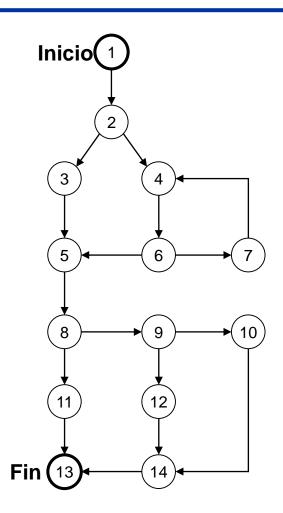
- 1) 1-8, 9-10, 29
- 2) 1-8, 11-13, 14-15, 29
- 3) 1-8, 11-13, 16-18, 19, 25-28, 29
- 4) 1-8, 11-13, 16-18, 19, 20, 21, 24, 19, 25-28, 29
- 5) 1-8, 11-13, 16-18, 19, 20, 22-23, 24, 19, 25-28, 29





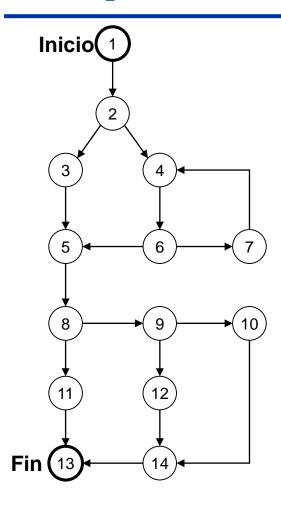
Ejercicio en parejas

Encontrar y escribir los caminos básicos de este grafo:





Respuesta



- nCB = 17 A 14 N + 2 = 5
- Regla bloques independientes:
 - Fijar un camino del 1^{er} bloque
 - Recorrer los caminos del 2º
 - Fijar un camino del 2º bloque
 - Recorrer los caminos del 1º
- Respuesta:
 - <u>1, 2, 3, 5,</u> 8, 11, 13
 - <u>1, 2, 3, 5</u>, 8, 9, 12, 14, 13
 - <u>1, 2, 3, 5</u>, 8, 9, 10, 14, 13
 - 1, 2, 4, 6, 5, <u>8, 11, 13</u>
 - 1, 2, 4, 6, 7, 4, 6, 5, <u>8, 11, 13</u>



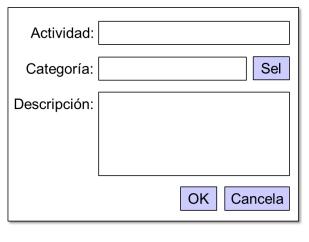


Para cada camino básico

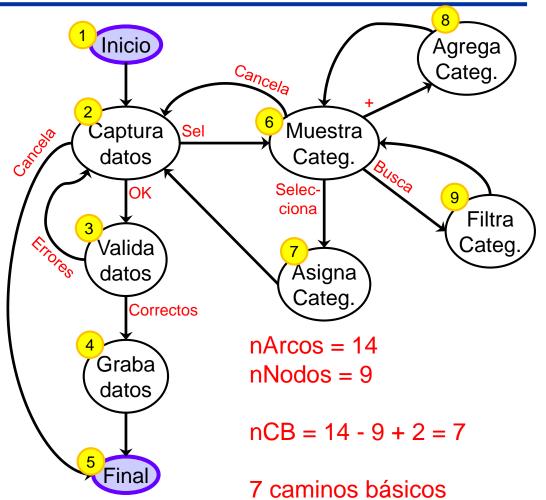
- Escoger "entradas" que obliguen al programa a seguir el camino
- Calcular las salidas esperadas utilizando los requerimientos



Ejemplo 1: eventos en una pantalla









Ejemplo 2

Caminos básicos:

- 1) 1-8, 9-10, 29
- 2) 1-8, 11-13, 14-15, 29
- 3) 1-8, 11-13, 16-18, 19, 25-28, 29
- 4) 1-8, 11-13, 16-18, 19, 20, 21, 24, 19, 25-28, 29
- 5) 1-8, 11-13, 16-18, 19, 20, 22-23, 24, 19, 25-28, 29

```
1. #include <fstream.h>
2. #include <iostream.h>
#include <stdlib.h>
4. int main() {
5.
       ifstream fin:
       ofstream fout:
       fin.open("cpub.dat");
8.
       if (fin.fail()) {
            cout << "Error en entrada.\n";
10.
            exit(1);
11.
12.
       fout.open("cmaspub.dat");
13.
       if (fout.fail()) {
            cout << "Error en salida.\n":
14.
15.
            exit(1);
16.
17.
       char siguiente;
18.
       fin.get(siguiente);
19.
       while (! fin.eof()) {
20.
            if (siguiente == 'C')
21.
                 fout << "C++":
22.
            else
23.
                 fout << siguiente;
24.
            fin.get(siguiente);
25.
26.
       fin.close();
27.
       fout.close();
28.
       return 0:
```

29.}



Posible respuesta

ID	Instrucciones	Resultado Esperado
1	- Renombrar cpub.dat → x.x	mensaje "error en entrada"
2	Renombrar x.x → cpub.datHacer read-only cmaspub.dat	mensaje "error en salida"
3	Quitar read-only a cmaspub.datCopiar c0.dat (archivo vacío)a cpub.dat	cmaspub.dat debe estar vacío
4	- Copiar c1.dat (archivo con una "C") a cpub.dat	cmaspub.dat debe contener solo es string "C++"
5	- Copiar c2.dat (archivo con una "R") a cpub.dat	cmaspub.dat debe contener solo la letra "R"