



Ingeniería del Software

EJERCICIO 5 - Unidad 5: Pruebas

EJERCICIO 5

Dada la siguiente función “C”:

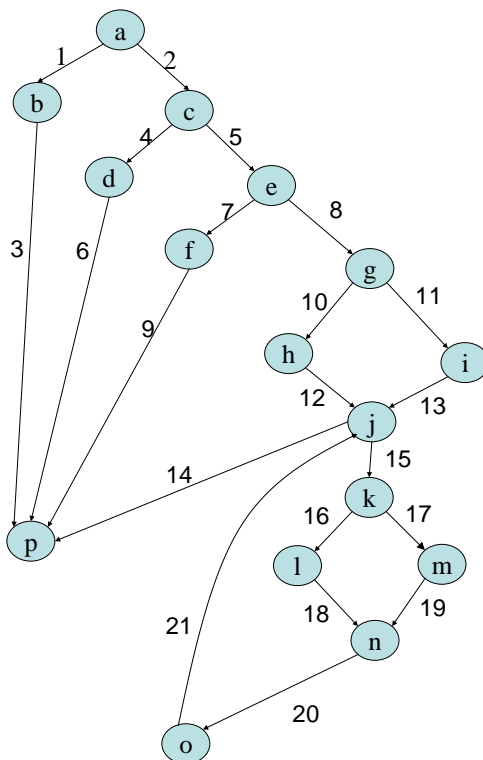
```
#include <stdio.h>
#include <math.h>

#define PRECISION 1e-6

double calculaRaizCuadrada ( double num )
/* calcula la raiz cuadrada de un número positivo iterativamente */
{
    double sup, inf, raiz, division;
```

a	if (num<0)
b	raiz = -1;
c	else if (num==0)
d	raiz = 0;
e	else if (num==1)
f	raiz = 1;
	else
	{
g	if (num > 1)
	{
h	sup = num/2.0;
	raiz = num/3.0;
	}
	else
	{
i	sup = 1.0;
	raiz = 0.5;
	}
j	inf = 0.0;
	division = num/raiz;
	while (fabs(raiz-division)>PRECISION)
	{
k	if (raiz*raiz > num)
l	sup = raiz;
m	else inf = raiz;
n	raiz = (sup+inf)/2.0;
	division = num/raiz;
o	}
	}
p	return raiz;
	}

a) Dibuja el grafo de flujo ¿Cuál es la complejidad ciclomática?



NOTA1: El nodo O puede desaparecer, al igual que las aristas 20 y 21 y, en su lugar, poner una arista de n a j.

NOTA2: Convendría dividir el nodo j en dos para separar el while.

$$V(G) = \text{Aristas} - \text{Nodos} + 2 = 21 - 16 + 2 = 7$$

b) ¿Cuáles son los caminos básicos?

- Caso 1: 1-3
- Caso 2: 1-2-4-6
- Caso 3: 1-2-5-7-9
- Caso 4: 1-2-5-8-10-12-14
- Caso 5: 1-2-5-8-11-13-14
- Caso 6: 1-2-5-8-10-12-15-16-18-20-21-14
- Caso 7: 1-2-5-8-11-13-15-17-19-20-21-14

c) Prepara un conjunto de casos de pruebas que ejerciten todos los caminos básicos.

- Caso 1: num= -4, raiz=-1
- Caso 2: num= 0, raiz=0
- Caso 3: num= 1, raiz=1
- Caso 4: num= 9, raiz=3 (sup=9/2, raiz=9/3, division=3)
- Caso 5: num= 0.25, raiz=0.5 (num/0,5=0,5 => num=0,25)
- Caso 6: num= 36, raiz=6 (sup=n/2, raiz=n/3, div=n/3, bucle::raiz=(n/3)/2=n/6, Div=n/(n/6)=6, n/6-6=0 => n=36)
- Caso 7: num=0.5625, raiz=0.75 (Sup=1.0, raiz=0,5, inf=0, div=num/0.5 condición:: 0.5*0.5 <= num, Bucle:: inf=0.5, raiz=(1+0.5)/2=0.75 , div=num/0.75 => 0.75-num/0.75=0 => num=0.75²)