Module 3. Kỹ thuật hộp trắng

Bài 1. Vẽ lược đồ CFG cho đoạn mã sau:

1. int getPositionOf2Cir(int firstRadius, int secRadius, int distance)

2. START

3. If distance = 0

4. If firstRadius = secRadious 5. return 0

6 .Else If firstRadius < secRadious

7. return 1

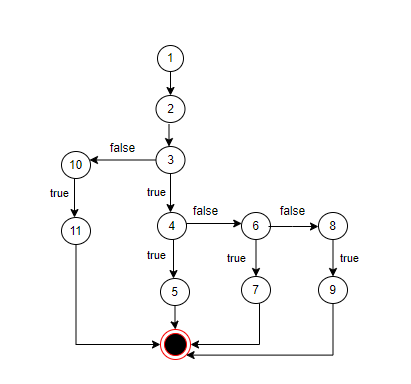
8.Else If firstRadius > secRadious

9. return 2

10. If distance > 0

11. return 3

12. END



Bài 2. Vẽ lược đồ CFG cho đoạn mã sau:

1. X,Y,Z is integer

2. START

3. Check the value of X input

4. If is 1 or 2

5. Return A

6. Else

7. Check the value of Y input

8. If Y <=10

9. Return B

10. Else

11. Check the value of Z input

12. If Z<5

13. Return C

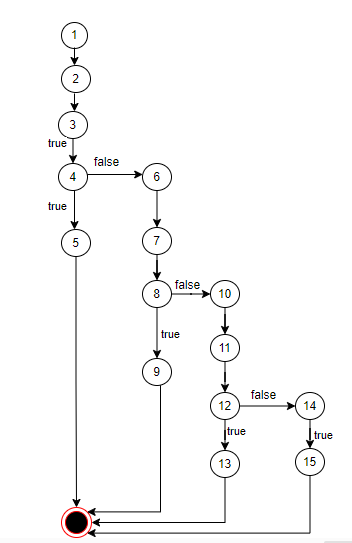
14. Else 15. Return D

16. . End If

17 End If

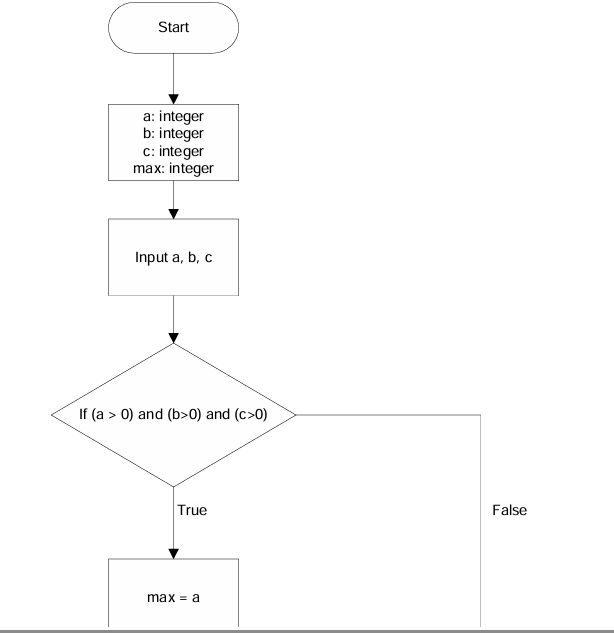
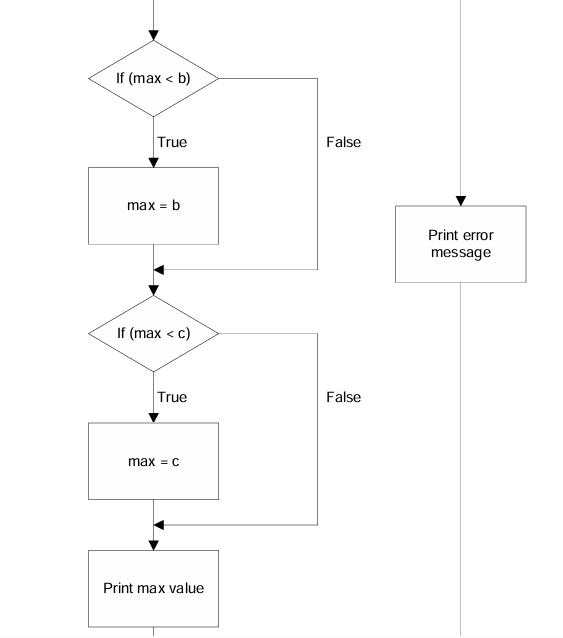
18. End If

19. END



Bài 3. Thiết kế các test case sao cho lược đồ sau đạt bao phủ nhánh 100%. Đây là thuật toán tìm số lớn nhất trong 3 số a,b,c.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Condition | Input | Path | Expect result |
| 1 | a>0,b>0,c>0(T)  max<b(T)  max<c(T) | a=3,b=4,c=5 | 0-1-2-3-4-5-6-7-8-9-10 | 5 |
| 2 | a>0,b>0,c>0(T)  max<b(F)  max<c(F) | a=5,b=4,c=3 | 0-1-2-3-4-5-7-9-10 | 5 |
| 3 | a>0,b>0,c>0(T)  max<b(T)  max<c(F) | a=3,b=5,c=4 | 0-1-2-3-4-5-6-7-9-10 | 5 |
| 4 | a>0,b>0,c>0(F) | a=-3,b=-4,c=-5 | 0-1-2-3-10 | Error |

Bài 4. Cho hàm Triangle kiểm tra tam giác là cân, đều hay tam giác thường sau đây. Hàm trả về "Not a Triangle" nếu không phải tam giác; "Triangle is Scalene" nếu là tam giác thường; "Triangle is Isosceles" nếu là tam giác cân; "Triangle is Equilateral" nếu là tam giác đều. Viết các test case để đạt được 100% độ bao phủ lệnh, 100% độ bao phủ nhánh.

**1. String Triangle( int a, int b, int c ) {**

**2. int match = 0;**

**3. if (a == b)**

**4. match = match + 1;**

**5. if (a == c)**

**6. match = match + 2;**

**7. if (b == c)**

**8. match = match + 3;**

**9. if (match == 0)**

**10. if ((a + b) <= c)**

**11. return ("Not a Triangle");**

**12. else if ((b + c) <= a)**

**13. return ("Not a Triangle");**

**14. else if ((a + c) <= b)**

**15. return ("Not a Triangle");**

**16. else return ("Triangle is Scalene");**

**17. else if (match == 1)**

**18. if ((a + c) <= b)**

**19. return ("Not a Triangle");**

**20. else return ("Triangle is Isosceles");**

**21. else if (match == 2)**

**22. if ((a + c) <= b)**

**23. return ("Not a Triangle");**

**24. else return ("Triangle is Isosceles");**

**25. else if (match == 3)**

**26. if ((b + c) <= a)**

**27. return ("Not a Triangle");**

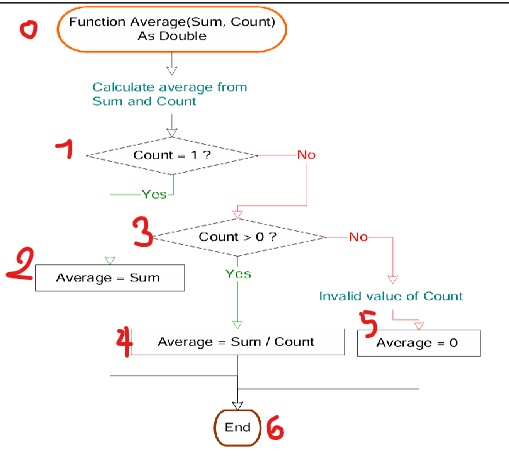
**28. else return ("Triangle is Isosceles");**

**29. else return ("Triangle is Equilateral");**

**30. }**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Condition | Input | Path | Expect result |
| 1 | Int a, int b, int c (T)  a==b (F)  a==c (F)  b==c (F)  match = 0 (T)  (a+b)<=c (T)  (b+c)<=a (F)  (a+c)<=b (F)  match == 1 (F)  (a+c)<=b (F)  match == 2 (F)  match == 3 (F) | a=1,b=2,c=4 | 1-2-3-5-7-9-10-11-30 | "Not a Triangle" |
| 2 | Int a, int b, int c (T)  a==b (F)  a==c (F)  b==c (F)  match = 0 (T)  (a+b)<=c (F)  (b+c)<=a (F)  (a+c)<=b (T)  match == 1 (F)  (a+c)<=b (F)  match == 2 (F)  match == 3 (F) | a=1,b=4,c=2 | 1-2-3-5-7-9-10-12-14-15-30 | "Not a Triangle" |
| 3 | Int a, int b, int c (T)  a==b (F)  a==c (F)  b==c (F)  match = 0 (T)  (a+b)<=c (F)  (b+c)<=a (T)  (a+c)<=b (F)  match == 1 (F)  (a+c)<=b (F)  match == 2 (F)  match == 3 (F) | a=4,b=1,c=2 | 1-2-3-5-7-9-10-12-13-30 | "Not a Triangle" |
| 4 | Int a, int b, int c (T)  a==b (F)  a==c (F)  b==c (F)  match = 0 (F)  (a+b)<=c (F)  (b+c)<=a (F)  (a+c)<=b (F)  match == 1 (T)  (a+c)<=b (F)  match == 2 (F)  match == 3 (F) | a=3,b=3,c=4 | 1-2-3-5-7-9-17-18-20-30 | "Triangle is Isosceles" |
| 5 | Int a, int b, int c (T)  a==b (T)  a==c (T)  b==c (T)  match = 0 (F)  (a+b)<=c (F)  (b+c)<=a (F)  (a+c)<=b (F)  match == 1 (F)  (a+c)<=b (F)  match == 2 (F)  match == 3 (F) | a=2,b=2,c=2 | 1-2-3-5-7-9-17-21-25-29-30 | "Triangle is Equilateral" |
| 6 | Int a, int b, int c (T)  a==b (F)  a==c (F)  b==c (F)  match = 0 (T)  (a+b)<=c (F)  (b+c)<=a (F)  (a+c)<=b (F)  match == 1 (F)  (a+c)<=b (F)  match == 2 (F)  match == 3 (F) | a=3,b=4,c=5 | 1-2-3-5-7-9-10-12-14-16-30 | "Triangle is Scalene" |
| 7 | Int a, int b, int c (F)  a==b (F)  a==c (F)  b==c (F)  match = 0 (F)  (a+b)<=c (F)  (b+c)<=a (F)  (a+c)<=b (F)  match == 1 (F)  (a+c)<=b (F)  match == 2 (F)  match == 3 (F) | a=-2,b=3,c=4 | 1-30 | "Not a Triangle" |

Bài 5. Cho hàm tính trung bình Sum/Count theo lược đồ sau. Nếu Count<=0 hàm trả về 0. Thiết kế các test case sao cho hàm đạt bao phủ nhánh 100%.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Condition | Input | Path | Expect result |
| 1 | Count = 1 (F)  Count > 0 (F) | Sum = 10, Count = 0 | 0-1-3-5-6 | 0 |
| 2 | Count = 1 (T)  Count > 0 (F) | Sum = 10, Count = 1 | 0-1-2-6 | 10 |
| 3 | Count = 1 (F)  Count > 0 (T) | Sum = 10, Count = 5 | 0-1-3-4-6 | 2 |