

## CÁC TIÊU CHÍ BAO PHỦ MÃ NGUỒN

### VI. Tiêu chí bao phủ lộ trình độc lập – Basic Path Coverage

Thiết kế Testcase sao cho đảm bảo thực thi được tất cả các lộ trình có trong CFG. Hai lộ trình độc lập là 2 lộ trình nếu lt1 có cung này mà lt2 không có (HOẶC ngược lại).

**Bước 1:** Tính số lộ trình độc lập

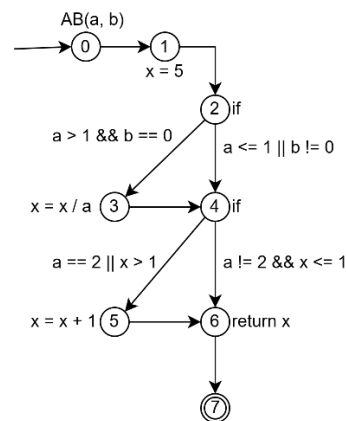
- Số đỉnh quyết định + 1
- Số cung - Số đỉnh + 2
- Số miền kín + 1

**Bước 2:** Chọn lộ trình bao phủ được nhiều đỉnh quyết định nhất

**Bước 3:** Thay cung tại từng đỉnh quyết định để tìm ra các cung còn lại

#### Ví dụ 1:

```
int AB (int a, int b){  
    int x = 5;  
    if ((a > 1) && (b == 0)){  
        x = x / a;  
    }  
    if((a == 2) || (x > 1)){  
        x = x + 1;  
    }  
    return x;  
}
```



Số lộ trình độc lập:  $C = 2 + 1 = 3$

Lộ trình		Data test	Out put
1	01234567	a = 2, b = 0	x = 3
2	0124567	a = 1, b = 1	x = 6
3	0123467	a = 5, b = 1	x = 1

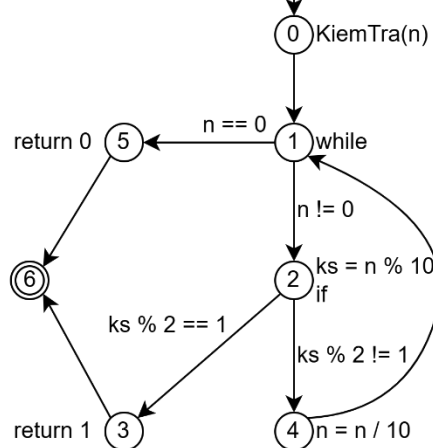
Basic paths = {01234567, 0124567, 0123467}

#### Ví dụ 2:

```

char KiemTra(int n){
    char ks;
    while(n != 0){
        ks = n % 10;
        if(ks % 2 == 1)
            return 1;
        n = n / 10;
    }
    return 0;
}

```



Số lộ trình độc lập:  $C = 2 + 1 = 3$

	Lộ trình	Data test	Out put
1	01241236	$n = 13$	return 1
2	01236	$n = 3$	return 1
3	0124156	$n = 4$	return 0

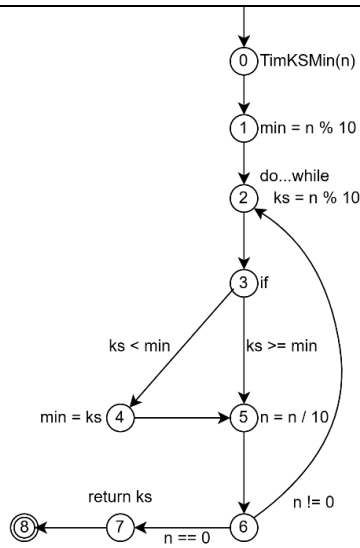
Basic paths = {01241236, 01236, 0124256}

### Ví dụ 3:

```

char TimKSMin (int n){
    char ks;
    min = n % 10;
    do{
        ks = n % 10;
        if(ks < min){
            min = ks;
        }
        n = n / 10;
    }
    while(n != 0);
    return ks;
}

```



### 1. Lộ trình kiểm thử tối thiểu thỏa tiêu chí bao phủ quyết định – TC2:

Path: 0123562345678

Input:  $n = 13$

Expected Output:  $ks = 1$

Actual Output:  $ks = 1$

## 2. Xác định tập lộ trình độc lập:

Số lộ trình độc lập:  $C = 2 + 1 = 3$

	Lộ trình	Data test	Out put
1	01235678	$n = 5$	$ks = 5$
2	0123562345678	$n = 13$	$ks = 1$
3	012356235678	$n = 33$	$ks = 3$

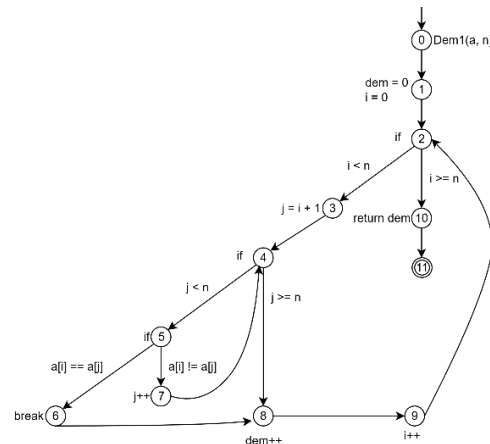
Basic paths = {01235678, 0123562345678, 012356235678}

## Ví dụ 4:

```

int Dem1(int a[], int n){
    int dem = 0;
    for(int i = 0; i < n; i++){
        for(int j = i + 1; j < n; j++){
            if(a[i] == a[j])
                break;
        }
        dem++;
    }
    return dem;
}

```



## 1. Theo tiêu chí 1

Lộ trình 01234568923457489234892(10)(11) là lộ trình khả thi tối thiểu bao phủ được các đỉnh trong CFG.

Input:  $a = [1; 1; 2]$        $n = 3$

Output:

- dem = 2 (ER)

- dem = 3 (AR)

## 2. Xác định tập lộ trình độc lập

Xác định Basic Paths:  $C = 3 + 1 = 4$

	Lộ trình	Data test	Out put (AR)
1	01234574892(10)(11)	$n = 2, a = [1; 3]$	$dem = 2$
2	012(10)(11)	$n = 0, a = []$	$dem = 0$
3	01234892(10)(11)	$n = 1, a = [1]$	$dem = 1$
4	0123456892(10)(11)	$n = 2, a = [2; 2]$	$dem = 2$

Basic paths = {01234574892(10)(11), 012(10)(11), 01234892(10)(11), 0123456892(10)(11)}