

## Bài tập kiểm thử dòng điều khiển

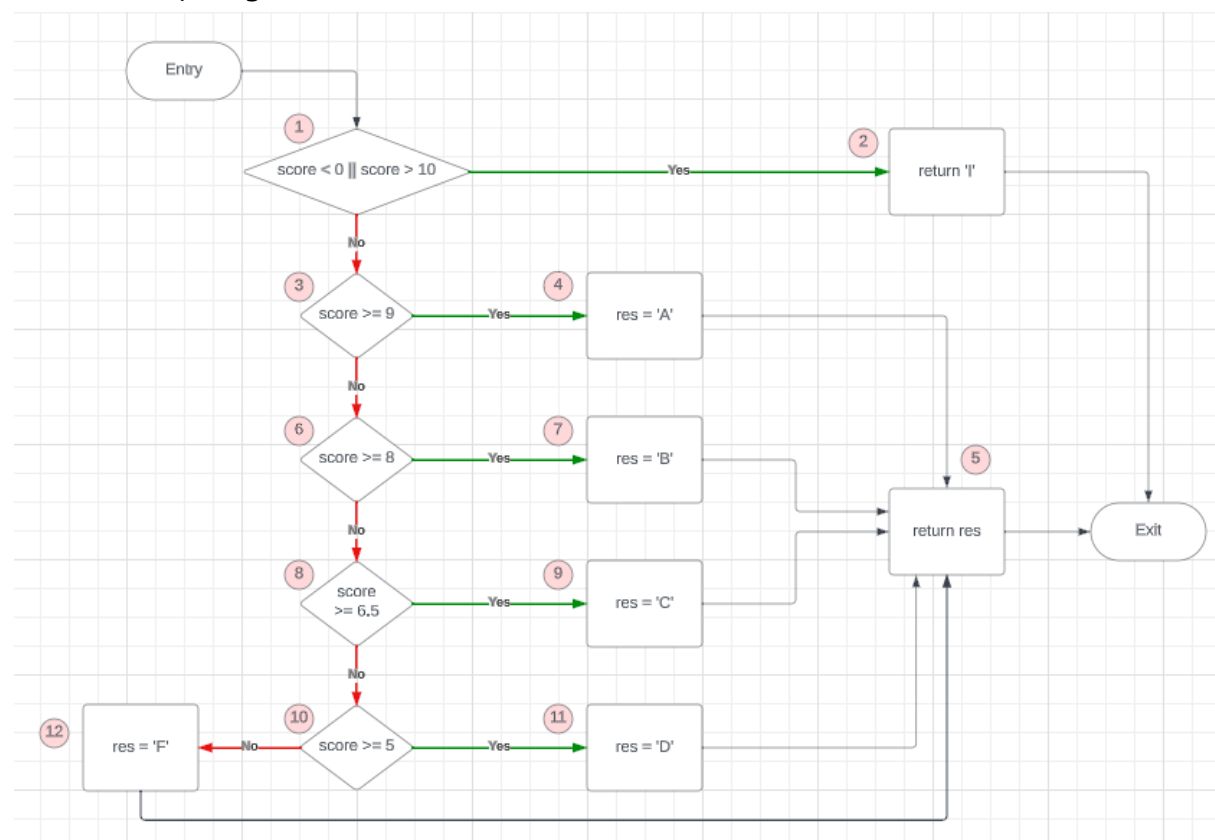
Họ và tên	Phạm An Đức Vinh
Mã sinh viên	21020097

### Phần I: Các bài tập trong slide

**Bài 1:** Các bước nhằm kiểm thử một đơn vị chương trình theo phương pháp kiểm thử dòng điều khiển với một độ đo kiểm thử cho trước:

- Xây dựng đồ thị dòng điều khiển( control flow graph) tương ứng từ mã nguồn của chương trình/ đơn vị chương trình và độ đo kiểm thử( C1, C2 hoặc C3). Đồ thị dòng điều khiển là một đồ thị có hướng gồm các đỉnh tương ứng với các câu lệnh/nhóm câu lệnh và các cạnh là các dòng điều khiển giữa các câu lệnh/nhóm câu lệnh. Nếu i và j là các đỉnh của đồ thị dòng điều khiển thì tồn tại một cạnh từ i đến j nếu lệnh tương ứng với j có thể được thực hiện ngay sau lệnh tương ứng với i.
- Xác định các đường đi( test paths) tương ứng: Các đường đi của chương trình (xuất phát từ điểm bắt đầu, đi qua các đỉnh của đồ thị và kết thúc ở điểm cuối) được xác định sao cho khi chúng được thực hiện thì độ đo kiểm thử tương ứng được thỏa mãn. Dựa trên ý tưởng của T. J. McCabe [McC76b,WM96], số đường đi chương trình ứng với đồ thị dòng điều khiển của nó được tính bằng một trong các phương pháp sau:
  - Số cạnh – số đỉnh + 2
  - Số đỉnh quyết định + 1
- Sinh các ca kiểm thử tương ứng với mỗi đường đi đã được xác định
- Thực hiện các ca kiểm thử trên những test case đã sinh
- So sánh system output với expected output (dựa trên đặc tả yêu cầu)

**Bài 2:** Đồ thị dòng điều khiển cho hàm Grade



Độ phức tạp Cyclomatic  $C = 5 + 1 = 6$

Với độ đo C1: phủ hết tất cả các lệnh

ID	Path	Input	Output
1	Entry -> 1(Y) -> 2 -> exit	15	'I'
2	Entry -> 1(N) -> 3(Y) -> 4 -> 5 -> exit	10	'A'
3	Entry -> 1(N) -> 3(N) -> 6(Y) -> 7 -> 5 -> exit	8	'B'
4	Entry -> 1(N) -> 3(N) -> 6(N) -> 8(Y) -> 9 -> 5 -> exit	7	'C'
5	Entry -> 1(N) -> 3(N) -> 6(N) -> 8(N) -> 10(Y) -> 11 -> 5 -> exit	6	'D'
6	Entry -> 1(N) -> 3(N) -> 6(N) -> 8(N) -> 10(N) -> 12 -> 5 -> exit	1	'F'

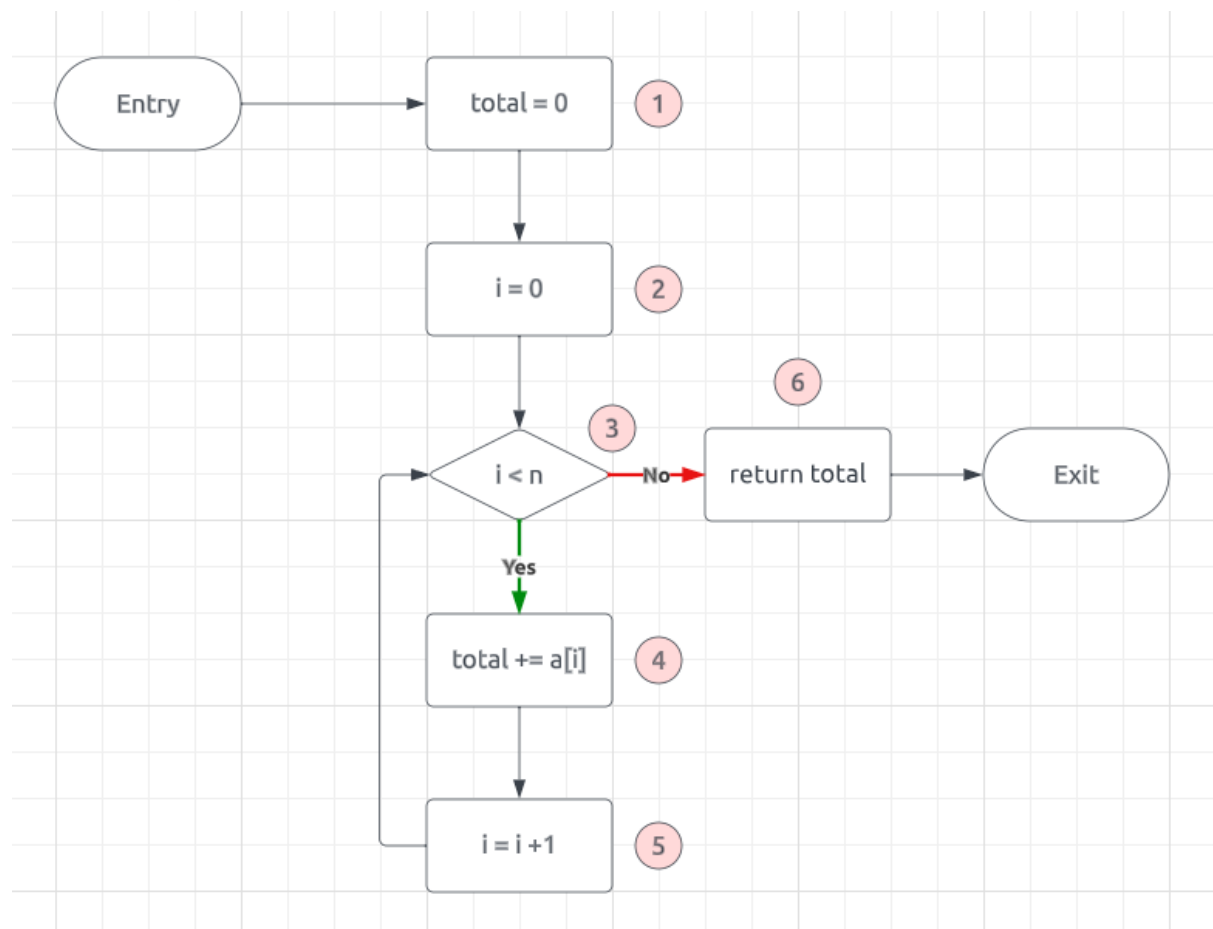
Với độ đo C2: phủ hết các nhánh

Dễ thấy các đường đi trong trường hợp kiểm thử với độ đo C1 ở trên đã phủ hết các nhánh

-> Các đường đi cho mức đo C2 giống với các đường đi với độ đo C1

-> Vậy có thể sử dụng các test case ở trên để kiểm thử cho mức đo C2

### Bài 3: Đồ thị điều khiển hàm Sum



Sinh đường đi và ca kiểm thử với độ đo C1, C2:

Ta thấy qua flowchart của hàm `sum(int a[], int n)`: các đường đi để phủ hết các lệnh cũng có thể phủ hết các nhánh.

-> Các đường đi sinh cho ca kiểm thử với độ đo C1 có thể dùng cho ca kiểm thử với độ đo C2

-> Danh sách các đường đi và ca kiểm thử với độ đo C1 + C2 là:

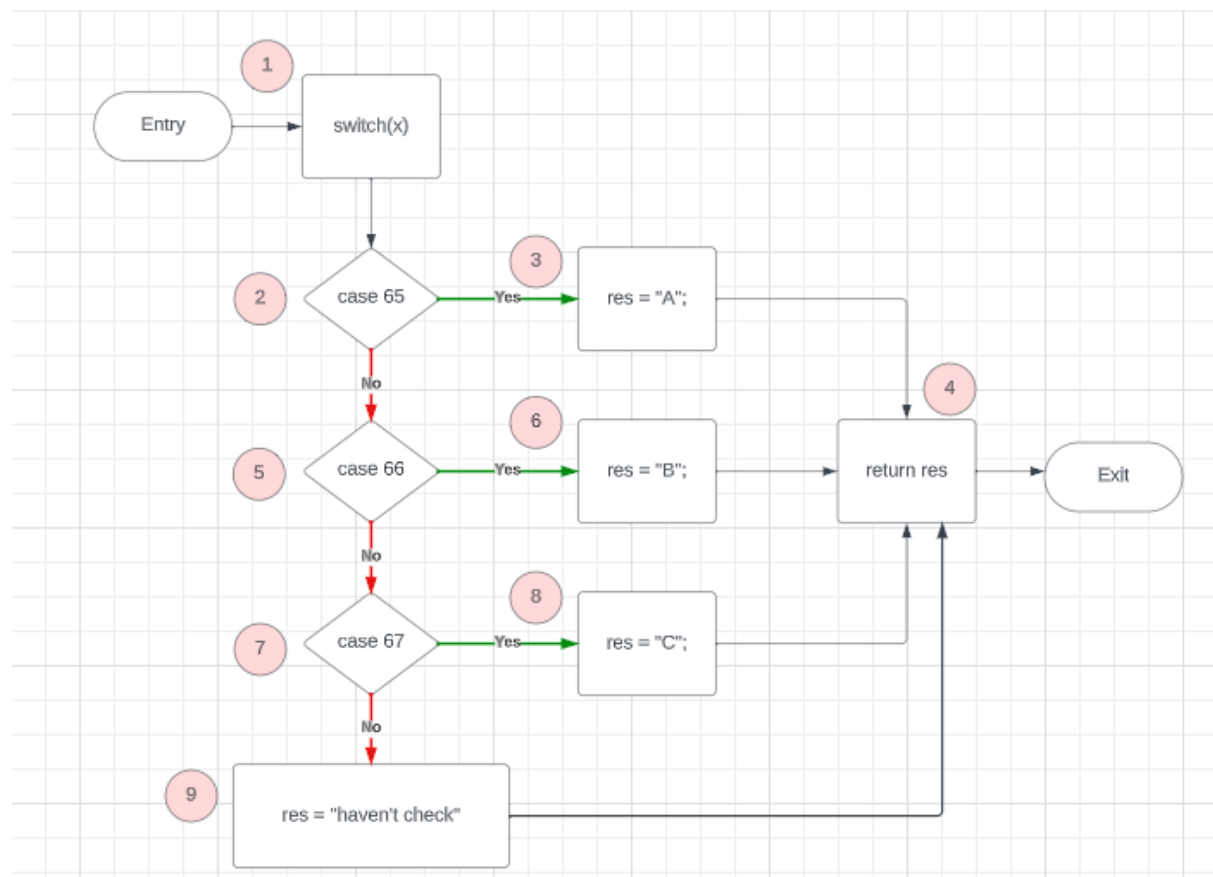
Test ID	Path	Input	Output
1	Entry->1->2->3(Y)->4->5->3(N)->6->Exit	a[] = { 10, 11, 12, 21 }, n = 1	10

Sinh các ca kiểm thử cho vòng lặp for:

Số vòng lặp = k (kiểu int) -> số vòng lặp tối đa = MAXINT =  $2^{31} - 1$  (nếu sinh test case cho trường hợp k = MAXINT - 1; k = MAX INT; ... thì size(a) >= k -> kích thước mảng a rất lớn -> không thực hiện test case này)

Test ID	Số lần thực hiện vòng lặp	Path	Input	Output
1	0	Entry -> 1 -> 2 -> 3(N) -> 6 -> Exit	a[]={2,4,5,6,8,10,12}, n=0	0
2	1	Entry -> 1 -> 2 -> (3(Y) -> 4 -> 5)^1 -> 3(N) -> 6 -> Exit	a[]={2,4,5,6,8,10,12}, n=1	2
3	2	Entry -> 1 -> 2 -> (3(Y) -> 4 -> 5)^2 -> 3(N) -> 6 -> Exit	a[]={2,4,5,6,8,10,12}, n=2	6
4	5	Entry -> 1 -> 2 -> (3(Y) -> 4 -> 5)^5 -> 3(N) -> 6 -> Exit	a[]={2,4,5,6,8,10,12}, n=5	25

#### Câu 4: Đồ thị điều khiển hàm foo



Kiểm thử với mức đo C2: Lựa chọn các đường đi sao cho mỗi nhánh đều được đi qua ít nhất một lần

Độ phức tạp Cyclomatic  $C = P + 1 = 3 + 1 = 4$

Test ID	Path	Input	Output
1	Entry -> 1 -> 2(Y) -> 3 -> 4 -> Exit	65	'A'
2	Entry -> 1 -> 2(N) -> 5(Y) -> 6 -> 4 -> Exit	66	'B'
3	Entry -> 1 -> 2(N) -> 5(N) -> 7(Y) -> 8 -> 4 -> Exit	67	'C'
4	Entry -> 1 -> 2(N) -> 5(N) -> 7(N) -> 9 -> 4 -> Exit	20	'haven't check'

## Phần II: Bài tập kiểm thử chương trình riêng với độ đo C2

**Mô tả bài toán:** Tính toán số tiền mà khách hàng phải trả dựa trên bảng giá có sẵn với đầu vào là số điện đã tiêu thụ, kết quả trả về là số tiền tương ứng với số điện mà khách hàng đã tiêu thụ.

Input	Output
<ul style="list-style-type: none"><li>Số điện tiêu thụ: <math>W</math> (<math>W \in \mathbb{N}</math>; <math>0 \leq W &lt; 2^{31}</math>)</li></ul>	Số tiền phải trả

Bảng định mức sử dụng như sau:

Bậc	Định mức (kWh)	Tiền phải trả (đồng)
1	0 - 50	1678
2	51 - 100	1734
3	101 - 200	2014
4	201 - 300	2536
5	301 - 400	2834
6	> 400	2927

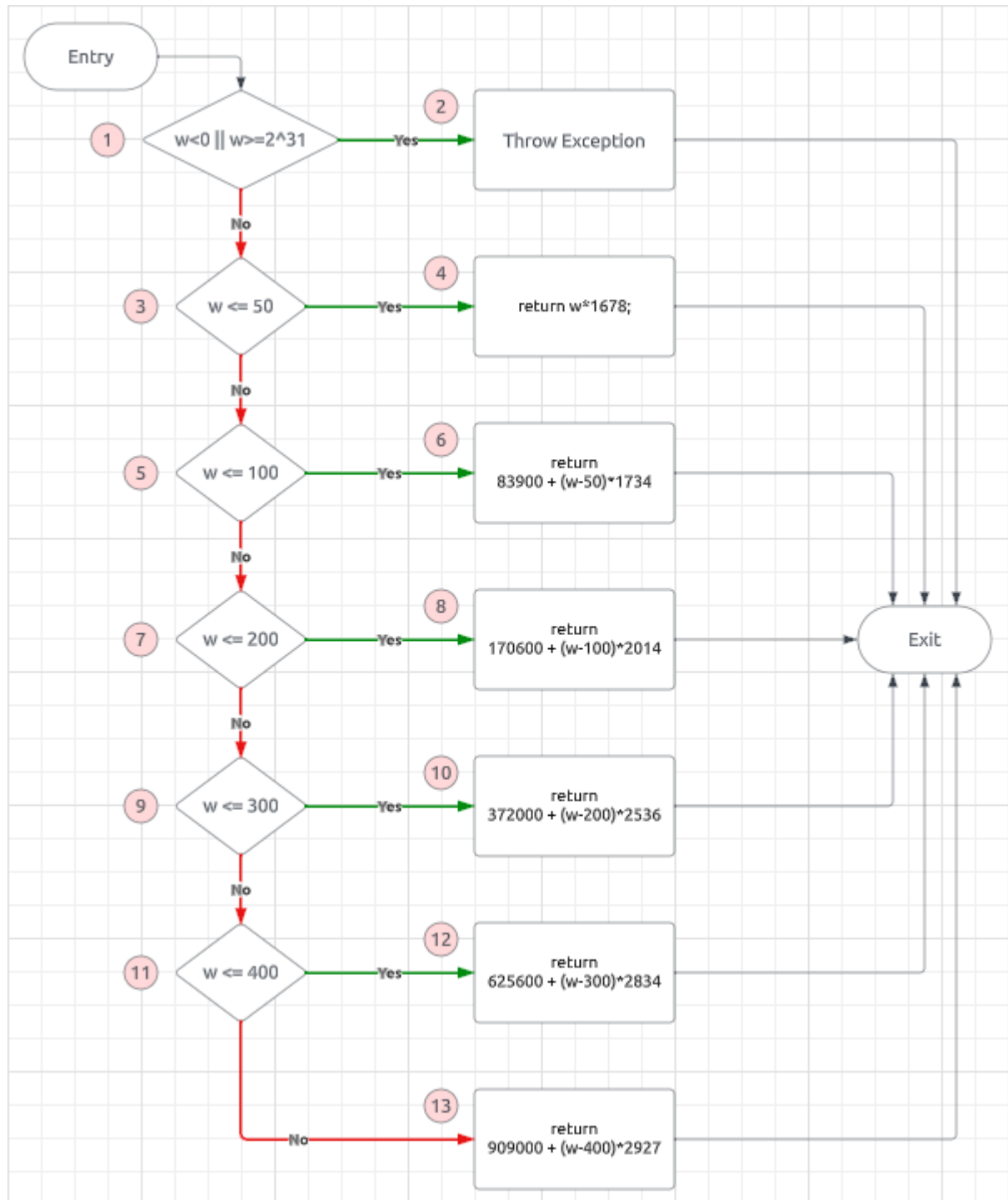
### Yêu cầu:

- Trả về số tiền phải trả với số điện đã sử dụng
- Throw Exception với trường hợp không hợp lệ input không đúng như mô tả

### Source và test case:

[https://github.com/vinhpap/INT3117/tree/main/week4/source\\_code\\_and\\_test\\_case](https://github.com/vinhpap/INT3117/tree/main/week4/source_code_and_test_case)

## Đồ thị điều khiển chương trình



Độ phức tạp Cyclomatic  $C = P + 1 = 6 + 1 = 7$

Test ID	Path	Test case
1	Entry -> 1(Y) -> 2 -> Exit	w = -1
2	Entry -> 1(N) -> 3(Y) -> 4 -> Exit	w = 49
3	Entry -> 1(N) -> 3(N) -> 5(Y) -> 6 -> Exit	w = 100
4	Entry -> 1(N) -> 3(N) -> 5(N) -> 7(Y) -> 8 -> Exit	w = 199
5	Entry -> 1(N) -> 3(N) -> 5(N) -> 7(N) -> 9(Y) -> 10 -> Exit	w = 300
6	Entry -> 1(N) -> 3(N) -> 5(N) -> 7(N) -> 9(N) -> 11(Y) -> 12 -> Exit	w = 399
7	Entry -> 1(N) -> 3(N) -> 5(N) -> 7(N) -> 9(N) -> 11(N) -> 13 -> Exit	w = 500

Test ID	Input(w)	Output	Expected Output	Status
1	w = -1	Throw Exception	Throw Exception	PASS
2	w = 49	82222	82222	PASS
3	w = 100	170600	170600	PASS
4	w = 199	369986	369986	PASS
5	w = 300	625600	625600	PASS
6	w = 399	906166	906166	PASS
7	w = 500	1201700	1201700	PASS