**A diagram of a flowchart

Description automatically generatedĐinh Văn Thạch – 21020789**  
Bài 1:  
 Xây dựng đồ thị dòng điều khiển cho hàm Grade ứng với độ đo C1 và C2  
Các đường đi:

1. 0 🡺 1 🡺 2 (T) 🡺 3

2. 0 🡺 1 🡺 2 (F) 🡺 4 (T) 🡺 5 🡺 13

3. 0 🡺 1 🡺 2 (F) 🡺 4 (F) 🡺 6 (T) 🡺 7 🡺 13

4. 0 🡺 1 🡺 2 (F) 🡺 4 (F) 🡺 6 (F) 🡺 8 (T) 🡺 9 🡺 13

5. 0 🡺1 🡺 2 (F) 🡺 4 (F) 🡺6 (F) 🡺 8 (F) 🡺 10 (T) 🡺 11 🡺 13

6. 0 🡺 1 🡺 2 (F) 🡺 4 (F) 🡺 6 (F) 🡺 8 (F) 🡺 10 (F) 🡺 12 🡺 13

Các ca kiểm thử tương ứng:

1. grade(11)

2. grade(9.5)

3. grade(8.5)

4. grade(7)

5. grade(6)

6. grade(4)

Bài 2: Đoạn mã 6.5

A notebook with writing on it

Description automatically generated

Bài 3: Bài string foo()A diagram of a graph

Description automatically generated

Đường đi:  
0🡺 1 🡺 2(T) 🡺 9 ;test case foo(65)

0🡺 1 🡺 2(F)🡺 4(T) 🡺 5 🡺 9 ;test case foo(66)

0🡺 1 🡺 2(F) 🡺 4(F) 🡺 6(T) 🡺 7 🡺 9 ;test case foo(67)

0🡺 1 🡺 2(F) 🡺 4(F) 🡺 6(F) 🡺 8 🡺 9 ;test case foo(100000)

Bài 4: Kiểm thử mức C2 với chương trình của bạn

1. Mô tả bài toán
2. Bài toán: Bài toán tính tiền điện là quá trình tính toán số tiền cần thanh toán dựa trên lượng điện tiêu thụ trong một khoảng thời gian nhất định (chương trình tôi sử dụng khoảng thời gian là **tháng**), số giá điện trong từng mốc KWh sẽ là khác nhau cụ thể:

* Bậc 1: 0 - 50 kWh (1.806 đồng/ kWh).
* Bậc 2: 50 - 100 kWh (1.866 đồng/ kWh).
* Bậc 3: 100 - 200 kWh (2.167 đồng/ kWh).
* Bậc 4: 200 - 300 kWh (2.729 đồng/ kWh).
* Bậc 5: 300 - 400 kWh (3.050 đồng/ kWh).
* Bậc 6: 400 - 500 kWh (3.151 đồng/ kWh).

1. Đề bài: Cho đầu vào là một số nguyên n là số numKWh tiêu thụ trong 1 tháng chủ thể đó (gia đình, tổ chức, doanh nghiệp…). Đầu ra là tiền điện của chủ thể đó phải trả.
2. Ràng buộc:

0 <= numKwh <=20.000

1. Chương trình

public static double CalElecMoney(long numKWh) {  
 long ElecMoneyTotal = 0;  
 if(numKWh < 0 || numKWh > 20000) return -1;  
 // Bậc 1: 0 đến 50 kWh  
 if (numKWh <= 50) {  
 ElecMoneyTotal = numKWh \* 1806;  
 }  
 // Bậc 2: 51 đến 100 kWh  
 else if (numKWh <= 100) {  
 ElecMoneyTotal = 50 \* 1806 + (numKWh - 50) \* 1866;  
 }  
 // Bậc 3: 101 đến 200 kWh  
 else if (numKWh <= 200) {  
 ElecMoneyTotal = 50 \* 1806 + 50 \* 1866 + (numKWh - 100) \* 2167;  
 }  
 // Bậc 4: 201 đến 300 kWh  
 else if (numKWh <= 300) {  
 ElecMoneyTotal = 50 \* 1806 + 50 \* 1866+ 100 \* 2167 + (numKWh - 200) \* 2729;  
 }  
 // Bậc 5: 301 đến 400 kWh  
 else if (numKWh <= 400) {  
 ElecMoneyTotal = 50 \* 1806 + 50 \* 1866 + 100 \* 2167 + 100 \* 2729 + (numKWh - 300) \* 3050;  
 }  
 // Bậc 6: 401 kWh trở lên  
 else {  
 ElecMoneyTotal = 50 \* 1806 + 50 \* 1866 + 100 \* 2167 + 100 \* 2729 + 100 \* 3050 + (numKWh - 400) \* 3151;  
 }  
 return ElecMoneyTotal;  
}

1. Các ca kiểm thử

Hình vẽ ở **trang cuối**.  
Link GitHub [tại đây](https://github.com/ThachNhe/Testing_UET.git):

Các test case được viết trong file TestCase3.java

1. 0🡺1 🡺 2(T) 🡺 3 🡺 15; test case CalElecMoney(-10)
2. 0🡺1 🡺 2(F) 🡺 4(T) 🡺 5 🡺 15; test case CalElecMoney(30)
3. 0🡺1 🡺 2(F) 🡺 4(F) 🡺 6(T) 🡺 7 🡺 15; test case CalElecMoney(74)
4. 0🡺1 🡺 2(F) 🡺 4(F) 🡺 6(F) 🡺 8(T) 🡺 9 🡺 15; test case CalElecMoney(150)
5. 0🡺1 🡺 2(F) 🡺 4(F) 🡺 6(F) 🡺 8(F) 🡺 10(T) 🡺 11 🡺 15; test case CalElecMoney(260)
6. 0🡺1 🡺 2(F) 🡺 4(F) 🡺 6(F) 🡺 8(F) 🡺 10(F) 🡺 12(T) 🡺 13 🡺 15; test case CalElecMoney(355)
7. 0🡺 1 🡺 2(F) 🡺 4(F) 🡺 6(F) 🡺 8(F) 🡺 10(F) 🡺 12(F) 🡺 14 🡺 15; test case CalElecMoney(600)

A diagram of a flowchart

Description automatically generated