**Báo cáo: Kiểm thử dòng dữ liệu**

**Họ và tên**: Đỗ Huy Hoàng

**MSSV**: 21020073

# **Mô tả bài toán:**

* Cho một số lượng nước tiêu thụ của một gia đính trong một tháng được biểu diễn dưới dạng số nguyên (tính bằng ). **Yêu cầu**: Tính số tiền mà hộ gia đính đó phải trả (không tính thuế VAT). Biết rằng giá tiền nước được chia thành bốn bậc sau đây:

|  |  |  |
| --- | --- | --- |
| **STT** | **Lượng nước sử dụng** ( | **Giá tiền** (đồng/) |
| Bậc |  |  |
| Bậc |  |  |
| Bậc |  |  |
| Bậc | Trên |  |

* Ví dụ: Chẳng hạn một hộ gia đính tiêu thụ trong một tháng, thì số tiền mà gia đính đó phải trả là:
  + Bậc 1: giá nước sử dụng trong nước đầu tiên ;
  + Bậc 2: giá nước sinh hoạt từ
  + Bậc 3: giá sử dụng nước trên
  + Bậc 4: giá sử dụng nước trên .
  + Do đó, tổng số tiền mà hộ gia đính đó sẽ phải trả là bậc bậc bậc bậc .

# **Phân tích:**

* Dựa vào [mã nguồn](https://github.com/hoang1803/Software-Testing/blob/main/Ki%E1%BB%83m%20th%E1%BB%AD%20d%C3%B2ng%20d%E1%BB%AF%20li%E1%BB%87u/water.cpp) của chương trình, ta có thể chia thành hai thành phần:
  + Nhập dữ liệu: Sử dụng hàm cin trong C++.
  + Xử lý và in ra kết quả: Sử dụng hàm getPrice của lập trình viên.
* Từ hai thành phần trên, ta có thể thấy rằng cần kiểm thử hàm getPrice là đủ. Bởi vì hàm cin được cung cấp bởi C++, nên chúng sẽ không xảy ra lỗi.
* Dưới đây là mã nguồn của hàm getPrice:

|  |
| --- |
| 1. void getPrice(const int &water) { 2. if (water < 0) { 3. cout << "Invalid Input"; 4. return; 5. } 6. // Tổng giá tiền 7. int totalPrice = 0; 8. // Số nước theo từng bậc mà gia đình do sử dụng 9. int deg1 = 0, deg2 = 0, deg3 = 0, deg4 = 0; 10. if (water <= 10) { 11. deg1 = water \* price[0]; 12. } else if (water <= 20) { 13. deg1 = 10 \* price[0]; 14. deg2 = (water - 10) \* price[1]; 15. } else if (water <= 30) { 16. deg1 = 10 \* price[0], deg2 = 10 \* price[1]; 17. deg3 = (water - 20) \* price[2]; 18. } else { 19. deg1 = 10 \* price[0], deg2 = 10 \* price[1], deg3 = 10 \* price[2]; 20. deg4 = (water - 30) \* price[3]; 21. } 22. totalPrice = deg1 + deg2 + deg3 + deg4; 23. cout << totalPrice; 24. return ; 25. } |

* Từ đoạn mã nguồn trên, ta sẽ xây dựng được đồ thị dòng điều khiển của hàm getPrice với độ phủ C2 như sau:

A diagram of a flowchart

Description automatically generated

* Xác định def, c-use, p-use của từng biến:

|  |  |  |  |
| --- | --- | --- | --- |
| Biến | def | c-use | p-use |
| water |  |  |  |
| totalPrice |  |  |  |
| deg1 |  |  |  |
| deg2 |  |  |  |
| deg3 |  |  |  |
| deg4 |  |  |  |

# **Thiết kế các ca kiểm thử:**

Từ đồ thị luồng điều khiển, ta sẽ thiết kế các đường đi cho các ca kiểm thử như sau:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Biến** | **ID** | **Du-pair** | **Def-clear path** | **Complete path** |
| water | WT1 |  |  |  |
|  | WT2 |  |  |  |
|  | WT3 |  |  |  |
|  | WT4 |  |  |  |
|  | WT5 |  |  |  |
|  | WT6 |  |  |  |
|  | WT7 |  |  |  |
|  | WT8 |  |  |  |
|  | WT9 |  |  |  |
|  | WT10 |  |  |  |
|  | WT11 |  |  |  |
|  | WT12 |  |  |  |
| totalPrice | TP1 |  |  |  |
| deg1 | DG1 |  |  |  |
|  | DG2 |  |  |  |
|  | DG3 |  |  |  |
|  | DG4 |  |  |  |
| deg2 | DG5 |  |  |  |
|  | DG6 |  |  |  |
|  | DG7 |  |  |  |
| deg3 | DG8 |  |  |  |
|  | DG9 |  |  |  |
| deg4 | DG10 |  |  |  |

Từ các đường đi với mỗi testcase, ta sẽ thiết kế các input của từng test như sau:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Input** | **EO** | **Note** |
| WT1 |  | Invalid Input |  |
| WT2 |  |  |  |
| WT3 |  |  | WT2 |
| WT4 |  |  |  |
| WT5 |  |  | WT2 |
| WT6 |  |  | WT4 |
| WT7 |  |  |  |
| WT8 |  |  | WT4 |
| WT9 |  |  | WT7 |
| WT10 |  |  |  |
| WT11 |  |  | WT7 |
| WT12 |  |  | WT10 |
| TP1 |  |  | WT2 |
| DG1 |  |  | WT2 |
| DG2 |  |  | WT4 |
| DG3 |  |  | WT7 |
| DG4 |  |  | WT10 |
| DG5 |  |  | WT4 |
| DG6 |  |  | WT7 |
| DG7 |  |  | WT10 |
| DG8 |  |  | WT7 |
| DG9 |  |  | WT10 |
| DG10 |  |  | WT10 |

# **Kết quả kiểm thử:**

* Sau đây sẽ là kết quả của tất cả các ca kiểm thử:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Input** | **Expected Output** | **Actual Output** | **Result** |
| WT1 |  | Invalid Input | Invalid Input | Passed |
| WT2 |  |  |  | Passed |
| WT4 |  |  |  | Passed |
| WT7 |  |  |  | Passed |
| WT10 |  |  |  | Passed |

* Kết luận: Chương trình đã vượt qua hết tất cả các test case với độ phủ all-uses .

# **Link github: Click** [**here**](https://github.com/hoang1803/Software-Testing/tree/main/Ki%E1%BB%83m%20th%E1%BB%AD%20d%C3%B2ng%20d%E1%BB%AF%20li%E1%BB%87u)