**Mục tiêu**

Luyện tập sử dụng cấu trúc lặp while.

**Mô tả**

Trong phần này, chúng ta sẽ phát triển một ứng dụng cho phép kiểm tra một số xem có phải là số nguyên tố hay không.

Số nguyên tố là một số nguyên dương chỉ chia hết cho 1 và chính nó.

Ứng dụng cho phép người dùng nhập vào một số và hiển thị kết quả là số đó là số nguyên tố hoặc không phải là số nguyên tố.

**Hướng dẫn**

Đầu tiên chúng ta xác định yêu cầu bài toán:

* Ứng dụng cho phép nhập vào 1 số và kiểm tra số đó có phải là số nguyên tố không

Phân tích:

* Số nguyên tố là số tự nhiên chỉ có hai ước số dương phân biệt là 1 và chính nó. Các số có nhiều hơn 2 ước số dương được gọi là hợp số.
* Do số 1 chỉ có một (1) ước số dương là chính nó, nên số  1 không phải là số nguyên tố và cũng không phải là hợp số.
* Số 2 là số nguyên tố nhỏ nhất, và cũng là số nguyên tố chẵn duy nhất.
* Tất cả các số chẵn không phải là số nguyên tố do nó có 3 ước số dương là 1, 2 và chính nó => Số nguyên tố là số lẻ.

Cách giải:

* Viết code kiểm tra số nhập vào có nhỏ hơn 2 không, nếu nhỏ hơn 2 => không phải là số nguyên tố.
* Viết code kiểm tra số nhập vào có phải là 2 hay không, nếu bằng 2 => là số nguyên tố.
* Dùng cấu trúc lặp while lặp qua tất cả các số lẻ nhỏ hơn căn bậc 2 của số vừa nhập vào. Bắt đầu từ số 3, mỗi lần lặp tăng giá trị số lẻ lên 2 (+2) đơn vị. Nếu:
  + Số nhập vào chia hết cho số lẻ thứ i => không phải là số nguyên tố
  + Số nhập vào không chia hết cho bất kì số lẻ nào => là số nguyên tố
* Viết code đọc dữ liệu từ bàn phím.

Chi tiết

* Viết hàm kiểm tra số nguyên tố. Hàm này có 1 tham số là số tự nhiên n
* Kiểm tra xem n có nhỏ hơn 2 không. Nếu có => không phải là số nguyên tố, return hàm.
* Kiểm tra xem n có = 2 không. Nếu có => n là số nguyên tố, return hàm.
* Tiếp tục tìm căn bậc 2 của n
* Dùng vòng lặp while bắt đầu từ số i = 3, mỗi lần lặp tăng giá trị lên 2
  + Kiểm tra nếu n chia hết cho i => n không là số nguyên tố, return hàm
  + N không chia hết cho i => n là số nguyên tố