**Bài 6:**

**Bước 1: Đặt câu hỏi cho AI**

**Nhập câu hỏi cho ChatGPT như sau:**

“Viết một đoạn code kiểm tra số nguyên tố bằng ngôn ngữ C.”

**Bước 2: Code AI tạo ra**

**Dưới đây là đoạn code ChatGPT sinh ra:**

#include <stdio.h>

int main() {

int n, i, isPrime = 1;

printf("Nhap mot so nguyen: ");

scanf("%d", &n);

if (n <= 1)

isPrime = 0;

else {

for (i = 2; i \* i <= n; i++) {

if (n % i == 0) {

isPrime = 0;

break;

}

}

}

if (isPrime)

printf("%d la so nguyen to.\n", n);

else

printf("%d khong phai la so nguyen to.\n", n);

return 0;

}

**Chạy thử và kiểm chứng**

Tôi sao chép đoạn code vào Code::Blocks, sau đó biên dịch và chạy thử với các giá trị khác nhau:

* **Trường hợp 1:** nhập n = 7  
  ➜ Kết quả: 7 la so nguyen to.  
  (Đúng vì 7 chỉ chia hết cho 1 và chính nó.)
* **Trường hợp 2:** nhập n = 12  
  ➜ Kết quả: 12 khong phai la so nguyen to.  
  (Đúng vì 12 chia hết cho 2, 3, 4, 6.)
* **Trường hợp 3:** nhập n = 1  
  ➜ Kết quả: 1 khong phai la so nguyen to.  
  (Đúng vì 1 không được tính là số nguyên tố.)

Tôi cũng kiểm tra logic vòng lặp for (i = 2; i \* i <= n; i++) và nhận thấy đây là cách tối ưu, vì chỉ cần kiểm tra đến căn bậc hai của n.

Chương trình chạy hoàn toàn chính xác, không có lỗi cú pháp hay sai kết quả.

**Bước 4: Kết luận**

**Bài học rút ra:**  
AI có thể cung cấp đoạn code chính xác và hiệu quả, giúp tôi tiết kiệm thời gian khi học lập trình. Tuy nhiên, việc hiểu rõ từng bước hoạt động của code là điều quan trọng để nắm chắc kiến thức, chứ không nên chỉ sao chép.

**Cần làm để không bị lệ thuộc hoặc hiểu sai khi dùng AI:**

* Đọc kỹ và phân tích từng dòng lệnh để hiểu ý nghĩa, thay vì chỉ chạy thử.
* Kiểm tra lại thuật toán bằng cách tự viết lại theo cách hiểu của mình.
* Đối chiếu với tài liệu hoặc giáo trình, đảm bảo AI không sinh ra lỗi logic.
* Dùng AI như một “trợ giảng ảo” — để gợi ý, giải thích, và học hỏi thêm, không thay thế cho việc tự tư duy và luyện tập lập trình.

Nhờ cách làm này, chúng ta vừa kiểm chứng được độ chính xác của AI, vừa hiểu sâu hơn về thuật toán kiểm tra số nguyên tố trong C — một bài tập cơ bản nhưng rất quan trọng khi học lập trình.