**3.1 QUESTIONS**

**3.1.1** Ưu điểm của Makefile là gì? Cho ví dụ?

Ưu điểm của Makefile gồm:

* Nó làm cho mã ngắn gọn và rõ ràng hơn để đọc và gỡ lỗi.
* Không cần phải biên dịch lại toàn bộ chương trình bất cứ khi nào cần thực hiện thay đổi đối với một chức năng hoặc một lớp. Makefile sẽ tự động biên dịch chỉ những tệp đã xảy ra thay đổi.
* Trong các mã hoặc dự án dài, Makefile được sử dụng rộng rãi để trình bày dự án một cách có hệ thống và hiệu quả hơn.

Ví dụ: Như ở các mục problem1, problem2, problem3 tại phần 3.3, ta hoàn toàn có thể code chương trình đầy đủ với các hàm như sau:

Ảnh có chứa văn bản

Mô tả được tạo tự động

Nhưng việc viết tất cả các hàm chung một file sẽ làm chúng ta khó phát hiện lỗi, khó gỡ lỗi hơn dài dòng và không được rõ ràng. Ngoài ra, mỗi lần thay đổi hay sửa đổi một hàm nào đó, ta đều phải biên dịch lại toàn bộ các hàm có trong file.

Thay vào đó, việc tách thành các file factrial.h, factorial.c, readline.h, readline.c, main.c và sử dụng makefile để liên kết giúp cho từng hàm được trình bày ngắn gọn, rõ ràng, và mang những ưu điểm của makefile như trên.

**3.1.2** Trong trường hợp các tệp mã nguồn nằm ở những nơi khác nhau, làm thế nào chúng ta có thể viết một Makefile?

Đối với các dự án nhỏ, ta có thể trực tiếp liệt kê địa chỉ các tệp nguồn trong makefile.

Đối với các dự án lớn, Cách truyền thống là có một makefile trong mỗi thư mục con (part1, part2, v.v.) cho phép bạn xây dựng chúng một cách độc lập. Hơn nữa, có một Makefile trong thư mục gốc của dự án để xây dựng mọi thứ.

**3.1.3** Kết quả đầu ra tại LINE A là bao nhiêu? Giải thích câu trả lơi của bạn.

Ảnh có chứa văn bản

Mô tả được tạo tự động

Kết quả đầu ra tại LINE A là: PARENT: value = 15

Giải thích như sau:

* Trước khi chạy đến fork() thì chương trình chạy tương đương với một tiến trình, và chỉ có một tiến trình ở thời điểm hiện tại. Sau đó thì chương trình sẽ tách thành 2 tiến trình giống nhau hoàn toàn để chạy câu lện tiếp theo. Trừ một diểm là process ID của 2 tiến trình khác nhau.
* Trong đó, tiến trình ban đầu là tiến trình cha, tiến trình được tách ra là tiến trình con. **Hai tiến trình này có vùng nhớ riêng,** **có không gian địa chỉ riêng.** Điều này cho phép nhiều tác vụ chạy độc lập với nhau như thể chúng có toàn bộ bộ nhớ của máy cho riêng mình.cho nên việc thay dổi giá trị (biến, vùng nhớ, …) của tiến trình con không ảnh hưởng đến tiến trình cha và ngược lại.
* Vậy ta thấy việc value ở tiến trình con được thay đổi thành 20. Nhưng value ở tiến trình cha vẫn là 5.