

§8. SOẠN THẢO, DỊCH, THỰC HIỆN VÀ HIỆU CHỈNH CHƯƠNG TRÌNH

A. Mục đích, yêu cầu

- Biết các bước: soạn thảo, dịch, thực hiện và hiệu chỉnh chương trình;
- Biết một số công cụ của môi trường Turbo Pascal;
- Bước đầu sử dụng được chương trình dịch để phát hiện lỗi;
- Bước đầu chỉnh sửa được chương trình dựa vào thông báo lỗi của chương trình dịch và tính hợp lí của kết quả thu được.

B. Những điểm cần lưu ý và gợi ý tổ chức dạy học

- a) Giáo viên trình bày ở mức giới thiệu sơ lược để học sinh có khái niệm ban đầu về soạn thảo, dịch, thực hiện và hiệu chỉnh chương trình. Chú ý nêu các phím chức năng để thực hiện dịch (**F9**), chạy chương trình (**Ctrl+F9**). Nếu có điều kiện có thể bố trí một máy tính và máy chiếu để minh họa hoặc giấy khổ lớn minh họa cửa sổ màn hình TP.
- b) Chương trình dịch TP thuộc loại biên dịch. Lỗi cú pháp được phát hiện ngay trong quá trình dịch. Khi gặp lỗi, TP sẽ không dịch tiếp mà dừng lại cho biết lỗi và vị trí của lỗi. Vì vậy, trong quá trình soạn thảo chương trình có thể dùng chương trình dịch (kết hợp sử dụng các phím **F7**, **F8**, **F4** và **F9** một cách thích hợp) để sửa đổi các lỗi của chương trình. Giáo viên nên giới thiệu phương pháp thường dùng khi viết một chương trình là chia bài toán thành nhiều nhiệm vụ độc lập và viết bổ sung dần từng đoạn chương trình tương ứng với các nhiệm vụ ấy. Mỗi lần viết xong một đoạn chương trình tương ứng với một nhiệm vụ người ta lại dùng chương trình dịch để sửa lỗi ngay đoạn vừa viết.
- c) Để thực hiện chương trình dùng tổ hợp phím **Ctrl+F9** (thuộc bảng chọn **Run**). Nếu trước đó ta vừa dịch và không có lỗi thì chương trình được thực hiện ngay. Trong trường hợp ngược lại, chương trình sẽ được dịch và được thực hiện nếu không có lỗi.
- d) Trong quá trình thực hiện chương trình nếu gặp thủ tục *readln* không có tham số thì người dùng phải nhấn phím **Enter** hoặc gặp *readln* có kèm theo danh sách biến thì nhập giá trị cho biến rồi nhấn phím **Enter**. Để tạo giao diện thân thiện với người dùng, chẳng hạn như thông báo cho người dùng biết lúc nào cần nhập dữ liệu, giáo viên nên hướng dẫn học sinh sử dụng kết hợp các thủ tục *read/readln* với các thủ tục *write/writeln*.

- e) Khi thực hiện chương trình, TP có thể thông báo lỗi, chẳng hạn như lỗi chia cho số 0, lỗi vượt phạm vi khai báo, lỗi tràn stack, lỗi tràn vùng nhớ heap,... Giáo viên khuyến khích học sinh xem phụ lục các lỗi khi thực hiện chương trình ở cuối sách giáo khoa.
- f) Có trường hợp không còn lỗi khi thực hiện chương trình nhưng kết quả thu được vẫn sai. Trong trường hợp này, nguyên nhân thường do đọc dữ liệu sai hoặc viết chương trình chưa thể hiện đúng thuật toán hoặc thuật toán trong chương trình không bảo đảm tính đúng đắn. Kiểm tra tính đúng đắn của thuật toán trong chương trình là một vấn đề phức tạp. Thông thường, người ta dùng một số bộ dữ liệu đặc biệt để kiểm tra (gọi là các bộ test). Các bộ test thường có các dạng như sau:
- Kích thước nhỏ, dễ dàng đoán nhận kết quả (không cần dùng máy tính);
 - Kích thước nhỏ, có thể dễ dàng đoán nhận kết quả nhưng có những tính chất đặc biệt (dễ dẫn đến kết quả sai);
 - Kích thước lớn nhưng có quy luật tìm ra output bằng suy luận không nhờ máy tính;
 - Kích thước lớn và giá trị ngẫu nhiên để kiểm tra các lỗi về tổ chức dữ liệu và thời gian thực hiện cho phép.
- g) Trong quá trình dạy học, giáo viên nên thường xuyên hướng dẫn học sinh tạo những bộ test phù hợp với các bài toán cụ thể.
- h) Ngoài cách kiểm tra bằng các bộ test, có thể dùng thủ tục *write*, *writeln* để hiển thị giá trị một vài biến đang cần quan tâm tại một số thời điểm thực hiện trên màn hình hoặc sử dụng **Ctrl+F7** để theo dõi giá trị các biến trong cửa sổ *Watches* (chạy chương trình từng bước). Căn cứ vào giá trị của các biến đó có thể điều chỉnh chương trình để có được kết quả như mong muốn. Sau đó, khi chương trình đã hoàn thiện thì xoá những lệnh hiển thị này.
- i) Giáo viên có thể minh hoạ những vấn đề trên trong giờ thực hành.