

Bài 14: Thuật toán sắp xếp (Sorting algorithms)

Bubber sort	<p>*Ý tưởng thuật toán: So sánh các cặp số liền kề nhau và trao đổi thứ tự nếu chúng không tuân theo thứ tự mà yêu cầu đề ra. Có nghĩa là nếu như người dùng muốn tăng dần \Rightarrow 2 số mà số trước lớn hơn số sau thì sẽ đổi chỗ. Tương tự với trường hợp còn lại. Và cứ mỗi vòng lặp kết thúc sẽ đưa số lớn nhất về cuối dãy. \Rightarrow Đặt tên là sắp xếp nổi bọt.</p> <p>*Độ phức tạp:</p> <ul style="list-style-type: none"> - Tốt nhất: $O(n)$ - Tệ nhất: $O(n^2)$ - Trung bình: $O(n^2)$
Selection sort	<p>*Ý tưởng thuật toán: Tìm ra số lớn nhất/nhỏ nhất và đưa về cuối mảng. Rồi loại bỏ phần tử đó ra khỏi vòng lặp tiếp theo \Rightarrow Tìm số nhỏ nhất/lớn nhất trong $n-1$ số còn lại \Rightarrow Cứ tiếp tục như thế cho đến khi hết mảng.</p> <p>*Độ phức tạp: Tệ nhất/Tốt nhất/Trung bình: $O(n^2)$</p>
Insertion sort	<p>Tạo ra danh sách con luôn luôn duy trì dưới dạng đã sắp xếp. Sắp xếp chèn là chèn một phần tử vào danh sách đã qua sắp xếp. Phần tử được chèn vào vị trí thích hợp sao cho vẫn đảm bảo rằng danh sách con đó vẫn sắp xếp theo thứ tự.</p>
*Lựa chọn cách sort	<ul style="list-style-type: none"> - Mảng đã được sắp xếp: nên dùng Bubber sort, Insertion sort (đã nâng cấp) - Mảng đi ngược lại với mong muốn của chúng ta (Đề bài yêu cầu mảng tăng dần nhưng đưa vào là mảng giảm dần): Selection sort - MergeSort, HeapSort, QuickSort,...

Bài 15: Xử lý ngoại lệ và Debug

Bug là gì?	Bug là lỗi phần mềm hay hỏng hóc trong chương trình hoặc hệ thống khiến nó tạo ra kết quả không mong muốn, không chính xác hoặc hành xử theo cách không lường trước được.
Exception	<p>Là một sự kiện bất thường xảy ra trong quá trình thực thi \Rightarrow phá vỡ luồng thực thi của chương trình và đôi khi làm chết hệ thống.</p> <p>* <u>CheckedException</u>: là một Exception xảy ra trong lúc compile, không thể được bỏ qua và bắt buộc chúng ta phải handle (xử lý).</p> <ul style="list-style-type: none"> - FileNotFoundException, DatabaseException, IOException,... <p>* <u>UncheckedException</u>: Là loại Exception xảy ra trong quá trình runtime, lỗi không chắc chắn có thể xảy ra. Loại Exception này có thể bỏ qua trong quá trình compile và không bắt buộc phải handle.</p> <ul style="list-style-type: none"> - NullPointerException (NPE): Ví dụ: Student student = null; student.name. - ArrayIndexOutOfBoundsException: Truy cập giá trị index ngoài vùng nhớ lưu trữ của mảng. Ví dụ: int[] array = new int[4].Array[6] <ul style="list-style-type: none"> - NumberFormatException, ... <p>Ngoài 2 exception trên thì còn 1 cái là anh em của exception là error. Error là một lỗi nghiêm trọng liên quan đến môi trường thực thi của ứng dụng, hệ thống mà lập trình viên không thể kiểm soát \Rightarrow Lỗi gây chết chương trình và nó không thể được handle.</p> <ul style="list-style-type: none"> - OutOfMemoryError, VirtualMachineError, StackOverflowError, ...
Xử lý ngoại lệ	<p>Dùng throws, throw hoặc try-catch</p> <ul style="list-style-type: none"> - Dùng try-catch khi mình muốn xử lý lỗi ngay đoạn bị lỗi - Sử dụng throw khi chủ động ném ra một ngoại lệ <p>*<u>Lưu ý khi sử dụng try-catch</u>:</p> <ul style="list-style-type: none"> - Tại 1 thời điểm thì chỉ có 1 khối catch được thực thi.

	<ul style="list-style-type: none">- Trong một try-catch thì có thể có nhiều khối catch. Khối catch phải được sắp xếp từ cụ thể nhất cho đến chung nhất (từ exception con đến exception cha)- Try bắt buộc phải có nhưng khối catch hoặc khối finally 1 trong 2 có là được.