**CÂU HỎI AUDIT MODULE 2**

**ADVANCED PROGRAMING WITH JAVA 2.0**

|  |  |
| --- | --- |
| STT | Câu hỏi |
| 1 | JAVA là gì ?  (là ngôn ngữ lập trình) |
| 2 | JDK, JRE, JVM  (java develop kit, java runtime environment, java virtual machine) |
| 3 | Thông dịch, biên dịch khác nhau như thế nào ?  (biên dịch xác định chương trình nguồn có dịch được hay không. Thông dịch – dịch từng câu lệnh lỗi ở đâu báo ở đó) |
| 4 | JAVA là ngôn ngữ thông dịch hay biên dịch ?  (vừa thông dịch vừa biên dịch) |
| 5 | Các kiểu dữ liệu trong java, và giá trị mặc định khi khai báo. |
| 6 | Có bao nhiêu lại mệnh đề if |
| 7 | So sánh if và switch-case |
| 8 | Khi nào dùng for, while, do-while ? |
| 9 | Phân biệt for-i và for-each |
| 10 | Khác nhau giữa while, do-while. Cho ví dụ khi nào dùng ? |
| 11 | Break, Continue có tác dụng gì trong mệnh đề lặp ? |
| 12 | Trình bày các cách khởi tạo một mảng trong JAVA ? |
| 13 | Phần tử của mảng có thể dùng kiểu dữ liệu nào, và có giá trị mặc định là gì ? |
| 14 | OOP là gì ? |
| 15 | Phân biệt class và object |
| 16 | Constructor là gì  (là hàm khởi tạo) |
| 17 | Cách khai báo constructor và đặc điểm constructor trong JAVA  (public NameClass(){}) |
| 18 | Phân biệt constructor và method  (constructor là hàm đặc biệt, tên phải trùng với tên lớp, không có kiểu trả về, được sử dụng thông qua từ khoá new) |
| 19 | Tính bao đóng là gì ? Làm sao để thu được tính bao đóng trong java ?  (tính bao đóng là các dữ liệu được che dấu không cho thấy được từ bên ngoài.  Cách thu được tính bao đóng là khai báo private cho dữ liệu đó, và sử dụng nó thông qua get set) |
| 20 | Tham trị, tham chiếu  (tham trị là vùng nhớ copy giá trị và không làm thay đổi giá trị gốc, tham chiếu là cả 2 giá trị đều bị vùng nhớ tương tác lẫn nhau – nếu giá trị b gọi giá trị a và thay đổi giá trị thì a sẽ bị thay đổi theo b ) |
| 21 | Từ khóa static dùng để làm gì ?  (dùng để quản lý (bộ nhớ) dễ dàng hơn được truy cập trực tiếp thông qua lớp mà không cần khởi tạo) |
| 22 | Ràng buộc khi sử dụng static |
| 23 | Các loại biến trong JAVA  (biến cục bộ, và biến toàn cục, cục bộ thì nằm trong khối lệnh – ra ngoài thì mất, biến toàn cục thì nằm ngoài khối lệnh chạy vào khối lệnh và ra cũng không mất) |
| 24 | Trình bày các loại access modifier, và phạm vi truy cập.  (private : trong class  Default : private + bên trong gói  Protected : default + bên trong gói con  Public : protected + bên ngoài gói ) |
| 25 | Kế thừa trong JAVA là gì ?  ( lớp con kế thừa lại tất cả các thuộc tính và phương thức của lớp cha ) |
| 26 | Lớp con kế thừa được những tài sản nào(thuộc tính, phương thức) của lớp cha ?  (kế thừa được tất cả) |
| 27 | Lớp Object là gì  ( là cha của tất cả các lớp ) |
| 28 | Khái niệm đa hình  ( đa hình sẽ gắn liền với kế thừa, như 1 phương thức nhưng có nhiều cách triển khai khác nhau ) |
| 29 | Phân biệt Overloading và Overriding  ( overload : nhiều phương thức có cùng 1 tên nhưng khác kiểu dữ liệu tham số truyền vào.  Override : định nghĩa lại phương thức của lớp cha ) |
| 30 | ép kiểu là gì ? các loại ép kiểu  (là biến kiểu dữ liệu giá trị này thành kiểu dữ liệu khác, có 2 loại ép kiểu, ép từ kiểu nguyên thuỷ sang đối tượng và ngược lại ) |
| 31 | Tính trừu tượng là gì ? |
| 32 | Phân biệt abstract class và abstract interface  (quan hệ giữa một class khi thừa kế một abstract class được gọi là is-a, và một class khi hiện thực một interface được gọi là can-do ) |
| 33 | So sánh Array và ArrayList |
| 34 | So sánh ArrayList và LinkedList |
| 35 | Set là gì, các lớp triển khai của Set |
| 36 | Generic là gì  (tham số hoá kiểu dữ liệu) |
| 37 | Ưu điểm và hạn chế khi dùng generic ?  ( ưu điểm : dễ quản lý … chả nhớ @@  Hạn chế : không thể gọi kiểu generic bằng kiểu dữ liệu nguyên thuỷ, không thể sử dụng static cho reneric, không thể khởi tạo đối tượng của kiểu dữ liệu generic) |
| 38 | Stack là gì, các phương thức của stack ? |
| 39 | Queue là gì, các class triển khai của queue ? |
| 40 | Phương thức cơ bản của queue ? |
| 41 | So sánh Comparable và Comparator, khi nào dùng cái nào ? |
| 42 | Map là gì, các class triển khai |
| 43 | Cây nhị phân là gì |
| 44 | Trình bày các cách duyệt cây nhị phân. |
| 45 | Ngoại lệ (Exception là gì) ? |
| 46 | Phân loại Exception |
| 47 | Phân biệt Error và Exception |
| 48 | Có bao nhiêu cách để xử lý ngoại lệ (handle) |
| 49 | Một số lưu ý khi dùng try-catch |