Tên: Nguyễn Thành Phát - 2000006273

Lớp: 20DTH2B

**A. Hãy vẽ lưu đồ biểu diễn thuật toán giải quyết các bài toán sau**

1/ Tính tổng các chữ số của một số nguyên *n*.

* Ví dụ 01: *n* = 35234

Kết quả: 17

Giải thích: 3 + 5 + 2 + 3 + 4 = 17

* Ví dụ 02: *n* = -94830

Kết quả: 24

Giải:

|  |  |
| --- | --- |
| Lưu đồ | Code mẫu |
| C:\Users\Thanh Phat\AppData\Local\Microsoft\Windows\INetCache\Content.Word\LuuDoBt1.png | class Main {  public static void main(String args[]) {  int n = 123;//nhap n  int du = 0;  int tong = 0;  while (n > 0) {  du = n % 10;  n = n / 10;  tong += du;  }  System.out.println(tong);  }  } |

2/ Tính số hạng thứ *n* của dãy

* Ví dụ 01: *n* = 5

Kết quả: 2.223

* Ví dụ 02: *n* = 31

Kết quả: 378.983.862.378.400.042.113.903

Giải:

|  |  |
| --- | --- |
| Lưu đồ | Code mẫu |
| C:\Users\Thanh Phat\AppData\Local\Microsoft\Windows\INetCache\Content.Word\LuuDoBt2.png | class Main {  public static void main(String args[]) {  float a\_0 = -1;  float a\_1 = 3;  int n = 5;// nhap n  float a\_n = 0;  if (n == 1) {  a\_n = 3;  }  else{  for (int i = 0; i < (n - 1); i++) {  a\_n = 5 \* a\_1 + 6 \* a\_0;  a\_0 = a\_1;  a\_1 = a\_n;  }  }  System.out.println(a\_n);  }  } |

**B. Hãy thể hiện thuật toán bằng mã giả hoặc ngôn ngữ tự nhiên giải quyết các bài toán sau đây**

1/ Liệt kê tất cả các ước số của một số nguyên dương *n*

* Ví dụ 01: *n* = 20

Kết quả: 1, 2, 4, 5, 10, 20

* Ví dụ 02: n = 19

Kết quả : 1, 19

Giải:

|  |  |
| --- | --- |
| Ngôn ngữ tự nhiên | Code |
| n = 24// (số nguyên)  Dò từ 1 đến n (với i là số đếm).  Nếu số dư của phép n chia i bằng 0.  In ra màn hình số ở vị trí i. | class Main {  public static void main(String args[]) {  int n = 24;  for(int i = 1;i <= n;i++){  if(n % i == 0){  System.out.println(i);  }  }  }  } |

2/ Tìm ước chung lớn nhất (UCLN) của hai số nguyên dương *a* và *b*

* Ví dụ 01: *a* = 100, *b* = 64

Kết quả: 4

* Ví dụ 02: *a* = 80, *b* = 48

Kết quả : 16

|  |  |
| --- | --- |
| Ngôn ngữ tự nhiên | Code |
| Hai số nguyên vào 2 biến.  a = 100;  b = 64;  Xét nếu a = 0 hoặc b = 0 hay không  Nếu có thì in ra a + b  Nếu không thì tạo ra vòng lặp chạy với điều kiện a phải khác b.  Trong quá trình lặp ta xét:  Nếu a lớn hơn b:  Thì ghi đè a sẽ lại bằng a – b  Ngược lại:  Ghi đè b sẽ bằng b – a;  Sau khi lặp xong sẽ xuất ra màn hình kết quả a.  Kết thúc. | class Main {  public static void main(String args[]) {  int a = 100;  int b = 64;  if (a == 0 || b == 0) {  System.out.println(a + b);  } else {  while (a != b) {  if (a > b) {  a -= b;  } else {  b -= a;  }  }  System.out.println(a);  }  }  } |

3/ Nhập vào một số nguyên dương *n* ở hệ cơ số 10. Hãy chuyển đổi số *n* về hệ nhị phân.

* Ví dụ 01: *n* =

Kết quả:

* Ví dụ 02: *n* = 4

Kết quả:

|  |  |
| --- | --- |
| Ngôn ngữ tự nhiên | Code |
| Tạo ra mảng tối đa 31 giá trị.  Lấy n chia 2 tuần tự rồi lấy chính kết quả đó chia tiếp để lấy các số dư nhập vào trong mảng.(với điều kiện chỉ lấy phần nguyên, không lấy phần thập phân của số)  Lặp quá trình chia đó cho đến khi n = 0.  Ta xuất ra giá trị nhị phân bằng cách đảo ngược mảng lại rồi đưa ra kết quả. | class Main {  public static void main(String args[]) {  int n = 16;  int nhiphan[] = new int[31];  int i = 0;  while (n > 0) {  nhiphan[i++] = n % 2;  n /= 2;  }  for (int j = i - 1; j >= 0; j--) {  System.out.print(nhiphan[j]);  }  }  } |

**II. Ngôn ngữ lập trình**

**Hãy trả lời các câu hỏi sau đây**

1/ Thế nào là tính khả chuyển của một chương trình (máy tính)?

Là khả năng chạy được trên nhiều hệ thống máy tính hay trên nhiều hệ điều hành khác nhau.

2/ Hãy cho biết đầu vào (input) và đầu ra (output) của trình biên dịch Java là gì ?

- Đầu vào: Toàn bộ trường trình.

- Đầu ra: Mã đối tượng trung gian.

3/ Sự khác nhau giữa trình biên dịch (compiler) và trình thông dịch (interpreter)

Việc biên dịch sẽ phải hoàn thành trước rồi mới thực thi.

Việc biên dịch và thực thi sẽ diễn ra đồng thời cùng lúc.

4/ Cơ chế nào giúp Java có hiệu năng cao ?

Cơ chế Biên dịch tốc độ nhanh hơn cơ chế Thông dịch.

5/ Java bytecode khác biệt với các ngôn ngữ cấp thấp khác như thế nào ?

Thay vì biên dịch mã nguồn thành mã máy hoặc thông

dịch mã nguồn khi chạy, Java được thiết kế để biên dịch mã

nguồn thành bytecode, bytecode sau đó sẽ được thực thi trong

một môi trường (runtime environment) độc lập, hoạt động trên

hệ điều hành gọi là máy ảo Java (Java Virtual Machine)

6/ Phân biệt các công nghệ sau của Java: J2SE, J2EE, J2ME

J2SE (Java 2 Standard Edition) là một nền tảng thực thi (bao gồm cả phát triển ứng dụng và triển khai) cho các ứng dụng Java. J2SE cung cấp các API, các kiến trúc chuẩn, các thư viện lớp và các công cụ cần thiết để xây các ứng dụng Java. J2SE vẫn được xem là nền tảng thiên về phát triển các sản phẩm chạy trên máy tính để bàn.

J2SE gồm 2 thành phần chính: JRE & JDK

• JRE – Java 2 Runtime Environment

• JDK – Java 2 Software Development Kit

J2EE (Java 2 Platform, Enterprise Edition, hay Java EE) là một nền tảng cho phép lập trình để phát triển ứng dụng phân tán trên kiến trúc đa tầng (nLayer), chủ yếu dựa vào các module chạy trên các máy chủ ứng dụng.

J2ME được phát triển từ kiến trúc Java Card, Embeded Java vf Personal Java của phiên bản Java 1.1. Đến sự ra đời của Java 2 thì Sun quyết định thay thế Personal Java và được gọi với tên mới là Java 2 Micro Edition, hay viết tắt là J2ME. Đúng với tên gọi, J2ME là nền tảng cho các thiết bị có tính chất nhỏ gọn.