**ĐẠI HỌC NÔNG LÂM TP. HỒ CHÍ MINH LẦN THI THỨ**

**KHOA CÔNG NGHỆ THÔNG TIN NGÀY THI *(GV không cần ghi)***

**ĐỀ THI MÔN CHUYÊN ĐỀ JAVA- ĐỀ THI SỐ 4**

**THỜI GIAN LÀM BÀI 60 PHÚT**

*(Không được sử dụng tài liệu)*

1. **Cho đoạn chương trình sau:**

String test = "1a2b3c4d5";

String[] tokens = test.split("c");

System.*out*.println(Arrays.*toString*(tokens));

**Kết quả sẽ là gì?**

* 1. [a, b, c, d]
  2. [, 1, a, 2, b, 3, c, 4, d, 5]
  3. [, a, b, c, d]
  4. [1a2b3, 4d5]

1. **Cho đoạn chương trình sau:**

Map<String, Integer> map = **new** HashMap<String, Integer>();

map.put("key12", 3);

map.put("key99", 1);

map.put("key8", 2);

map.put("key12", 1);

Collection<Integer> values = map.values();

System.*out*.println(values);

**Kết quả sẽ là gì?**

* 1. [1, 1, 2, 3]
  2. [1, 1, 2]
  3. [3, 1, 2]
  4. [3, 1, 2, 1]

1. **Cho đoạn chương trình sau:**
   * 1. **public** **class** MyClass {
     2. **private** **String** i;
     3. // insert code here
     4. **public** MyClass(**String** i) {
     5. **this**.i = i;
     6. }}
     7. **class** MyOtherClass **extends** MyClass {//*lỗi*
     8. }

**Bạn cần thêm dòng lệnh nào sau đây vào dòng 3 để dòng 7 hết lỗi?**

1. **public** MyClass() {MyClass(1); }
2. **public** MyClass() { MyOtherClass (); }
3. **public** MyClass() {**this**(1); }
4. **public** MyClass() { MyOtherClass (1); }
5. **public** MyClass() {**this**(“1”); }
6. **Khai báo hàm main nào dưới đây là không hợp lệ?**
   1. **public static final void** main(String args[]) {}
7. **public static void main(String args) {}**
   1. **public static void** main(String[] args) {}
   2. **public static void** main(String args[]) {}
8. **Cho đoạn chương trình sau:**

**public** **class** D {

**private** **int** a = 7;

**public** D(**int** a) { **this**.a = a;}

**public** **void** modify() {**this**.a--;}

**public** **static** **void** main(String[] args) {

**int** a = 10;

D d = **new** D(a);

d.modify();

System.*out*.println(“a = ” + a);

}}

**Kết quả nào sau đây là đúng?**

* 1. a = 7
  2. Một ngoại lệ được ném ra
  3. Lỗi biên dịch
  4. a = 10

1. **Để khai báo thuộc tính của một lớp có thể được truy xuất trong tất cả các lớp khácta dùng quyền truy xuất hay khai báo nào sau đây:**
   1. private
   2. protected
   3. public
   4. Không khai báo gì cả
2. **Cho đoạn chương trình sau:**

**boolean** bool = **true**;

**if** (bool = **false**)

System.*out*.print("c ");

**if** (bool)

System.*out*.print("b ");

**if** (!bool)

System.*out*.print("a ");

System.*out*.print("d ");

**Chương trình sẽ in ra lựa chọn nào sau đây?**

* 1. c d
  2. a d
  3. a c d
  4. a b d

1. **Trường hợp nào sau đây, trình biên dịch cung cấp contructor mặc nhiên cho lớp A**
2. **class** A {  
    **public** A(**int** x){}  
   }
3. **private class** A {

}

1. **class** Z { }  
   **class** A **extends** Z {  
    **void** A(){}  
   }
2. **class** A {  
    **public** A(){}  
   }
3. **Cho đoạn chương trình sau:**

**public** **class** MyClass {

**public** **enum** Alphabet{*C*, *E*, *D*}

**public** **static** **void** main(String[] args) **throws** Exception {

Set< Alphabet > set = **new** TreeSet< Alphabet >();

set.add(Alphabet.*E*);

set.add(Alphabet.*C*);

set.add(Alphabet.*D*);

System.*out*.println(set);

}}

**Kết quả sẽ là gì?**

1. [C, E, D]
2. [D, E, C]
3. [E, C, D]
4. [E, D, C]
5. **Cho trước đoạn chương trình sau:**

**public** **class** Test {

**private** **static** **int** *x* = *getValue*();

**private** **static** **final** **int** *y* = 5;

**private** **static** **int** getValue(){

**return** *y*; }

**public** **static** **void** main(String[] args) {

System.*out*.println(*x*);

}}

**Kết quả sẽ là gì khi biên dịch và chạy chương trình?**

* 1. Xuất ra 0
  2. Lỗi khi chạy
  3. Lỗi cú pháp (Syntax error)
  4. Xuất ra 5

1. **Phát biểu nào đúng về các giá trị trả về của các phương thức hashCode() và equals() được trả về từ 2 thể hiện của cùng một lớp**
   1. Nếu các giá trị hashCode khác nhau, các đối tượng phải khác nhau.
   2. Nếu các giá trị hashCode giống nhau, các đối tượng phải bằng nhau
   3. Nếu các giá trị hashCode khác nhau, các đối tượng có thể bằng nhau.
   4. Nếu các giá trị hashCode giống nhau, các đối tượng phải bằng nhau.
2. **Cho đoạn chương trình sau:**

**public** **class** Base {

**public** **enum** MyEnum { *A*, *B*, *C*, *D*};

**public** **static** **void** main(String[] args) {

MyEnum myEnum = MyEnum.*B*;

**switch** (myEnum) {

**case** *C*:

System.*out*.print("C ");

**case** *D*:

System.*out*.print("D ");

**default**:

System.*out*.print("B ");

**case** *A*:

System.*out*.print("A ");

}}}

**Kết quả sẽ là gì?**

* 1. B A
  2. C D
  3. C
  4. B

1. **Cho đoạn chương trình sau:**
   * 1. **interface** A{}
     2. **class** B **implements** A{}
     3. **class** C **extends** B{}
     4. **public** **class** D **extends** C{
     5. **public** **static** **void** main(String[] args) {
     6. C x = **new** C();
     7. //insert code here
     8. }}

**Lựa chọn nào sau đây là hợp lệ khi muốn thêm vào dòng 7**

1. A a = (B)x;
2. A a = (D) x;
3. A a = x;
4. C a = (C)(B)x;
5. **Cho đoạn chương trình sau:**

**public** **static** List reverse(List list) {

Collections.*reverse*(list);

**return** list;}

**public** **static** **void** main(String[] args) {

List list = **new** ArrayList();

list.add("1");

list.add("2");

list.add("3");

**for** (Object oblect : *reverse*(list)) {

System.*out*.print(oblect + ", ");

} }

**Kết quả sẽ là gì?**

* 1. 3, 2, 1,
  2. 3, 2, 1
  3. 1, 2, 3,
  4. 1, 2, 3

1. **Trường hợp nào sau đây gây lỗi biên dịch**
2. **Integer** results[] = {**new** Integer(3), 6, **new** Integer(10)};
3. **String**[] dogs = **new** String[] {**new** String("Fido"), **new** String("Spike"), "Aiko"};
4. **int**[] scores = {3, 5, 10};
5. **String**[] cats = {"Fluffy", "Spot", "Zeus"};
6. **int**[][] scores1 = {2, 4, 5}, {9, 3, 2};
7. **Cho phương thức sau:**

LinkedList list = new LinkedList();

list.add("Dai hoc");

list.add("Nong Lam");

list.add("TP.HCM");

**Để in được từ “Nong Lam” ta sử dụng phương thức:**

1. System.*out*.println(list.get(“Nong lam”));
2. System.*out*.println(list[1]);
3. System.*out*.println(list[2]);
4. System.*out*.println(list.get(1));
5. **Cho đoạn chương trình sau:**

**class** A {

**public** **byte** getNumber() {

**return** 1;

}

}

**public** **class** B **extends** A {

**public** **int** getNumber() {

**return** 2;

}

**public** **static** **void** main(String args[]) {

B b = **new** B();

System.*out*.println(b.getNumber());

}

}

**Kết quả nào sau đây là đúng?**

* 1. Lỗi biên dịch
  2. Một ngoại lệ được ném ra
  3. 2
  4. 1

1. **Cho đoạn chương trình sau:**

**import** java.util.\*;

**public** **class** B **implements** Comparable<B> {

**private** String maSV;

**public** B(String mSSV) { maSV = mSSV; }

**public** **int** compareTo(B o) { **return** 0; }

**public** **static** **void** main(String[] args) {

B sv1 = **new** B("123");

B sv2 = **new** B("123");

Set<B> s = **new** TreeSet<B>();

s.add(sv1);

s.add(sv2);

System.*out*.println(s.size());

}}

**Kết quả sẽ là gì?**

* 1. Lỗi biên dịch
  2. Một ngoại lệ được ném ra
  3. 1
  4. 2

1. **Cho đoạn chương trình sau:**

Object[] array = { **new** Double(1.0), **new** Double(5.0), **new** Double(7.0) };

System.*out*.println(Arrays.*binarySearch*(array, **new** Double(5)));

**Kết quả sẽ là gì?**

* 1. In ra 2
  2. Ném lỗi ClassCastException
  3. In ra 0
  4. In ra 1

1. **Cho đoạn chương trình sau:**

**interface** A{

**public** **void** method1();

**public** **void** method2();

}

**abstract** **class** B **implements** A{

**public** **abstract** **void** method3();

**public** **void** method2(){};

}

**class** C **extends** B{

//code here

}

**Class C cần phải override tối thiểu bao nhiêu phương thức để đoạn code trên không có lỗi?**

* 1. 3
  2. 2
  3. 1
  4. 0

1. **Cho đoạn lệnh sau:**

**class** Alpha {**public** Alpha () {System.*out*.print("Alpha ");}}

**public** **class** Base **extends** Alpha {

**public** Base () { System.*out*.print("Base ");}

**public** **static** **void** main(String[] args) {

**new** Base();

}

}

**Kết quả sẽ là gì?**

* 1. Alpha
  2. Alpha Base
  3. Base
  4. Base Alpha

1. **Cho đoạn chương trình sau:**

**for** (**int** i = 10; i < 10; i++)

method();

**Theo đoạn chương trình trên sẽ gọi phương thức method() bao nhiêu lần?**

* 1. 2147483647 lần (Integer.MAX\_VALUE)
  2. 1 lần
  3. 0 lần
  4. Chưa biết (vô tận)

1. **Cho trước đoạn chương trình sau:**

**class** Hexy {

**public** **static** **void** main(String[] args) {

Integer i = 42;

String s = (i > 40) ? "life" : (i > 50) ? "universe" : "everything";

System.out.println(s);

}}

**Kết quả sẽ là gì?**

1. universe
2. everything
3. life
4. Lỗi biên dịch.
5. **Khai báo nào dưới đây là đúng?**
   1. **private abstract** **class** Canine {**public** **void** speak() {}}
   2. **abstract** **class** Canine {**public** **void** speak() {}}
   3. **class** Canine {**public** **abstract** **void** speak();}
   4. **protected abstract** **class** Canine {**public** **void** speak() {}}
   5. **class** Canine {**public** **void** speak();}
6. **Cho đoạn lệnh sau:**

Integer[] array = **new** Integer[]{1,2,3,4,5};

Lựa chọn nào sau đây giúp bạn tạo ra một danh sách (List) từ mảng này

* 1. List<Integer> list = Collections.asList(array);
  2. List<Integer> list = **new** ArrayList<Integer>(array);
  3. List<Integer> list = Arrays.*asList*(array);
  4. List<Integer> list = List.asList();

1. **Cho đoạn chương trình sau:**

**for** (int i = 9; i > 4; i -=3 )

System.out.print(i+" ");

**Kết quả sẽ là gì?**

* 1. 6 9
  2. Lỗi
  3. 9 8 7 6 5
  4. 9 6

1. **Cho đoạn chương trình sau:**

**void** waitForSignal() **throws** Exception{

Object obj = **new** Object();

**synchronized** (Thread.*currentThread*()) {

obj.wait();

obj.notify();

}}

**Chọn đáp án đúng trong các lựa chọn sau đây?**

1. Ném lỗi IllegalStateception
2. Ném lỗi NullPointerException
3. Không cần ném ra lỗi gì hết
4. Ném lỗi InterruptedException
5. **Cho đoạn chương trình sau:**
   * 1. **public** **class** Test **implements** Runnable {
     2. **public** **void** run() {
     3. System.*out*.print(" running ");}
     4. **public** **static** **void** main(String[] args) **throws** Exception {
     5. Thread t = **new** Thread(**new** Test());
     6. System.*out*.print("begin");
     7. t.start();
     8. t.join();
     9. System.*out*.print("end");
     10. }}

**Kết quả nào sau đây là đúng khi chạy đoạn chương trình này?**

1. running begin end
2. begin running end
3. begin end running
4. Chưa xác định được
5. **Cho đoạn chương trình sau:**

**public** **class** Test {

**public** **void** add3(Integer i) {

**int** val = i.intValue();

val += 3;

i = **new** Integer(val);

}

**public** **static** **void** main(String args[]) {

**int** i = 0;

*add3*(i);

System.*out*.println(i);

}

}

**Kết quả nào sau đây là đúng?**

* 1. Một ngoại lệ được ném ra
  2. Lỗi biên dịch
  3. 3
  4. 0

1. **Điều gì xảy ra nếu dịch và chạy đoạn chương trình sau:**

1. **public** **class** Q1 **extends** Thread {

2. **public** **void** run() {

3. System.*out*.println("Before start method");

4. **this**.stop();

5. System.*out*.println("After stop method");

6. }

7. **public** **static** **void** main(String[] args) {

8. Q1 a = **new** Q1();

9. a.start();

10. }

11.}

* 1. Lỗi biên dịch xảy ra tại dòng 7.
  2. Xuất ra màn hình "Before start method" và "After stop method".
  3. Chỉ xuất ra màn hình "Before start method" .
  4. Một ngoại lệ ném ra ờ thời gian thực hiện chương trình tại dòng 7.

1. **Cho trước đoạn chương trình sau:**

**int** x = 3;

**int** y = 1;

**if** (x == y) System.out.println("x = " + x);

**Kết quả sẽ là gì?**

* 1. Chương trình chạy thành công nhưng không xuất gì ra màn hình
  2. x = 1
  3. x = 3
  4. Lỗi biên dịch
  5. Một ngoại lệ ném ra ờ thời gian thực hiện chương trình.

1. **Cho đoạn chương trình sau:**

**public** **class** Mixer {

Mixer m1 = **new** Mixer();

Mixer() { }

Mixer(Mixer m) {**this**.m1 = m;}

**void** go() {System.*out*.print("hi "); }

**public** **static** **void** main(String[] args) {

Mixer m2 = **new** Mixer();

Mixer m3 = **new** Mixer(m2);

m3.go();

Mixer m4 = m3.m1;

m4.go();

}}

**Kết quả sẽ là gì?**

* 1. Lỗi
  2. hi, sau đó báo lỗi
  3. hi
  4. hi hi

1. **Cho đoạn chương trình sau đây:**

**int** i = 0, j = 5;

**do** {

**if** (i++ < --j) **continue**;

} **while** (j > 5);

System.*out*.println("i = " + i + " and j = " + j);

**Kết quả sẽ là gì?**

* 1. i = 0 and j = 4
  2. i = 1 and j = 4
  3. i = 4 and j = 0
  4. i = 5 and j = 0

1. **Cho đoạn chương trình sau:**
   * 1. **int** number;
     2. **try** {
     3. number = Integer.*parseInt*("123");
     4. } **catch** (NumberFormatException e) {
     5. System.*out*.println(e);
     6. } **finally** {
     7. System.*out*.println("number = " + number);
     8. }

**Chọn câu trả lời đúng?**

1. Kết quả in ra là 123
2. Lỗi ở dòng thứ 7
3. Lỗi ở dòng thứ 3
4. Lỗi ở dòng thứ 1
5. **Cho trước đoạn chương trình sau:**

String s1 = **new** String("hello");

String s2 = **new** String("hello");

Double d = 9.0;

Float f = 9.0F;

System.*out*.print(f.equals(d) + “ ”);

System.*out*.print(s1=s2+ “ ”);

System.*out*.print(s1.equals(s2) + “ ”);

**Kết quả sẽ là gì khi biên dịch và chạy chương trình?**

1. Xuất ra "true hello true"
2. Xuất ra "false hello false"
3. Xuất ra "false false true"
4. Lỗi cú pháp (Syntax error)
5. **Cho đoạn chương trình sau:**

**float** pi = **new** Float(3.14);

**if** (pi < 3)

System.*out*.print("pi < 3 ");

**else**

System.*out*.print("pi = 3 ");

**if** (pi > 3)

System.*out*.print("pi > 3 ");

**Kết quả sẽ là gi?**

* 1. pi > 3
  2. pi = 3
  3. pi = 3 pi > 3
  4. pi = 3 pi < 3

1. **Cho đoạn lệnh sau:**

**class** A {

**static** **int** *y* = 2;

**public** A(**int** x) {

**this**();

*y* \*= 3;

}

**public** A() {*y*+=5; }

}

**public** **class** B **extends** A {

**public** B() {

**super**(*y*);

*y* += 2;

}

**public** **static** **void** main(String[] args) {

**new** B();

System.*out*.println(*y*);

}

}

**Kết quả sẽ là gì?**

* 1. 22
  2. 11
  3. 6
  4. 23

1. **Để gán tất cả các phần tử trong mảng ký tự có giá trị là 1 ta dùng:**
2. **char**[] ch = **new** **char**[10];

Arrays.*fill*(ch, "one");

1. **char**[] ch = **new** **char**[10];

Arrays.*fill*(ch, '1');

1. **char**[] ch = **new** **char**[10];

Arrays.*fill*(ch, 1);

1. **char**[] ch = **new** **char**[10];

Arrays.*fill*(ch, "1");

1. **Cho trước đoạn chương trình sau:**

1. **public** **class** ArrayTest {

2. **public** **static** **void** main(String[] args) {

3. **float** fl[], f2[];

4. fl = **new** **float**[10];

5. f2 = f1;

6. System.*out*.println("f2[0]= " + f2[0]);

7. } }

**Kết quả sẽ là gì?**

* 1. Lỗi biên dịch tại dòng 5
  2. Xuất ra màn hình f2[0] = NaN.
  3. Lỗi biên dịch tại dòng 6.
  4. Xuất ra màn hình f2[0] = 0.0.

1. **Cho trước đoạn chương trình sau:**

**public** **class** Book {

**public** Book() { }

**static**{

System.*out*.println("Static"); }

**public** String toString(){

**return** "Book";

}}

**public** **class** Test {

**public** **static** **void** main(String[] args) {

**new** Book();

System.*out*.println("Test");

}}

**Kết quả sẽ là gì khi biên dịch và chạy chương trình?**

1. Kết quả : Test
2. Kết quả : Static Test
3. Kết quả : Book Test
4. Kết quả : Static

***Ghi chú:***

Sinh viên chỉ chọn 1 đáp án đúng nhất cho mỗi câu hỏi.

Sinh viên khoanh tròn vào đáp án mà mình lựa chọn, nếu muốn bỏ chọn lựa đó thì đánh dấu chéo vào phương án đã chọn lựa rồi chọn đáp án khác. Nếu muốn chọn lại đáp án đã bỏ thì tô đen kín khoanh tròn.

**--- HẾT ---**

**Trưởng Bộ Môn duyệt Giảng viên ra đề**

*(Ký và ghi rõ họ tên) (Ký và ghi rõ họ tên)*