**ĐẠI HỌC NÔNG LÂM TP. HỒ CHÍ MINH LẦN THI THỨ 1**

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

**ĐỀ THI MÔN CHUYÊN ĐỀ JAVA- ĐỀ THI SỐ 1**

**LỚP CD06TH - THỜI GIAN LÀM BÀI 60 PHÚT**

*(Không được sử dụng tài liệu)*

1. **Cho đoạn chương trình sau:**

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

map.put("key12", "orange");

map.put("key99", "mango");

map.put("key8", "lemon");

map.put("key12", "banana");

Set<String> keySet = map.keySet();

keySet = **new** TreeSet<String>(keySet);

System.*out*.println(keySet);

**Kết quả sẽ là gì?**

* 1. [key12, key99, key8, key12]
  2. [key8, key12, key12, key99]
  3. [key12, key8, key99] Sai0
  4. [key8, key12, key99] chon
  5. Báo lỗi

1. **Cho đoạn chương trình sau đây:**

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

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

} **while** (i < 5);

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

**Kết quả sẽ là gì? Sai0**

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

1. **Khai báo hàm main nào dưới đây là không hợp lệ?**
   1. **public static void** main(String[] args) {}
   2. **public static void** main(String args[]) {}
   3. **public static** void main(String args) {}
   4. **public static final void** main(String[] args) {}
2. **Cho đoạn chương trình sau:**
   * 1. **int** number = 3;
     2. **try** { number = Integer.*parseInt*("invalid");
     3. } **catch** (NumberFormatException e) {
     4. **int** number = 0;
     5. }
     6. **finally**{ System.*out*.println("number = " + number);
     7. }

**Chọn câu trả lời đúng?**

* 1. Kết quả in ra là 0 Sai0
  2. Lỗi ở dòng thứ 1
  3. Lỗi ở dòng thứ 2 chon
  4. Lỗi ở dòng thứ 4
  5. Lỗi ở dòng thứ 6

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

**interface** A{

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

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

}

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

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

}

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

1. **Cho đoạn chương trình sau:**

**boolean** bool = **false**;

**if** (bool = **true**)

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

**if** (bool)

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

**if** (!bool)

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

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

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

* 1. c d
  2. a d
  3. b d
  4. a b d chon
  5. a c d
  6. c b d

1. **Cho đoạn chương trình sau:**

**int** i = 0;

**for** (i = 2; i < 4; i += 3)

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

System.*out*.println(i);

**Kết quả sẽ là gì?**

* 1. 2 3
  2. 2 5 chon
  3. 0 3 6
  4. 0 1 2 3

1. **Cho đoạn lệnh sau:**

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

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

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

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

**new** Alpha();

}

}

**Kết quả sẽ là gì?**

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

1. **Cho đoạn chương trình sau:**

**public** **class** Alpha {

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

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

MyEnum myEnum = MyEnum.*B*;

**switch** (myEnum) {

**case** *A*:

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

**default**:

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

**case** *C*:

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

**case** *D*:

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

}

}

}

**Kết quả sẽ là gì?**

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

1. **Cho đoạn chương trình sau:**

**for** (**int** i = 0; ; i++)

method();

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

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

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. }
     9. }

**Lựa chọn nào sau đây khi thêm vào dòng số 7 sẽ gây nên lỗi java.lang.ClassCastException**

* 1. B a = x;
  2. A a = (D) x; Sai0
  3. A a = (B)x;
  4. C a = (C)(B)x; chon

//chi ep kieu tu lop con ve lop cha ko the ep kieu nguoc lai

1. **Cho đoạn chương trình sau:**

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

**private** **int** a;

**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 = 11 chon
  2. a = 10 Sai0
  3. Lỗi biên dịch
  4. Một ngoại lệ được ném ra trong thời gian thực hiện chương trình

1. **Cho đoạn lệnh sau:**

**class** A {

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

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

**this**();

*y* \*= 3;

}

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

}

**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. 2
  2. 4
  3. 6
  4. 11 chon
  5. 12

1. **Cho đoạn chương trình sau:**
   * 1. **int**[] y = { 1, 2, 3, 4, 5, 6 };
     2. System.*out*.print("Output: ");
     3. // insert code here
     4. System.*out*.print(x + " ");

**Thêm vào dòng 3 đoạn code nào sau đây để có kết quả là** “**Output: 1 2 3 4 5 6”**

* 1. **for** (**int** x : y) Sai0
  2. foreach(y as x)
  3. **for**(**int** x = y[])
  4. foreach(**int** x : y) chon
  5. **for**(**int** x = 1; x = y[]; ++x)

1. **Cho đoạn chương trình sau:**

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

**public** **static** **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. 0 chon
  2. 3
  3. Lỗi biên dịch
  4. Một ngoại lệ ném ra ở thời gian thực hiện chương trình

1. **Cho đoạn chương trình sau:**

**class** A {

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

**return** 1;

}

}

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

**public** **byte** 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. 1
  2. 2 chon
  3. Lỗi biên dịch
  4. Một ngoại lệ ném ra ở thời gian thực hiện chương trình

1. **Cho đoạn chương trình sau:**

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

**public** **class** SinhVien {

**private** String maSV;

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

**public** **boolean** equals(SinhVien sv) {

**return** **this**.maSV.equalsIgnoreCase(sv.maSV);

}

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

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

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

Set<SinhVien> s = **new** HashSet<SinhVien>();

s.add(sv1);

s.add(sv2);

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

}

}

**Kết quả sẽ là gì?**

* 1. 1 chon
  2. 2 Sai0
  3. Lỗi biên dịch
  4. Một ngoại lệ ném ra ở thời gian thực hiện chương trình

1. **Cho đoạn lệnh như sau:**

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

**public** **static** **void** method1(){//more code here

}

**public** **void** method2(){//more code here

}

}

**Lựa chọn nào sau đây là đúng?**

* 1. method1() có thể gọi trực tiếp method2()
  2. method2() có thể gọi trực tiếp method1() chon
  3. Cả 2 phương thức đều có thể gọi trực tiếp nhau
  4. Cả 2 phương thức đều không thể gọi trực tiếp nhau

1. **Cho đoạn chương trình sau:**
   * 1. **class** Outer{
     2. **public** **static** **class** Inner{}
     3. }
     4. **public** **class** Test{
     5. //insert code here
     6. }

**Lựa chọn nào sau đây là đúng khi muốn tạo một đối tượng thuộc lớp Inner ở dòng thứ 5**

**a.** Inner in = **new** Inner();

* 1. Outer.Inner in = **new** Outer.Inner(); Sai0
  2. Outer out = **new** Outer(); out.Inner = **new** out.Inner();
  3. Ở dòng 5 không thể tạo ra một đối tượng thuộc lớp Inner

1. **Lựa chọn nào dưới đây là không lỗi?**
   1. **abstract** **class** Canine {**public** void speak();}
   2. **abstract** **class** Canine {**public** **void** speak() {}} chon
   3. **class** Canine {**public** **abstract** **void** speak();}
   4. **class** Canine {**public** **void** speak();}
2. **Cho đoạn chương trình sau:**

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

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

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

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

**Kết quả sẽ là gi?**

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

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

1. **Cho đoạn chương trình sau:**

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

**int** x = 1;

**public** **void** method(**int** x) {

x += **this**.x;

System.*out*.println(x);

}

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

Test t = **new** Test();

t.method(10);

}

}

**Kết quả sẽ là gì?**

* 1. 2
  2. 11 chon
  3. 20
  4. Lỗi biên dịch

1. **Cho đoạn chương trình sau:**
   * 1. **public** **class** MyObj {
     2. **public** MyObj go(MyObj cb) {
     3. cb = **null**;
     4. **return** cb;
     5. }
     6. **public** **static** **void** main(String[] args) {
     7. MyObj c1 = **new** MyObj();
     8. MyObj c2 = **new** MyObj();
     9. MyObj c3 = c1.go(c2);
     10. }
     11. }

**Có bao nhiêu đối tượng sẫn sàng cho quá trình gom rác sau dòng 9**

* 1. 0
  2. 1 Sai0
  3. 2 chon
  4. 3
  5. Lỗi biên dịch

1. **Cho đoạn chương trình sau:**

String test = "1a2b3c4d5";

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

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

**Kết quả sẽ là gì?**

* 1. [1a2b3c4d5]
  2. [a, b, c, d] chon
  3. [, a, b, c, d] Sai0
  4. [, a, b, c, d, ]
  5. [, 1, a, 2, b, 3, c, 4, d, 5]

1. **Cho đoạn chương trình sau:**

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

Mixer m1;

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. hi
  2. hi hi chon
  3. hi, sau đó báo lỗi
  4. hi hi, sau đó báo lỗi

1. **Cho đoạn chương trình sau:**
   * 1. //insert code here
     2. set.add(34);
     3. set.add(1);
     4. System.*out*.println(set);

**Đoạn lệnh nào dưới đây khi thêm vào dòng 1 sẽ đảm bảo rằng kết quả in ra là [1, 34]**

**a.** Set<Integer> set = **new** ArraySet<Integer>();

* 1. Set<Integer> set = **new** HashSet<Integer>();
  2. Set<Integer> set = **new** TreeSet<Integer>(); chon
  3. Set<Integer> set = **new** SortedSet<Integer>();

1. **Trong các khai báo cho hàm main sau, khai báo nào là không hợp lệ?**
2. **public** **static** **void** main(String[] args) {}
3. **public** **static** **Void** main(String[] args) {} chon
4. **public** **static** **final** **void** main(String[] args) {}
5. **public** **final** **static** **void** main(String[] args) {}
6. **Cho đoạn chương trình sau:**

**void** waitForSignal() {

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

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

obj.wait();

obj.notify();

}

}

**Đoạn lệnh này cần ném ra lỗi nào sau đây?**

1. NullPointerException
2. InterruptedException Sai0
3. IllegalStateception
4. Không cần ném ra lỗi gì hết
5. **Cho đoạn chương trình sau:**
   * 1. Object[] array = { **new** Double(1.0), **new** Object(), **new** Boolean(**false**)};
     2. System.*out*.println(Arrays.*binarySearch*(array, **new** Double(1)));

**Kết quả sẽ là gì?**

* 1. In ra 0
  2. Lỗi ở dòng 1
  3. Lỗi ở dòng 2
  4. ClassCastException chon

1. **Cho đoạn chương trình sau:**
   * 1. **public** **class** MyClass {
     2. **private** **int** i;
     3. // insert code here
     4. **public** MyClass(**int** i) {
     5. **this**.i = i;
     6. }
     7. }
     8. **class** MyOtherClass **extends** MyClass {//lỗi
     9. }

**Bạn cần thêm dòng lệnh nào sau đây vào dòng 3 để dòng 8 hết lỗi?**

1. **public** MyClass() {**this**(1); } Sai0
2. **public** MyClass() {MyClass(1); }
3. **public** MyClass() { MyOtherClass (); }
4. **public** MyOtherClass () { MyClass (); }
5. **public** MyClass() { MyOtherClass (1); }
6. **Cho đoạn chương trình sau:**

**public** **class** Forest **implements** Serializable {

**private** Tree tree = **new** Tree(**this**);

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

Forest f = **new** Forest();

**try** {

FileOutputStream fs = **new** FileOutputStream("Forest.ser");

ObjectOutputStream os = **new** ObjectIntputStream(fs);

os.writeObject(f);

os.close();

} **catch** (Exception exp) { exp.printStackTrace();}

}

}

**class** Tree **implements** Serializable {

}

**Kết quả sẽ là gì?**

1. Báo lỗi (chon)
2. Ném ra ngoại lệ trong quá trình biên dịch
3. Một thể hiện của lớp Forest được ghi vào file Forest.ser
4. Một thể hiện của lớp Forest và một thể hiện của lớp Tree được ghi vào file Forest.ser
5. **Phát biểu nào sau đây là sai với java.util.Map?**
6. Các giá trị (values) trong map được phép trùng nhau
7. Các giá trị (values) trong map được phép là **null**
8. Các khóa (key) của map phải là duy nhất trừ trường hợp nó là **null (chon)**
9. Các khóa (key) của map phải là duy nhất kể cả trường hợp nó là **null Sai0**
10. **Cho đoạn chương trình sau:**
    * 1. **public** **class** Test **implements** Runnable {
      2. **public** **void** run() {
      3. System.*out*.print(" running ");
      4. }
      5. **public** **static** **void** main(String[] args) **throws** Exception {
      6. Thread t = **new** Thread(**new** Test());
      7. System.*out*.print("begin");
      8. t.start();
      9. //insert code here
      10. System.*out*.print("end");
      11. }}

**Bạn cần thêm đoạn lệnh nào sau đây vào dòng 9 để chương trình in ra “begin running end”?**

1. t.notify(); chon
2. t.interrupt();
3. t.join(); Sai0
4. t.wait();
5. **Cho đoạn chương trình sau:**

List<String> list = **new** LinkedList<String>();

list.add("aa");

list.add("Aa");

list.add("aA");

System.*out*.println(**new** TreeSet<String>(list));

**Kết quả sẽ là gì?**

1. [Aa, aA, aa] Sai0
2. [aa, aA, Aa] chon
3. [aA, aa, Aa]
4. Không xác định được
5. **Cho đoạn chương trình sau:**

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

**public** **enum** Fruit{*BANANA*, *ORANGE*, *COCONUT*}

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

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

set.add(Fruit.*BANANA*);

set.add(Fruit.*ORANGE*);

set.add(Fruit.*COCONUT*);

System.*out*.println(set);

}

}

**Kết quả sẽ là gì?**

1. [BANANA, ORANGE, COCONUT] Sai0
2. [ORANGE , BANANA, COCONUT]
3. [COCONUT, BANANA, ORANGE] chon
4. [BANANA, COCONUT , ORANGE]
5. **Cho đoạn chương trình sau:**
   * 1. **class** A{
     2. **public** **void** method(**int**...x){}
     3. **public** **void** method(**int** x){}
     4. }
     5. **class** B **extends** A{
     6. //insert code here
     7. }

**Trong các phương thức sau, phương thức nào là không thêm được vào dòng 6?**

1. **public** **void** method(**int** x){}
2. **public** **void** method(**int** x, **int** y){}
3. **public** **int** method(**int** x){} chon
4. **public** **void** method(String x){}
5. **Cho đoạn chương trình sau:**

PriorityQueue<String> queue = **new** PriorityQueue<String>();

queue.add("f");

queue.add("b");

queue.add("a");

queue.add("d");

System.*out*.println(queue);

**Kết quả sẽ là gì?**

1. [a, b, d, f] chon
2. [a, d, b, f]
3. [a, b, f, d]
4. [f, d, b, a]
5. **Cho đoạn chương trình sau:**

**int**[] array = {2,4,5};

List l = Arrays.*asList*(array);

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

**Kết quả sẽ là gì?**

1. 0
2. 1 Sai0
3. 2
4. 3 chon

***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)*