

# **Object-oriented Programming and Software Engineering CITS1001**

## **Multiple-choice Mid-semester Test**

**Semester 1, 2018**

- **Mark your solutions on the provided answer page, by filling in the appropriate circles.**
- **Write your name and student number on the answer sheet, and also fill in the circles for both.**
- **The papers will be marked by an automatic scanner, so make sure that your selections are clear.**
- **There are fifteen questions: ignore options 16–125 on the answer sheet.**
- **Use the blank pages at the end for rough work.**
- **Feel free to separate the answer sheet from the question sheets, but hand in both at the end of the test.**
- **The time allowed is forty minutes.**

1. What value does `mystery(12)` return?

```
public int mystery(int x)
{
    int n = 0;
    for (int k = x; k > 0; k = k / 2)
        if (k % 2 != 0) n++;
    return n;
}
```

- a) 0
- b) 1
- c) 2 \*\*\*
- d) 3
- e) 4

2. Given `int` variables `x > 0` and `y > 0`, which of these statements is true?

- a) `x % y` is 0 only when `x == y`
- b) `x % y` is always less than `x / y`
- c) `x % y` is always less than `y` \*\*\*
- d) `x % y` is always equal to `y % x`
- e) `x % y` is always greater than 0

3. What is the value of the expression `false == false == false == false`?

- a) It contains a syntax error
- b) It contains a type error
- c) It contains a run-time error
- d) `false`
- e) `true` \*\*\*

4. A `String` is represented in a Java program as

- a) an object. \*\*\*
- b) a primitive value.
- c) an array.
- d) an `ArrayList`.
- e) a binary number.

5. Which of these statements uses the correct syntax to copy the first element of the array `xs` into its last element?

- a) `xs[0] = xs[xs.length - 1];`
- b) `xs[xs.length - 1] = xs[0];` \*\*\*
- c) `xs.add(xs.get(0));`
- d) `xs[xs.length] = xs[0];`
- e) `xs[0] = xs[xs.size() - 1];`

6. What are the values of `m` and `n` after these statements?

```
int m = 10;
int n = 1;
n = n * m;
m = n / m;
n = n / m;
```

- a) 1, 1
- b) 1, 10 \*\*\*
- c) 10, 1
- d) 10, 10
- e) 100, 100

7. What value does `mystery(29)` return?

```
public int mystery(int n)
{
    int[] xs = new int[n+1];
    xs[1] = 33;
    for (int k = 2; k <= n; k++)
        xs[k] = xs[k-1] - xs[k-2];
    return xs[n];
}
```

- a) -66
- b) -33 \*\*\*
- c) 0
- d) 33
- e) 66

8. A method that has return type `void` and that assigns values to an object's instance variables is known as

- a) an accessor method.
- b) a class method.
- c) a constructor method.
- d) a general method.
- e) a mutator method. \*\*\*

9. If a class has multiple constructors, they must all have

- a) different argument types. \*\*\*
- b) different names.
- c) different return types.
- d) different statements.
- e) different visibility.

10. What value does `mystery(4, -2)` return?

```
public int mystery(int a, int b)
{
    while (a > b)
    {
        a = a + b;
        b = a * b;
    }
    return b;
}
```

- a) -4
- b) -2
- c) 2
- d) 8 \*\*\*
- e) 16

11. What sort of variable should be used to store data that is important throughout an object's lifespan?

- a) A constructor variable
- b) A heap variable
- c) A field variable \*\*\*
- d) A method variable
- e) A parameter variable

12. Which of these evaluates to the same result as the expression  $88 - 66 / 5 - 3$ ?

- a)  $88 - ((66 / 5) - 3)$
- b)  $(88 - (66 / 5)) - 3$  \*\*\*
- c)  $(88 - 66) / (5 - 3)$
- d)  $((88 - 66) / 5) - 3$
- e)  $88 - (66 / (5 - 3))$

13. If  $k$  is a big number, what does `mystery(k)` return?

```
public int mystery(int k)
{
    int z = k % 10;
    while (k >= 10)
    {
        z = k % 10;
        k = k / 10;
    }
    return z;
}
```

- a)  $k$ 's least significant digit
- b)  $k$ 's second least significant digit
- c)  $k$ 's middle digit
- d)  $k$ 's second most significant digit \*\*\*
- e)  $k$ 's most significant digit

14. Given this statement, how many `Strings` can `names` store?

```
String[][] names = new String[5][6];
```

- a) 5
- b) 6
- c) 11
- d) 30 \*\*\*
- e) 56

15. How many times does this for-loop execute?

```
for (int k = 53; k > 6; k = k / 2)
```

- a) 1
- b) 2
- c) 3 \*\*\*
- d) 4
- e) 5