

Chapter 4 Control Statements: Part 1

Section 4.2 Algorithms

4.2 Q1: Which of the following is *not* an algorithm?

- a. A recipe.
- b. Operating instructions.
- c. Textbook index.
- d. Shampoo instructions (lather, rinse, repeat).

ANS: c. Textbook index. A textbook index contains useful information, but does not specify actions and the order in which they are to be performed.

Section 4.3 Pseudocode

4.3 Q1: Which of the following is *true*?

- a. Pseudocode is used to describe an algorithm.
- b. Pseudocode is *not* an actual computer programming language.
- c. Pseudocode is used to describe executable statements that will eventually be translated by the programmer into a program.
- d. All of the above.

ANS: d. All of the above.

Section 4.4 Control Structures

4.4 Q1: Which of the following is *not* a benefit of “goto-less programming”?

- a. Easier to debug and modify
- b. Shorter
- c. Clearer
- d. More likely to be bug free

ANS: b. Shorter

4.4 Q2: Which of the following is *not* a control structure:

- a. Sequence structure.
- b. Selection structure.
- c. Repetition structure.
- d. Declaration structure.

ANS: d. Declaration structure.

Sequence Structure in Java

4.4 Q3: Which of the following is the shape of an action-state symbol?

- a. Diamond.
- b. Circle.
- c. Rectangle with left and right sides replaced with arcs curving outward.
- d. Rounded rectangle.

ANS: c. Rectangle with left and right sides replaced with arcs curving outward.

4.4 Q4: Which statement is *false*?

- a. Unless directed otherwise, the computer executes Java statements one after the other in the order in which they're written.
- b. Activity diagrams normally show the Java code that implements the activity.
- c. Like pseudocode, activity diagrams help programmers develop and represent algorithms.
- d. The arrows in the activity diagram represent transitions, which indicate the order in which the actions represented by the action states occur.

ANS: b. Activity diagrams normally show the Java code that implements the activity.

Selection Statements in Java

4.4 Q5: Which of the following is a double-selection control statement?

- a. do...while
- b. for
- c. if...else
- d. if

ANS: c. if...else

Repetition Statements in Java

4.4 Q6: Which of the following is *not* a Java keyword?

- a. do
- b. next
- c. while
- d. for

ANS: b. next

Section 4.5 if Single-Selection Statement

4.5 Q1: What is output by the following Java code segment?

```
int temp = 200;

if (temp > 90)
    System.out.println("This porridge is too hot.");

if (temp < 70)
    System.out.println("This porridge is too cold.");

if (temp == 80)
    System.out.println("This porridge is just right!");
```

- a. This porridge is too hot.
- b. This porridge is too cold.
- c. This porridge is just right!
- d. None of the above.

ANS: a. This porridge is too hot.

4.5 Q2: A decision symbol in an activity diagram takes the shape of a _____.

- a. Diamond.
- b. Rectangle.
- c. Circle.
- d. Triangle.

ANS: a. Diamond.

4.5 Q3: Which of the following is *not* represented graphically in activity diagrams for control structures?

- a. Transition arrow.
- b. Attribute.

- c. Action state.
- d. Decision symbol.

ANS: b. Attribute.

Section 4.6 *if...else* Double-Selection Statement

Conditional Operator (? :)

4.6 Q1: Which of the following statements about the conditional operator (*? :*) is *false*?

- a. The conditional operator is a ternary operator, meaning that it takes three operands.
- b. The first operand is a boolean expression.
- c. The second operand is the result value if the condition evaluates to *false*.
- d. The second operand is the result value if the condition evaluates to *true*.

ANS: c. The second operand is the result value if the condition evaluates to false.

Nested if...else Statements

4.6 Q2: What is output by the following Java code segment?

```
int temp = 180;

if (temp > 90)
{
    System.out.println("This porridge is too hot.");

    // cool down
    temp = temp - (temp > 150 ? 100 : 20);
}
else
{
    if (temp < 70)
    {
        System.out.println("This porridge is too cold.");

        // warm up
        temp = temp + (temp < 50 ? 30 : 20);
    }
}

if (temp == 80)
    System.out.println("This porridge is just right!");
```

- a. This porridge is too hot.
- b. This porridge is too cold.
This porridge is just right!
- c. This porridge is just right!
- d. None of the above.

**ANS: d. None of the above. (The output will be
This porridge is too hot.
This porridge is just right!)**

Dangling-else Problem

4.6 Q3: Which of the following would *not* be used to *clarify* a dangling-else?

- a. Indentation.
- b. Parentheses ().
- c. Braces {}.
- d. Comment //.

ANS: b. Parentheses ().

Blocks

4.6 Q4: The empty statement is denoted by what symbol?

- a. Semicolon ;
- b. Parentheses ()
- c. Braces {}
- d. Colon :

ANS: a. Semicolon ;

4.6 Q5: Which of the following statements is *true*?

- a. Both syntax errors and logic errors are caught by the compiler.
- b. Both syntax errors and logic errors have effects at execution time.
- c. Syntax errors are caught by the compiler. Logic errors have effects at execution time.
- d. Logic errors are caught by the compiler. Syntax errors have effects at execution time.

ANS: c. Syntax errors are caught by the compiler. Logic errors have their effects at execution time.

Section 4.7 Student Class: Nested if...else Statements

4.7 Q1: Which of the following statements is *false*?

- a. You should not call methods from constructors.
- b. Nested if statements can be useful for validating values.
- c. Logical operators can express nested if statements more concisely.
- d. One problem with calling methods from constructors is that it can lead to duplicated validation code.

ANS: d. One problem with calling methods from constructors is that it can lead to duplicated validation code. Actually, you should not call methods from constructors and this good practice leads to duplicate validation code.

Section 4.8 while Repetition Statement

4.8 Q1: What is output by the following Java code segment?

```
int temp = 180;

while (temp != 80)
{
    if (temp > 90)
    {
        System.out.print("This porridge is too hot! ");

        // cool down
        temp = temp - (temp > 150 ? 100 : 20);
    }
    else
    {
        if (temp < 70)
        {
```

```

        System.out.print(
            "This porridge is too cold! ");

        // warm up
        temp = temp + (temp < 50 ? 30 : 20);
    }
}

if (temp == 80)
    System.out.println("This porridge is just right!");

```

- a. This porridge is too cold! This porridge is just right!
- b. This porridge is too hot! This porridge is just right!
- c. This porridge is just right!
- d. None of the above.

ANS: b. This porridge is too hot! This porridge is just right!

4.8 Q2: Which of the following is *not* an error (either a syntax error or a logic error)?

- a. Neglecting to include an action in the body of a while statement that will eventually cause the condition to become false.
- b. Spelling a keyword (such as while or if) with a capitalized first letter.
- c. Using a condition for a while statement that is initially false.
- d. An infinite loop.

ANS: c. Using a condition for a while statement that is initially false.

4.8 Q3: In an activity diagram, the merge symbol has the same shape as what other symbol?

- a. Decision symbol.
- b. Action symbol.
- c. Transition arrows.
- d. Initial state.

ANS: a. Decision symbol.

Section 4.9 Formulating Algorithms: Counter-Controlled Repetition

Pseudocode Algorithm with Counter-Controlled Repetition

4.9 Q1: Counter-controlled repetition is also known as:

- a. Definite repetition
- b. Indefinite repetition
- c. Multiple-repetition structure
- d. Double-repetition structure

ANS: a. Definite repetition

Implementing Counter-Controlled Repetition

4.9 Q2: How many times is the body of the loop below executed?

```

int counter = 1;

while (counter > 20)
{
    // body of loop
    counter = counter - 1;
} // end while

```

- a. 19.
- b. 20.
- c. 21.
- d. 0.

ANS: d. 0.

4.9 Q3: Where can local variables declared within a method's body be used?

- a. Only in that method between the line in which they were declared and the closing brace of that method.
- b. Same as (a), but not within while or if statements.
- c. Only within while or if statements within the method in which they were declared.
- d. Anywhere within the class.

ANS: a. Only in that method between the line in which they were declared and the closing brace of that method.

Notes on Integer Division and Truncation

4.9 Q4: Which statement is *true*?

- a. Dividing two integers results in integer division.
- b. With integer division, any fractional part of the calculation is lost.
- c. With integer division, any fractional part of the calculation is truncated.
- d. All of the above.

ANS: d. All of the above.

Section 4.10 Formulating Algorithms: Sentinel-Controlled Repetition

4.10 Q1: Which of the following terms is *not* used to refer to a sentinel value that breaks out of a while loop?

- a. signal value.
- b. maximum value.
- c. dummy value.
- d. flag value.

ANS: b. maximum value.

4.10 Q2: Sentinel-controlled repetition is also known as:

- a. Definite repetition.
- b. Indefinite repetition.
- c. Multiple repetition.
- d. Double repetition.

ANS: b. Indefinite repetition.

Developing the Pseudocode Algorithm with Top-Down, Stepwise Refinement: The Top and First Refinement

4.10 Q3: Which of the following is *not* a common name for one of the *three phases* that a program often can be split into using pseudocode?

- a. Termination phase
- b. Initialization phase
- c. Processing phase

d. Action phase

ANS: d. Action phase

Implementing Sentinel-Controlled Repetition

4.10 Q4: Which of the following segments is a proper way to call the method `readData` four times?

a. `double k = 0.0;`

```
while (k != 4)
{
    readData();
    k = k + 1;
}
```

b. `int i = 0;`

```
while (i <= 4)
{
    readData();
    i = i + 1;
}
```

c. `int i = 0;`

```
while (i < 4)
{
    readData();
}
```

d. `int i = 0;`

```
while (i < 4)
{
    readData();
    i = i + 1;
}
```

ANS: d.

```
int i = 0;
```

```
while (i < 4)
{
    readData();
    i = i + 1;
}
```

Explicitly and Implicitly Converting Between Primitive Types

4.10 Q5: In an expression containing values of the types `int` and `double`, the _____ values are _____ to _____ values for use in the expression.

a. `int`, promoted, `double`.

b. `int`, demoted, `double`.

c. `double`, promoted, `int`.

d. `double`, demoted, `int`.

ANS: a. `int`, promoted, `double`.

4.10 Q6: Which of the following statements is *false*?

- a. To ensure that the operands in a mixed-type expression are of the same type, Java performs implicit conversion on selected operands.
- b. Cast operators are unary operators.
- c. Cast operators associate from right to left and are one level lower in precedence than the multiplicative operators.
- d. Cast operators are formed by placing parentheses around the name of a type.

ANS: c. Cast operators associate from right to left and are one level lower in precedence than the multiplicative operators.

Section 4.11 Formulating Algorithms: Nested Control Statements

Complete Second Refinement of Pseudocode and Conversion to Class Analysis

4.11 Q1: Local variables must be _____.

- a. initialized when they're declared.
- b. initialized before their values are used in an expression.
- c. declared and initialized in two steps.
- d. declared at the top of the method's body.

ANS: b. initialized before their values are used in an expression.

4.11 Q2: Which of the following statements is *true*?

- a. A while statement cannot be nested inside another while statement.
- b. An if statement cannot be nested inside another if statement.
- c. A while statement cannot be nested inside an if statement.
- d. None of the above is true.

ANS: None of the above is true.

Section 4.12 Compound Assignment Operators

4.12 Q1: Which of the following code segments does *not* increment `val` by 3:

- a. `val += 3;`
- b. `val = val + 1;`
`val = val + 1;`
`val = val + 1;`
- c. `c = 3;`
`val = val + (c == 3 ? 2 : 3);`
- d. All of the above increment `val` by 3.

**ANS: c. `c = 3;`
`val = val + (c == 3 ? 2 : 3);`**

4.12 Q2: What does the expression `x %= 10` do?

- a. Adds 10 to the value of `x`, and stores the result in `x`.
- b. Divides `x` by 10 and stores the remainder in `x`.
- c. Divides `x` by 10 and stores the integer result in `x`.
- d. None of the above.

ANS: b. Divides `x` by 10 and stores the remainder in `x`.

Section 4.13 Increment and Decrement Operators

4.13 Q1: Which of the following operators associates from left to right?

- a. =
- b. ?:
- c. %=
- d. /

ANS: d. /

4.13 Q2: What is the result value of c at the end of the following code segment?

```
int c = 8;  
c++;  
++c;  
c %= 5;
```

- a. 0.
- b. 1.
- c. 3.
- d. None of the above.

ANS: a. 0.

Section 4.14 Primitive Types

4.14 Q1: Java is considered a *strongly typed language* because:

- a. The primitive types in Java are portable across all computer platforms that support Java.
- b. Java requires all variables to have a type before they can be used in a program.
- c. Instance variables of the primitive types are automatically assigned a default value.
- d. All of the above.

ANS: b. Java requires all variables to have a type before they can be used in a program

4.14 Q2: Which of the following is *not* a primitive type?

- a. char
- b. float
- c. String
- d. int

ANS: c. String

4.14 Q3: Which primitive type can hold the *largest* value?

- a. int
- b. long
- c. float
- d. double

ANS: d. double

4.14 Q4: What is the size in bits of an int?

- a. 8
- b. 16
- c. 32
- d. 64

ANS: c. 32

Section 4.14 (Optional) GUI and Graphics Case Study: Creating Simple Drawings

4.15 Q1: In Java graphics, coordinate units are measured in _____.

- a. pixlets.
- b. pixels.
- c. points.
- d. pikels.

ANS: b. pixels.

4.15 Q2: Keyword _____ indicates the inheritance relationship.

- a. extends
- b. inherits
- c. super
- d. parent

ANS: a. extends