Due to the print book page limit, we cannot inlcude all good CheckPoint questions in the physical book. The CheckPoint on this Website may contain extra questions not printed in the book. The questions in some sections may have been reordered as a result. Nevertheless, it is easy to find the CheckPoint questions in the book on this Website. Please send suggestions and errata to Dr. Liang at y.daniel.liang@gmail.com. Indicate the book, edition, and question number in your email. Thanks!

Chapter 5 Check Point Questions

Section 5.2

▼5.2.1

Analyze the following code. Is count < 100 always true, always false, or sometimes true or sometimes false at Point A, Point B, and Point C?

```
int count = 0;
while (count < 100) {
  // Point A
  System.out.println("Welcome to Java!");
  count++;
  // Point B
}
// Point C
```

count < 100 is always true at Point A.
count < 100 always false at Point C.
count < 100 is sometimes true or sometimes false at Point B.

Hide Answer Read Answer

▼5.2.2

How many times are the following loop bodies repeated? What is the output of each loop?

```
(a)
int i = 1;
while (i < 10)
  if (i % 2 == 0)
    System.out.println(i);

(b)
int i = 1;
while (i < 10)
  if (i % 2 == 0)
    System.out.println(i++);

(c)
int i = 1;
while (i < 10)
  if ((i++) % 2 == 0)
    System.out.println(i);
```

(a) Infinite number of times.
(b) Infinite number of times.
(c) The loop body is executed nine times. The output is 3, 5, 7, 9 on separate lines.

Hide Answer Read Answer

▼5.2.3

What is the output of the following code? Explain the reason.

```
int x = 80000000;

while (x > 0)
  x++;

System.out.println("x is " + x);
```

x is -2147483648
The reason: When a variable is assigned a value that is too large (in size) to be stored, it causes overflow.
2147483647 + 1 is actually -2147483648

Hide Answer Read Answer

Section 5.3

▼5.3.1

What is wrong if guess is initialized to 0 in line 11 in Listing 5.3?

It would be wrong if it is initialized to a value between 0 and 100, because it could be the number you attempt to guess.

Hide Answer Read Answer

Section 5.4

▼5.4.1

Revise the code to increase the count only when the user gets a correct answer.

In the while loop, change the loop continuation condition to (correctCount < NUMBER_OF_QUESTIONS).

Hide Answer Read Answer

Section 5.5

▼5.5.1

Suppose the input is 2 3 4 5 0. What is the output of the following code?

```
import java.util.Scanner;

public class Test {
  public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    int number, max;
    number = input.nextInt();
    max = number;

    while (number != 0) {
      number = input.nextInt();
```

```
      if (number > max)
        max = number;
    }

    System.out.println("max is " + max);
    System.out.println("number " + number);
  }
}
```

```
max is 5
number 0
```

Hide Answer Read Answer

Section 5.6

▼5.6.1

Suppose the input is 2 3 4 5 0. What is the output of the following code?

```
import java.util.Scanner;

public class Test {
  public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    int number, max;
    number = input.nextInt();
    max = number;

    do {
      number = input.nextInt();
      if (number > max)
        max = number;
    } while (number != 0);

    System.out.println("max is " + max);
    System.out.println("number " + number);
  }
}
```

```
max is 5
number 0
```

Hide Answer Read Answer

▼5.6.2

What are the differences between a while loop and a do-while loop? Convert the following while loop into a do-while loop.

```
Scanner input = new Scanner(System.in);
int sum = 0;
System.out.println("Enter an integer " +
  "(the input ends if it is 0)");
int number = input.nextInt();
while (number != 0) {
```

```
  sum += number;
  System.out.println("Enter an integer " +
    "(the input ends if it is 0)");
  number = input.nextInt();
}
```

The difference between a do-while loop and a while loop is the order of evaluating the continuation-condition and executing the loop body. In a while loop, the continuation-condition is checked and then, if true, the loop body is executed. In a do-while loop, the loop body is executed for the first time before the continuation-condition is evaluated.

```
Scanner input = new Scanner(System.in);
int sum = 0;
int number;
do {
  System.out.println("Enter an integer " +
    "(the input ends if it is 0)");
  number = input.nextInt();
  sum += number;
} while (number != 0);
```

Hide Answer Read Answer

Section 5.7

▼5.7.1

Do the following two loops result in the same value in sum?

```
(a)
for (int i = 0; i < 10; ++i) {
  sum += i;
}

(b)
for (int i = 0; i < 10; i++) {
  sum += i;
}
```

Same. When the i++ and ++i are used in isolation, their effects are same.

Hide Answer Read Answer

▼5.7.2

What are the three parts of a for loop control? Write a for loop that prints the numbers from 1 to 100.

The three parts in a for loop control are as follows: The first part initializes the control variable. The second part is a Boolean expression that determines whether the loop will repeat. The third part is the adjustment statement, which adjusts the control variable.

```
for (int i = 1; i <= 100; i++)
  System.out.println(i);
```

Hide Answer Read Answer

▼5.7.3

Suppose the input is 2 3 4 5 0. What is the output of the following code?

```
import java.util.Scanner;

public class Test {
  public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    int number, sum = 0, count;

    for (count = 0; count < 5; count++) {
      number = input.nextInt();
      sum += number;
    }

    System.out.println("sum is " + sum);
    System.out.println("count is " + count);
  }
}
```

```
sum is 14
count is 5
```

Hide Answer Read Answer

▼5.7.4

What does the following statement do?

```
for ( ; ; ) {
  // Do something
}
```

The loop keeps doing something indefinitely.

Hide Answer Read Answer

▼5.7.5

If a variable is declared in a for loop control, can it be used after the loop exits?

No. The scope of the variable is inside the loop.

Hide Answer Read Answer

▼5.7.6

Convert the following for loop statement to a while loop and to a do-while loop:

```
long sum = 0;
for (int i = 0; i <= 1000; i++)
  sum = sum + i;
```

while loop:

```
long sum = 0;
int i=0;
```

```
while (i <= 1000) {
  sum += i++;
}
```

do-while loop:

```
do-while loop:
long sum = 0;
int i = 0;
do {
  sum += i++;
}
while (i <= 1000);
```

Hide Answer Read Answer

▼5.7.7

Count the number of iterations in the following loops.

```
(a)
int count = 0;
while (count < n) {
  count++;
}

(b)
for (int count = 0;
  count <= n; count++) {
}

(c)
int count = 5;
while (count < n) {
  count++;
}

(d)
int count = 5;
while (count < n) {
  count = count + 3;
}
```

```
(a)
n times

(b)
n+1 times

(b)
n-5 times

(d)
The ceiling of (n-5)/3 times
```

Hide Answer Read Answer

Section 5.8

▼5.8.1

Can you convert a for loop to a while loop? List the advantages of using for loops.

Yes. When using a while loop, programmers often forget to adjust the control variable such as i++. Using for loop can avoid this error.

Hide Answer Read Answer

▼5.8.2

Can you always convert a while loop into a for loop? Convert the following while loop into a for loop.

```
int i = 1;
int sum = 0;
while (sum < 10000) {
  sum = sum + i;
  i++;
}
```

Yes.

```
for (int i=1; sum < 10000; i++)
   sum = sum + i;
```

Hide Answer Read Answer

▼5.8.3

Identify and fix the errors in the following code:

```
 1  public class Test {
 2    public void main(String[] args) {
 3      for (int i = 0; i < 10; i++);
 4        sum += i;
 5
 6      if (i < j);
 7        System.out.println(i)
 8      else
 9        System.out.println(j);
10
11      while (j < 10);
12      {
13        j++;
14      }
15
16      do {
17        j++;
18      } while (j < 10)
19    }
20  }
```

Line 2: missing static.
Line 3: The semicolon (;) at the end of the for loop heading should be removed.

Line 4: sum not defined.
Line 6: the semicolon (;) at the end of the if statement should be removed.
Line 6: j not defined.
Line 7: Missing a semicolon for the first println statement.
Line 11: The semicolon (;) at the end of the while heading should be removed.
Line 18: Missing a semicolon at the end of the do-while loop.

Hide Answer Read Answer

▼5.8.4

What is wrong with the following programs?

```
(a)
 1 public class ShowErrors {
 2   public static void main(String[] args) {
 3     int i = 0;
 4     do {
 5       System.out.println(i + 4);
 6       i++;
 7     }
 8     while (i < 10)
 8   }
 9 }

 (b)
 1 public class ShowErrors {
 2   public static void main(String[] args) {
 3     for (int i = 0; i < 10; i++);
 4       System.out.println(i + 4);
 5   }
 6 }
```

(a) Line 8: Missing ; at the end of this line.
(b) Line 3: The ; at the end of for loop should be removed.

```
for (int i = 0; i < 10; i++);
```

Hide Answer Read Answer

Section 5.9

▼5.9.1

How many times is the println statement executed?

```
for (int i = 0; i < 10; i++)
  for (int j = 0; j < i; j++)
    System.out.println(i * j)
```

When i is 0, the println statement is not executed.
When i is 1, the println statement is executed once.
When i is 2, the println statement is executed two times.
When i is 3, the println statement is executed three times.
When i is 9, the println statement is executed nine times.
So, the total is 0 + 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45

Hide Answer Read Answer

▼5.9.2

Show the output of the following programs. (Hint: Draw a table and list the variables in the columns to trace these programs.)

(a)

```
public class Test {
  public static void main(String[] args) {
    for (int i = 1; i < 5; i++) {
      int j = 0;
      while (j < i) {
        System.out.print(j + " ");
        j++;
      }
    }
  }
}
```

(b)

```
public class Test {
  public static void main(String[] args) {
    int i = 0;
    while (i < 5) {
      for (int j = i; j > 1; j--)
        System.out.print(j + " ");
      System.out.println("****");
      i++;
    }
  }
}
```

(c)

```
public class Test {
  public static void main(String[] args) {
    int i = 5;
    while (i >= 1) {
      int num = 1;
      for (int j = 1; j <= i; j++) {
        System.out.print(num + "xxx");
        num *= 2;
      }

      System.out.println();
      i--;
    }
  }
}
```

(d)

```
public class Test {
  public static void main(String[] args) {
    int i = 1;
    do {
      int num = 1;
```

```
      for (int j = 1; j <= i; j++) {
        System.out.print(num + "G");
        num += 2;
      }

      System.out.println();
      i++;
    } while (i <= 5);
  }
}
```

Tip for tracing programs: Draw a table to see how variables change in the program. Consider (a) for example.

```
i       j           output
1       0           0
1       1
2       0           0
2       1           1
2       2
3       0           0
3       1           1
3       2           2
3       3
4       0           0
4       1           1
4       2           2
4       3           3
4       4
```

```
(a).
0 0 1 0 1 2 0 1 2 3

(b).
****
****
2 ****
3 2 ****
4 3 2 ****
(c).
1xxx2xxx4xxx8xxx16xxx
1xxx2xxx4xxx8xxx1xxx2xxx4xxx
1xxx2xxx
1xxx
(d).
1G
1G3G
1G3G5G
1G3G5G7G
1G3G5G7G9G
```

Hide Answer Read Answer

▼5.11.1

Will the program work if n1 and n2 are replaced by n1 / 2 and n2 / 2 in line 17 in Listing 5.9?

No. Try n1 = 3 and n2 =3.

Hide Answer Read Answer

▼5.11.2

In Listing 5.11, why is it wrong if you change the code (char)(hexValue + '0') to hexValue + '0' in line 21?

Possible loss of precision for int to char.

Hide Answer Read Answer

▼5.11.3

In Listing 5.11, how many times the loop body is executed for a decimal number 245 and how many times the loop body is executed for a decimal number 3245?

The number of digits in the hex number determines the number of times the loop is executed. 245 is F5 in hex. So the loop body is executed 2 times. 3245 is CAD in hex. So, So the loop body is executed 3 times.

Hide Answer Read Answer

▼5.11.4

What is the hex number after E? What is the hex number after F?

The hex number after E is F. The hex number after F is 10.

Hide Answer Read Answer

▼5.11.5

What will be the output if the input is 0? Revise line 27 in Listing 5.11 so that the program displays hex number 0 if the input decimal is 0.

If the input is 0, the value in variable hex will be an empty string. This is an error. To fix it, use

```
  System.out.println("The hex number is " + (hex.length() == 0 ? "0" : hex));
```

or

```
if (hex.length() == 0)
  System.out.println("The hex number is 0");
else
  System.out.println("The hex number is " + hex);
```

Hide Answer Read Answer

Section 5.12

▼5.12.1

What is the keyword break for? What is the keyword continue for? Will the following programs terminate? If so, give the output.

```
(a)
int balance = 10;
while (true) {
  if (balance < 9)
    break;
  balance = balance - 9;
}

System.out.println("Balance is "
  + balance);

(b)
int balance = 10;
while (true) {
  if (balance < 9)
    continue;
  balance = balance - 9;
}

System.out.println("Balance is "
  + balance);
```

The keyword break is used to exit the current loop. The program in (A) will terminate. The output is Balance is 1.
The keyword continue causes the rest of the loop body to be skipped for the current iteration.
The while loop will not terminate in (B).

Hide Answer Read Answer

▼5.12.2

The for loop in (a) is converted into the while loop in (b). What is wrong? Correct it.

```
(a)
int sum = 0;
for (int i = 0; i < 4; i++) {
  if (i % 3 == 0) continue;
  sum += i;
}

(b)
int i = 0, sum = 0;
while (i < 4) {
  if (i % 3 == 0) continue;
  sum += i;
  i++;
}
```

If a continue statement is executed inside a for loop, the rest of the iteration is skipped, then the action-after-each-iteration is performed and the loop-continuation-condition is checked. If a continue statement is executed inside a while loop, the rest of the iteration is skipped, then the loop-continuation-condition is checked. Here is the fix:

```
int i = 0, sum = 0;
while (i < 4) {
  if (i % 3 == 0) {
```

```
    i++;
    continue;
  }
  sum += i;
  i++;
}
```

Hide Answer Read Answer

▼5.12.3

Rewrite the programs TestBreak and TestContinue in Listings 5.12 and 5.13 without using break and continue.

```
class TestBreak {
  public static void main(string[]args) {
    int sum = 0;
    int number = 0;

    while (number < 20 && sum < 100) {
      number++;
      sum += number;
    }

    System.out.println("The sum is " + sum);
  }
}

class TestContinue {
  public static void main(String[] args) {
    int sum = 0;
    int number = 0;

    while (number < 20) {
      number++;
      if (number != 10 && number != 11)
        sum += number;
    }

    System.out.println("The sum is " + sum);
  }
}
```

Hide Answer Read Answer

▼5.12.4

After the break statement in (a) is executed in the following loop, which statement is executed? Show the output. After the continue statement in (b) is executed in the following loop, which statement is executed? Show the output.

```
(a)
for (int i = 1; i < 4; i++) {
  for (int j = 1; j < 4; j++) {
    if (i * j > 2)
      break;
```

```
      System.out.println(i * j);
    }

    System.out.println(i);
  }

  (b)
  for (int i = 1; i < 4; i++) {
    for (int j = 1; j < 4; j++) {
      if (i * j > 2)
        continue;

      System.out.println(i * j);
    }

    System.out.println(i);
  }
```

(a) After the break statement is executed, the last println statement is executed. The output of this fragment is

```
1
2
1
2
2
3
```

(b) After the continue statement is executed, the statement

```
  j++
```

will be executed. The output of this fragment is

```
1
2
1
2
2
3
```

Hide Answer Read Answer

Section 5.13

▼5.13.1

What happens to the program if (low < high) in line 20 is changed to (low <= high)?

The program will work just fine, but there is no need to check for the case when low is equal to high. Obviously the change is unnecessary and not good.

Hide Answer Read Answer

Section 5.14

▼5.14.1

Simplify the code in lines 27-32 using a conditional operator.

```
System.out.print(count % NUMBER_OF_PRIMES_PER_LINE == 0 ?
  number + "\n" : number + " ");
```

Hide Answer Read Answer