

# CSE 101 Final

Olcay Taner YILDIZ, Esma MERAL

1. Which of the following can not be a variable name?

- a) lsum
- b) s1
- c) integer
- d) sum1
- e) toplam

2. Given  $T$  as an integer, how do you check if  $T$  is **not** between 0 and 10?

- a) **if** (not (0 < T < 10))
- b) **if** (T > 0 || T < 10)
- c) **if** (T < 0 || T > 10)
- d) **if** (T < 0 && T > 10)
- e) **if** (T > 0 && T < 10)

3. What is the value of integer variable x after the execution of the following line?

```
int x = 3 / 2 + 3 / 2 + 6 / 4 + 6 / 4;
```

- a) 0
- b) 8
- c) 2
- d) 4
- e) 6

4. Which of the following expressions result in value 1?

- a) 2 % 1
- b) 15 % 4
- c) 25 % 5
- d) 38 % 6
- e) 16 % 3

5. Which of the following code fragments prints "AA" if the grade is larger than 70, "BB" if the grade is less than 70 and larger than 50, "F" otherwise.

- a)
- ```

if (grade > 70){
    System.out.println("AA");
}
if (grade > 50){
    System.out.println("BB");
}

```

```

if (grade <= 50){
    System.out.println("F");
}

```

b)

```

if (grade > 70){
    System.out.println("AA");
}
if (grade > 50){
    System.out.println("BB");
} else {
    System.out.println("F");
}

```

c)

```

if (grade > 70){
    System.out.println("AA");
} else {
    if (grade < 50){
        System.out.println("BB");
    } else {
        System.out.println("F");
    }
}

```

d)

```

if (grade > 70){
    System.out.println("AA");
} else {
    if (grade > 50){
        System.out.println("BB");
    } else {
        System.out.println("F");
    }
}

```

e)

```

if (grade > 70){
    System.out.println("AA");
}
if (grade > 50){
    System.out.println("BB");
}
if (grade <= 50 && grade < 70){
    System.out.println("F");
}

```

6. Suppose that  $x$  is a user given 3 digit integer (e.g. 635), which of the following prints its hundreds digit (e.g. 6)?

- a) `System.out.println(x * 100);`
- b) `System.out.println(x / 100);`
- c) `System.out.println((x / 100) / 100);`
- d) `System.out.println((x % 100) % 100);`
- e) `System.out.println(x % 100);`

7. Which of the following defines a variable of type boolean?

- a) `boolean b1 = 0;`
- b) `boolean b2 = 'false';`
- c) `boolean b3 = false;`
- d) `boolean b4 = Boolean.false();`
- e) `boolean b5 = no;`

8. Which of the following codes is the same as the following code?

```
if (x == 4 || x == 9)
    System.out.println("Done");
else
    if (x >= 1 && x <= 3)
        System.out.println("Not_done");
    else
        System.out.println(" Invalid ");
```

a)

```
switch (x){
    case 4:
    case 9: System.out.println("Done");
    case 1:
    case 2:
    case 3: System.out.println("Not_done");
}
```

b)

```
switch (x){
    case 4:
    case 9: System.out.println("Done");
        break;
    case 1:
    case 2:
    case 3: System.out.println("Not_done");
        break;
    default: System.out.println(" Invalid ");
}
```

c)

```
switch (x){
    case 4:
    case 9: System.out.println("Done");
        break;
    case 1:
    case 2:
    case 3: System.out.println("Not_done");
```

```
        break;
}
```

d)

```
switch (x){
    case 4:
    case 9: System.out.println("Done");
    case 1:
    case 2:
    case 3: System.out.println("Not_done");
        break;
}
```

e)

```
switch (x){
    case 4:
    case 9: System.out.println("Done");
        break;
    case 1:
    case 2:
    case 3: System.out.println("Not_done");
    default: System.out.println(" Invalid ");
}
```

9. What is the output of the following code, if the user enters 0?

```
int choice = input.nextInt();
int j = 0;
switch (choice){
    case 0:
        j += 1;
    case 1:
        j += 2;
    case 2:
        j += 3;
    case 3:
        j += 4;
    default:
        j += 5;
}
System.out.println(j);
```

- a) 1
- b) 3
- c) 6
- d) 10
- e) 15

10. Which of the following is the correct way of defining an array?

- a) `int [] myList = {'1', '2', '3'};`
- b) `int [] myList = (5, 8, 2);`
- c) `int myList [] [] = {4, 9, 7, 0};`
- d) `int [] myList = {4, 3, 7};`
- e) `int myList = {2, 9, 0};`

11. What is the output of the following code?

```
int sum = 0;
int i = 0;
while (i < 5) {
    sum = sum + i;
    i++;
}
System.out.println(i + " " + sum);
```

- a) 5 10
- b) 6 10
- c) 4 10
- d) 5 11
- e) 6 11

12. Which of the following is the correct definition of a method?

- 1) public static void method() {}
  - 2) public static void method(void) {}
  - 3) public static method() {}
  - 4) public static method(void) {}
- a) 1, 2
  - b) 1
  - c) 1, 2, 3
  - d) 2, 4
  - e) 1, 2, 3, 4

13. What will be the output of the following program?

```
int sum = 0;
for (int i = 1; i <= 5; i++){
    for (int j = 1; j <= 5; j++){
        for (int k = 1; k <= 5; k++){
            sum++;
        }
    }
}
System.out.println(sum);
```

- a) 55
- b) 35
- c) 210
- d) 50
- e) 125

14. What is the content of the array list after executing the following code?

```
int [] list = {10, 20, 30, 40};
int tmp = list [ list .length - 1];
for (int i = 1; i < list .length; i++) {
    list [i] = list [i - 1];
}
list [0] = tmp;
```

- a) {10, 20, 30, 40}
- b) {40, 10, 10, 10}
- c) {40, 40, 40, 10}
- d) {10, 40, 40, 40}
- e) {10, 10, 10, 10}

15. What will be the output of the following program?

```
int sum = 0;
for (int i = 1; i <= 10; i++){
    for (int j = i; j <= 10 - i; j++){
        sum++;
    }
}
System.out.println(sum);
```

- a) 55
- b) 110
- c) 15
- d) 25
- e) 10

16. Consider the following for loop

```
for (A; B; C){
}
```

How do you convert this for loop into a while loop?

a)

```
A;
while (B){
}
C;
```

b)

```
B;
while (A){
    C;
}
```

c)

```
A;
while (C){
    B;
}
```

d)

```
A;
B;
while (C){
}
```

e)

```
A;
while (B){
    C;
}
```

17. What will be the output of the following program?

```
for (int i = 1; i <= 3; i++){
    for (int j = 1; j <= 3; j++){
        System.out.print("*");
    }
    System.out.print("\n");
    System.out.println ();
}
```

- a) \*\*\*\*  
 \*\*\*\*  
 \*\*\*\*
- b) \*\*\*  
 \*\*\*  
 \*\*\*
- c) \*\*\*  
 \*\*  
 \*
- d) \*  
 \*\*  
 \*\*\*
- e) \*\*\*\*\*

18. Let say you have the following function defined:

```
public static void printStar (int n){
    for (int i = 1; i <= n; i++){
        System.out.print("*")
    }
    System.out.println ();
}
```

Which of the following code fragments produce the following output?

```
*
**
***
****
***
**
*
```

- a) 

```
for (int i = 1; i <= 4; i++)
    printStar (1);
```
- b) 

```
for (int i = 1; i <= 4; i++)
    printStar (i);
```
- c) 

```
for (int i = 1; i <= 4; i++)
    printStar (1);
for (int i = 1; i <= 3; i++)
    printStar (1);
```

d) 

```
for (int i = 1; i <= 4; i++)
    printStar (i);
for (int i = 1; i <= 3; i++)
    printStar (i);
```

e) 

```
for (int i = 1; i <= 4; i++)
    printStar (i);
for (int i = 3; i >= 1; i--)
    printStar (i);
```

19. Let say you have the functions factorial and power

```
public static int factorial (int N)
public static int power(int x, int y)
```

which calculate  $N!$  and  $x^y$  respectively. Which of the following code fragments then calculate the following sum:

$$\frac{2^1}{1!} + \frac{2^2}{2!} + \dots + \frac{2^N}{N!}$$

a) 

```
double sum = power(2, N) / factorial(N);
```

b) 

```
double sum = 0;
for (int i = 1; i <= N; i++){
    sum += power(2, N) / ( factorial (i) + 0.0);
}
```

c) 

```
double sum = 0;
for (int i = 1; i <= N; i++){
    sum += power(2, i) / ( factorial (i) + 0.0);
}
```

d) 

```
double sum = 0;
for (int i = 1; i <= N; i++){
    sum += power(2, i) / factorial (i);
}
```

e) 

```
double sum = 0;
for (int i = 1; i <= N; i++){
    sum += power(2, i) / ( factorial (N) + 0.0);
}
```

20. Let say you have the function

```
public static int sumOfDivisorsExceptItself(int N)
```

which returns the sum of divisors of N except itself. Which of the following code fragments prints all perfect numbers until 1000, where a number is perfect if it is equal to the sum of its divisors?

a) 

```
System.out.println(sumOfDivisorsExceptItself(100));
```

b) 

```
for (int i = 2; i <= 1000; i++){
    System.out.println(sumOfDivisorsExceptItself(i));
}
```

c) 

```
for (int i = 2; i <= 1000; i++){
    if (sumOfDivisorsExceptItself(i)){
        System.out.println(i);
    }
}
```

d) 

```
for (int i = 2; i <= 1000; i++){
    System.out.println(i);
}
```

e) 

```
for (int i = 2; i <= 1000; i++){
    if (sumOfDivisorsExceptItself(i) == i){
        System.out.println(i);
    }
}
```

21. Which of the following functions creates and returns an array of 100 integers, where each element of the array is 1.

a) 

```
public static int createArray(){
    return 1;
}
```

b) 

```
public static int createArray(){
    int [] array;
    for (int i = 0; i < 100; i++)
        array[i] = 1;
    return array;
}
```

c) 

```
public static int [] createArray(){
    int [] array;
    for (int i = 0; i < 100; i++)
        array[i] = 1;
    return array;
}
```

d) 

```
public static int [] createArray(){
    int [] array;
    array = new int[100];
    for (int i = 0; i < 100; i++)
        array[i] = 1;
    return array;
}
```

e) 

```
public static int createArray(){
    int [] array;
    array = new int[100];
    for (int i = 0; i < 100; i++)
        array[i] = 1;
    return array;
}
```

22. Which of the following functions takes a parameter of an array, and returns the sum of elements of that array?

a) 

```
public static int sumArray(int[] array){
    sum(array);
}
```

b) 

```
public static int sumArray(int[] array){
    int sum = 0;
    for (int i = 0; i < array.length; i++)
        sum += array[i];
}
```

c) 

```
public static int [] sumArray(int[] array){
    int sum = 0;
    for (int i = 0; i < array.length; i++)
        sum += array[i];
    return sum;
}
```

d) 

```
public static int sumArray(int[] array){
    int sum = 0;
    for (int i = 0; i < array.length; i++){
        array[i] = input.nextInt();
        sum += array[i];
    }
    return sum;
}
```

e) 

```
public static int sumArray(int[] array){
    int sum = 0;
    for (int i = 0; i < array.length; i++)
        sum += array[i];
    return sum;
}
```

23. Which of the following can be used to get the number of elements of an array?

- a) arr.length
- b) arr.length-1
- c) arr.size
- d) arr.length()
- e) arr.size()

24. Which of the following functions takes a parameter of an array and a number N, and returns the number of elements of that array that are smaller than N?

a)

```
public static int smaller(int[] array, int N){
    N = input.nextInt();
    int count = 0;
    for (int i = 0; i < array.length; i++){
        if (array[i] < N){
            count++;
        }
    }
    return count;
}
```

b)

```
public static int smaller(int[] array, int N){
    int count;
    for (int i = 0; i < array.length; i++){
        if (array[i] < N){
            count++;
        }
    }
    return count;
}
```

c)

```
public static int smaller(int[] array, int N){
    int count = 0;
    for (int i = 0; i < array.length; i++){
        if (array[i] < N){
            count++;
        }
    }
    return count;
}
```

d)

```
public static int smaller(int[] array, int N){
    int count = 0;
    for (int i = 0; i < array.length; i++){
        if (array[i] < N){
            count++;
        } else {
            return count;
        }
    }
    return count;
}
```

e)

```
public static int smaller(int[] array){
    int count = 0;
    for (int i = 0; i < array.length; i++){
        if (array[i] < N){
            count++;
        }
    }
    return count;
}
```

25. Which of the following functions takes a parameter of an array and a number N, and returns true if the array contains N and false otherwise?

a)

```
public static boolean contains(int[] array, int N){
    array.contains(N);
}
```

b)

```
public static boolean contains(int[] array, int N){
    for (int i = 0; i < array.length; i++){
        if (array[i] == N){
            return true;
        } else
            return false;
    }
}
```

c)

```
public static boolean contains(int[] array, int N){
    for (int i = 0; i < array.length; i++){
        if (array[i] == N){
            return true;
        }
    }
    return false;
}
```

d)

```
public static boolean contains(int[] array, int N){
    for (int i = 0; i < array.length; i++){
        if (array[i] == N){
            return false;
        } else
            return true;
    }
}
```

e)

```
public static boolean contains(int[] array, int N){
    N = input.nextInt();
    for (int i = 0; i < array.length; i++){
        if (array[i] == N){
            return true;
        }
    }
    return false;
}
```