Algorithms and Programming Questions

DIRECTIONS: Each of the questions or incomplete statements below is followed by four suggested answers or completions. Select the one that is best in each case.

1. What will the following algorithm display?

```
a ← 13
b ← 17
a ← a+1
c ← a/7
DISPLAY c
DISPLAY a
DISPLAY b
```

(A) 2 14 17
(B) 13 17 5
(C) 2 12 2
(D) 14 17 2

2. What will the following algorithm display?

```
a ← 13
a ← 17
a ← a+1
DISPLAY a
```

(A) 13
(B) 17
(C) 18
(D) 19

3. What will the following algorithm display?

```
a ← 11
a ← a + 35
a ← a + 6
b ← a MOD 2
b ← a * b
DISPLAY b
```

(A) 0
(B) 42
(C) 84
(D) 126

4. What will the following algorithm display?

```
a ← "Milk"
a ← a + "Cookies Soda"
a ← a + "Chips"
b ← a + "put them in a bag so you know they stay crisp"
```

(A) Milk
(B) Milk Cookies Soda
(C) put them in a bag and they stay crisp
(D) Milk Cookies Soda Chips put them in a bag so you know they stay crisp

5. What is the value displayed after the program is run?

```
a ← 8
b ← 3
c ← 2
a ← b * c
c ← c + 4
DISPLAY "a"
DISPLAY c
```

(A) a 8
(B) a 6
(C) 8 6
(D) 16 4

6. What is the value displayed after the program is run?

```
a ← 8
b ← 3
c ← 2
a ← b * c
c ← c + 4
DISPLAY "a"
DISPLAY "c"
```

(A) a 8
(B) a c
(C) 8 6
(D) 16 4

7. What will the following algorithm display?

```
a ← 26 MOD 2
DISPLAY a
```

(A) 26
(B) 13
(C) 1
(D) 0

8. What will the following algorithm display?

```
a ← 5 MOD 2
DISPLAY a
```

(A) 26
(B) 13
(C) 1
(D) 0

9. What will the following algorithm display?

```
a ← 8 MOD 26
DISPLAY a
```

(A) 26
(B) 13
(C) 8
(D) 1

10. What will the following algorithm display?

```
a ← 13 MOD 26
DISPLAY a
```

(A) 26
(B) 13
(C) 8
(D) 0

11. What will the following algorithm display?

```
a ← 26 MOD 13
DISPLAY a
```

(A) 26
(B) 13
(C) 2
(D) 0

12. What will the following algorithm display?

```
a ← 26 MOD 3
b ← 7 MOD a
DISPLAY b
```

(A) 26
(B) 13
(C) 1
(D) 0

13. What will the following algorithm display?

```
a ← 2 MOD 3
b ← 7 MOD a
DISPLAY b
```

(A) 26
(B) 13
(C) 1
(D) 0

14. What will the following code segment display?

```
a ← 13
b ← 5
temp ← a
a ← b
b ← temp
DISPLAY a
DISPLAY b
```

(A) 13 5
(B) 5 13
(C) a b
(D) 13 13

15. If this statement is executed many times, about what percentage of times does it display `true`?

```
DISPLAY RANDOM 1,10 = 6
```

(A) 9%
(B) 10%
(C) 60%
(D) 100%

16. If this statement is executed many times, about what percentage of times does it display `true`?

```
DISPLAY RANDOM 5,9 = 6
```

(A) 10%
(B) 20%
(C) 60%
(D) 100%

17. If this statement is executed many times, about what percentage of times does it display `true`?

```
DISPLAY RANDOM 5,9 = 4
```

(A) 0%
(B) 20%
(C) 60%
(D) 100%

18. If this statement is executed many times, about what percentage of times does it display `true`?

```
DISPLAY RANDOM 5,9 ≤ 9
```

(A) 0%
(B) 20%
(C) 60%
(D) 100%

19. If this statement is executed many times, about what percentage of times does it display `true`?

```
DISPLAY RANDOM 5,9 > 9
```

(A) 0%
(B) 20%
(C) 60%
(D) 100%

20. If this statement is executed many times, about what percentage of times does it display `true`?

```
DISPLAY [RANDOM [1,10] ≤ 6]
```

(A) 9%
(B) 10%
(C) 60%
(D) 100%

21. If this statement is executed many times, about what percentage of times does it display `true`?

```
x ← [RANDOM [1, 5]]
[DISPLAY [x = 5] OR [x = 9]]
```

(A) 20%
(B) 40%
(C) 60%
(D) 100%

22. If this statement is executed many times, about what percentage of times does it display `true`?

```
DISPLAY [RANDOM [1,5] = 6] AND [RANDOM [1,5] ≤ 5]
```

(A) 0%
(B) 40%
(C) 60%
(D) 100%

23. What is the percentage of times that this algorithm displays `true`?

```
x ← [RANDOM [1, 5]]
[DISPLAY [x = 6] OR [x ≤ 5]]
```

(A) 0%
(B) 40%
(C) 60%
(D) 100%

24. The algorithm below displays `true` 60% of the time.

```
x ← [RANDOM [5,14]]
IF [missing condition]
    [DISPLAY "TRUE"]
ELSE
    [DISPLAY "FALSE"]
```

What can replace the missing condition so the code segment works as intended?

(A) $x < 10$
(B) $x \leq 10$
(C) $x > 10$
(D) $x \geq 10$

25. What is the value displayed after the algorithm is run?

```
x ← 10
REPEAT UNTIL [x ≥ 10]
    [x ← (x + 3)]
[DISPLAY (x)]
```

(A) 0
(B) 1
(C) 10
(D) 13

26. What is the value displayed after the algorithm is run?

```
x ← 0
REPEAT UNTIL [x < 10]
    [x ← (x + 1)]
[DISPLAY (x)]
```

(A) 0
(B) 1
(C) 10
(D) 12

27. What is the value displayed after the algorithm is run?

```
x ← 0
REPEAT UNTIL x ≥ 10
    x ← x + 3
DISPLAY x
```

(A) 0
(B) 1
(C) 9
(D) 12

28. What is the value displayed after the algorithm is run?

```
x ← 0
REPEAT UNTIL x ≤ 10
    x ← x + 3
    x ← 0
DISPLAY x
```

(A) 0
(B) 1
(C) 10
(D) Nothing is displayed due to an infinite loop.

29. What is the value displayed after the algorithm is run?

```
x ← 10
REPEAT UNTIL x ≤ 10
    x ← x + 3
DISPLAY x
```

(A) 0
(B) 1
(C) 10
(D) 12

30. What is the value displayed after the algorithm is run?

```
x ← 9
REPEAT UNTIL x ≤ 10
    x ← x + 3
DISPLAY x
```

(A) 0
(B) 9
(C) 10
(D) 12

31. What is the value displayed after the algorithm is run?

```
x ← 0
REPEAT 3 TIMES
    x ← x + 2
    DISPLAY x
```

(A) 2 4 6
(B) 0 2 4 6
(C) 6
(D) 8

32. What is the value displayed after the algorithm is run?

```
x ← 0
REPEAT 3 TIMES
    x ← x + 2
DISPLAY x
```

(A) 2 4 6
(B) 0 2 4 6
(C) 6
(D) 8

33. What is the value displayed after the algorithm is run?

```
x ← 1
REPEAT 3 TIMES
    x ← x + 2
    IF x MOD 2 = 0
        DISPLAY "EVEN"
```

(A) Nothing is displayed.
(B) Even Even Even
(C) ODD ODD ODD
(D) 2 4 6

34. What is the value displayed after the algorithm is run?

```
x ← 1
REPEAT 3 TIMES
    x ← x + 2
    IF x MOD 2 = 0
        DISPLAY EVEN
    ELSE
        DISPLAY ODD
```

(A) Nothing is displayed.
(B) Even Even Even
(C) ODD ODD ODD
(D) 2 4 6

35. What is the value displayed after the algorithm is run?

```
x ← 0
REPEAT 3 TIMES
    REPEAT 2 TIMES
        x ← x + 1
DISPLAY x
```

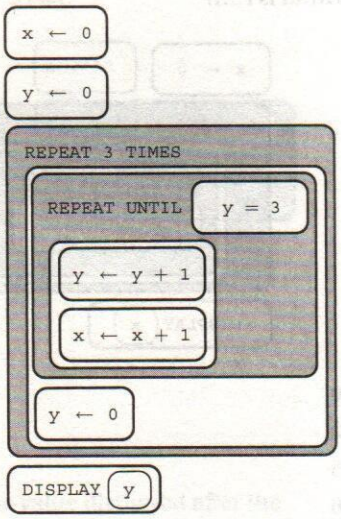
(A) 0
(B) 2
(C) 5
(D) 6

36. What is displayed after the algorithm is run?

```
x ← 0
y ← 0
REPEAT 3 TIMES
    REPEAT UNTIL y = 3
        y ← y + 1
        x ← x + 1
    y ← 0
DISPLAY x
```

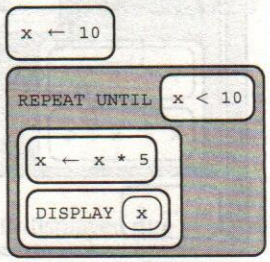
(A) 0
(B) 3
(C) 9
(D) 12

37. What is displayed after the algorithm is run?



(A) 0
(B) 3
(C) 9
(D) 12

38. How many numbers will the following algorithm display?



(A) 0
(B) 10
(C) 100
(D) An infinite amount

39. What will be printed when this algorithm is run?

```
x ← 4
y ← 8
z ← x

IF(x < 2)
{
  DISPLAY("Pineapple")
}
IF(z < y)
{
  DISPLAY("Kumquat")
}
ELSE(x < z)
{
  DISPLAY("Star Fruit")
}
```

(A) Pineapple
(B) Kumquat
(C) Star Fruit
(D) Pineapple Kumquat Star Fruit

40. How many times will "fish" be displayed?

```
a ← 1
REPEAT 5 TIMES
{
  IF(a > 3)
  {
    DISPLAY("fish")
  }
  a = a + 1
}
```

(A) 4
(B) 3
(C) 2
(D) 1

41. What is displayed at the end of the algorithm?

```
IF(TRUE = TRUE AND 7 < 6)
{
  DISPLAY("Elephant")
}
IF(8 > 4 OR TRUE = FALSE)
{
  DISPLAY("Alligator")
}
IF(3=3)
{
  DISPLAY("Ostrich")
}
```

(A) Elephant
(B) Alligator Ostrich
(C) Alligator
(D) Elephant Alligator Ostrich

42. A user inputs 5 for the value of a. What is a possible value of c?

$a \leftarrow$ INPUT( )
$b \leftarrow$ RANDOM(a, 6)
$c \leftarrow b - 6$

(A) −3
(B) −1
(C) 1
(D) 3

43. The following is a truth table for all possible values of A and B.

A	B	<missing condition>
T	T	T
T	F	F
F	T	F
F	F	F

Which of the following can replace the missing condition?

(A) (A AND B)
(B) (A OR B)
(C) (A OR NOT(A))
(D) (A AND NOT(A))

44. The following is a truth table for all possible values of A.

A	B	<missing condition>
T	T	T
T	F	T
F	T	T
F	F	T

Which of the following can replace the missing condition?

(A) (A AND B)
(B) (A OR B)
(C) (A OR NOT(A))
(D) (A AND NOT(A))

45. Which of the following statements describes the major building blocks of algorithms?

I. Sequencing—statements execute in a given order
II. Selection—Boolean conditions determine an algorithm's path
III. Iteration—the repetition of parts of an algorithm

(A) I only
(B) I and II only
(C) II and III only
(D) I, II, and III

46. A programmer is writing code to display the difference of the squares of two user-inputted numbers (i.e., $a^2 - b^2$). The following lines represent parts of the code. What order should they be placed in?

```
1. DISPLAY(a - b)
2. b ← INPUT()
3. a ← a * a
4. a ← INPUT()
5. b ← b * b
```

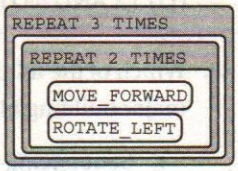
(A) 1, 2, 3, 4, 5
(B) 1, 2, 4, 5, 3
(C) 2, 4, 3, 5, 1
(D) 4, 2, 5, 1, 3

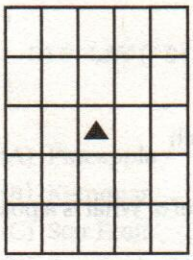
47. An algorithm compares the user-inputted number picked to the randomly selected number drawing and calls the method `victoryJingle()` if the two are the same. What should replace `<Missing Code>` in the following algorithm?

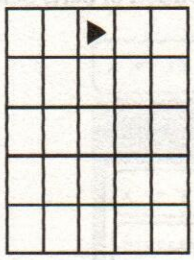
```
<Missing Code>
{
  victoryJingle()
}
```

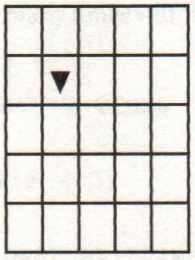
(A) `IF(picked AND drawing)`
(B) `IF(picked = drawing)`
(C) `IF(picked ≠ drawing)`
(D) `IF(picked NOT drawing)`

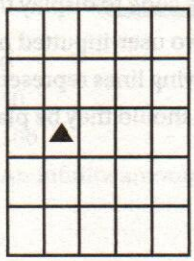
48. Using the program below, where will the robot land after the code segment executes?

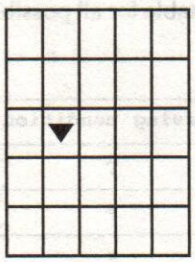




(A) 

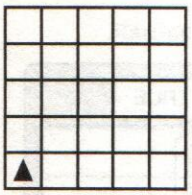
(C) 

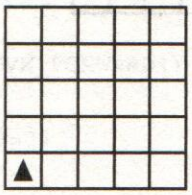
(B) 

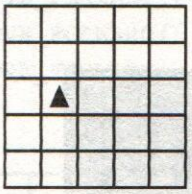
(D) 

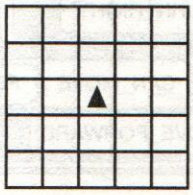
49. Using the program below, which of the following is a possible landing spot after the code segment executes?

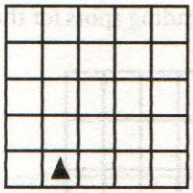
```
Line 1: move ← INPUT()
Line 2: REPEAT move TIMES
Line 3: {
Line 4: MOVE_FORWARD()
Line 5: ROTATE_RIGHT()
Line 6: }
```



(A) 

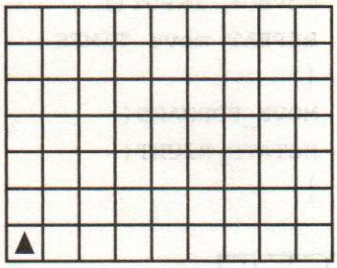
(B) 

(C) 

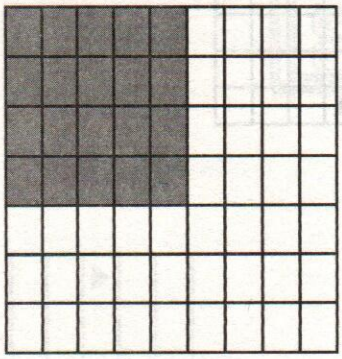
(D) 

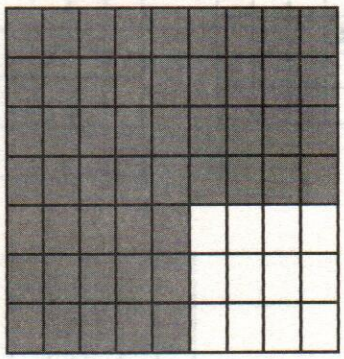
50. Using the program below, what are the possible locations the robot could visit?

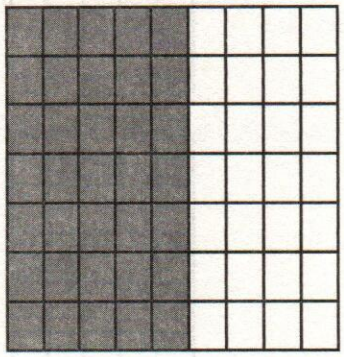
```
i ← FALSE
REPEAT UNTIL [ i = TRUE ]
    n ← [ RANDOM [ 1, 4 ] ]
    REPEAT n TIMES
        TURN_RIGHT
    IF [ CAN_MOVE ( FORWARD ) ]
        MOVE_FORWARD
```

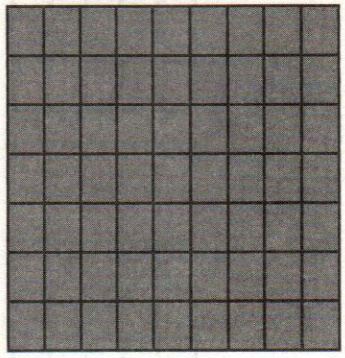


What are the possible landing spots for the robot?

(A) 

(C) 

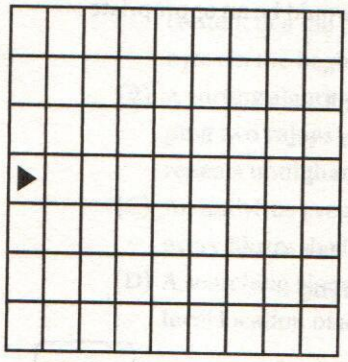
(B) 

(D) 

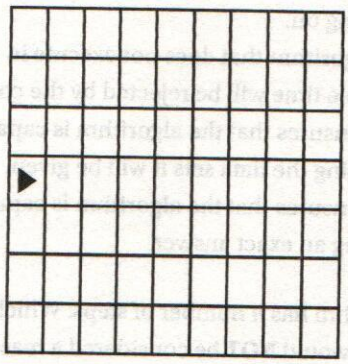
51. Using the program below, where will the robot land after the code segment executes?

```
IF(TRUE = TRUE OR 7 < 6)
{
  MOVE_FORWARD()
  MOVE_FORWARD()
  MOVE_FORWARD()
}
IF(4 > 4 AND TRUE = FALSE)
{
  MOVE_FORWARD()
}
IF(3=3)
{
  MOVE_FORWARD()
  MOVE_FORWARD()
  MOVE_FORWARD()
}
```

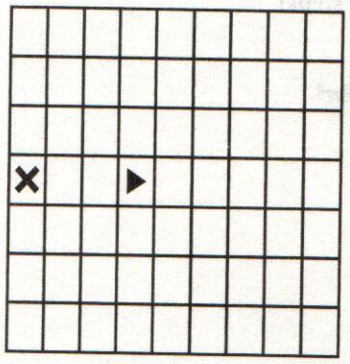
Starting Grid



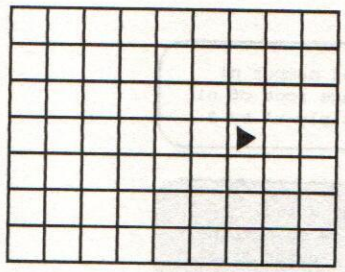
(A)



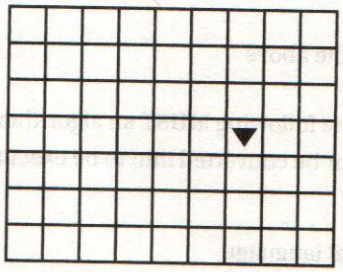
(B)



(C)



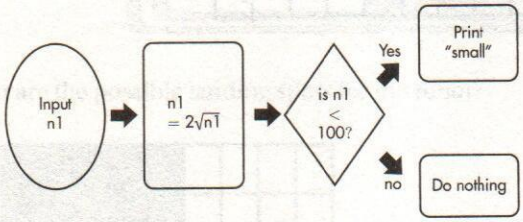
(D)



52. A programmer is creating an algorithm that doubles the square root of an inputted number and prints "small" if the result is less than 100. Which of the following would be an appropriate way to express it?

(A)
```
input number n1
square root of n1
multiply n1 by 2
is n1 < 100?
yes -> print "small"
no -> do nothing
```

(B)



(C)
```
input number n1
Square root of n1
multiply n1 by 2
```

```
is n1 ≥ 100?
yes -> do nothing
no -> print "small"
```

(D) All of the above

53. Which of the following **MUST** an algorithm be written in or be converted into to be executed by a computer?

(A) Natural language
(B) Pseudocode
(C) High-level language
(D) Low-level machine language

54. Suppose that a programmer has created an algorithm using a low-level assembly language. If the algorithm is translated exactly into a higher-level language such as Python, will the solution utilized still work? Why?

(A) Yes, because an exact translation of an algorithm affects only the way that it is read.
(B) Yes, because an algorithm will always work regardless of the language.
(C) No, because the readability of a language affects how complex algorithms can be.
(D) No, because an algorithmic solution can exist only in the language it is written in.

55. Why is it important that algorithms be executed in a reasonable time?

(A) An algorithm that does not execute in a reasonable time will break the computer it is running on.
(B) An algorithm that does not execute in a reasonable time will be rejected by the compiler.
(C) This ensures that the algorithm is capable of handling the data sets it will be given.
(D) This ensures that the algorithm is capable of finding an exact answer.

56. An algorithm has n number of steps. Which of the following would **NOT** be considered a reasonable number of steps?

(A) n
(B) $4n + 8n^2$
(C) $100n^4$
(D) 3^n

57. Why might a programmer decide to make a portion of an algorithm heuristic?
    (A) Heuristics are more accurate, so adding them makes for a stronger algorithm.
    (B) Although heuristics are not as accurate compared to an algorithmic solution, they are much faster to run, which would make the ultimate algorithm more efficient.
    (C) Heuristics are always easier to add into a program.
    (D) Heuristics make an algorithm much harder to copy.

58. Which of the following would be considered a heuristic solution?
    (A) A file-organizing algorithm determines the content of a file based on a certain number of bytes in the beginning of the file.
    (B) A sorting algorithm passes every value, swapping two values where the first is lower. This repeats until there are no more swaps left.
    (C) An antivirus program scans the entirety of every file on the hard drive.
    (D) A searching algorithm determines the bit-level location of a text string in a document.