Capgemini Pseudo Code MCQs (Latest Questions and Answers)

Q1. What will be the output of the following pseudocode?

```
Integer i
Set i = 3
do
print i + 3
i = i - 1
while(i not equals 0)
end while
```

[Note: A do-while loop is a control flow statement that executes a block of code at least once, and then repeatedly executes the given Boolean condition at the end of the block]

A) 6 6 6

B) 6 5 6

C) 5 5 5

D) 6 5 4

Ans: D

Explanation:

In this program, one variable declared as i, and the value initialized as 3. We are moving with do-while(Do while will execute the statement once and then it will check the condition).

Step 1:

It will print i+3, here i value is 3. So i+3 is 6. On the next line, i will be decremented by 1. Then checking the conditions in do-while() i!=0. Here updated i value is 2 (2!=0),so condition is true. The loop continues.

Step 2:

It will print i+3, here the updated i value is 2. So i+3 is 5. On the next line, i will be decremented by 1. Then checking the conditions in do-while() i!=0. Here updated i value is 1 (1!=0),so condition gets true. The loop continues

Step 3:

It will print i+3, here the updated i value is 1. So i+3 is 4. On the next line, i will be decremented by 1. Then check the condition in do-while() i!=0. Here updated i value is 0 (0!=0),so condition gets false. Thus the loop gets terminated!

E.g. code to explain this:

Do while will execute the statement for the first time and then it will check the condition.

```
 int i=3;
      do{
          cout<<i+3; => 3+3 =6 =>2+3 =>5 =>1+3=4
          i=i-1;      => 3-1=2  =>2-1 =>1 =>1-1=0
      }while(i!=0);
```

Q2. What would be the output of the following pseudocode?

```
 Integer a
String str1
Set str1 = "goose"
a = stringLength(str1)
Print (a ^ 1)
```

[Note- string-length(): string-length() function counts the number of characters in a given string and return the integer value.

^ is the bitwise exclusive OR operator that compares each bit of its first operand to the corresponding bit of its equal operand. If one bit is 0 and the other bit is 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0]

A) 0

B) 4

C) 5

D) 3

Ans: B

Explanation:

There are two variables **a** and **str1.** Value initialized for str1 is "goose". On the next line, we are finding the length of str1 that is 5. Finally, printing the output of a bitwise exclusive OR operator with 1. And the answer is 4.

E.g. code to explain this:

```
{
      int a;
      string str1="goose";
      a = str1.length(); => length of String is 5
      cout<<(a^1);      5 xor 1 => 4
}
```

Q3. What would be the output of the following pseudocode?

```
Integer a, b, c
Set a = 8, b = 51, c = 2
c = (a ^ c)^ (a)
b = b mod 4
Print a + b + c
```

[Note- mod finds the remainder after the division of one number by another. For example, the expression “5 mod 2” would evaluate to 1 because 5 divided by 2 leaves a quotient of 2 and a remainder of 1

^ is the bitwise exclusive OR operator that compares each bit of its first operand to the corresponding bit of its equal operand. If one bit is 0 and the other bit is 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0]

A. 13

B. 17

C. 26

D. 16

Ans: A

Explanation:

There are three variables a, b and c declared. Value initialized for a is 8, b is 51 and c is 2.

When we do a bitwise exclusive OR of (8^2), the answer is 10. Again 10 bitwise exclusive OR of a i.e (10 ^ 8) is 2, which will be stored in variable c.

Then taking ***modulo*** operation for b by 4 (b%4) the answer is 3

Finally adding all the updated values of a,b, and c (8+2+3) and the output of Pseudocode is 13.

E.g. code to explain this:

```
int main()
   {
      int a=8,b=51,c=2;
      c = (a ^ c)^ (a); =>(8^2) ^(8) => 10 ^ 8 => 2
```

```
        b = b % 4;     =>3
        cout<<a + b + c; =>8+2+3 => 13
    }
```

Note: ^ is the bitwise exclusive OR

Q4. Consider an array A = {1, 2, 4, 5, 6, 11, 12} and a key which is equal to 10. How many comparisons would be done to find the key element in the array using the binary search?

A) 5

B) 1

C) 2

D) 3

Ans: D

Explanation:

There is an Integer Array A = {1, 2, 4, 5, 6, 11, 12} and the key value is 10.

Binary search applies only to the sorted ordered list.

We know that binary search takes log n base 2 times to search for a particular element.

If there are N elements in the set you have chosen binary search then take log n base 2 times.

1. First, you check how many elements in your array. if an element in your array is greater than **1 then you** go for the next step.
2. Binary search divides the problem into two parts using the mean of the total number of an element which is sorted **mean=(0+n)/2**
3. And compare the searching element which is either greater than the mean or lesser or equal to the mean .then after comparison skips the one part either greater part or small part depends on the result of searching is done if the mean is equal to the searching element. So that problem is divide into **n/2 and** goes until searching is done.

If you apply the recursive equation for the binary search algorithm then

T(n)=T(n/2)+1 to solve the problem

n/2^k=1

taking log on both sides

log n=log (2^k)

log n= k log 2

k=log n base 2.

N=7

K=log 7 base 2 = 3

K=3

So, the answer becomes 3.

Q5. What would be the output of the following pseudocode?

```
 Integer i, j, k
Set k = 8
for(each i from 1 to 1)
        for(each j from the value of i to 1)
            print k+1
        end for
end for
```

A. 2

B. 9

C. 7

D. 8

Ans: B

Explanation:

There are three variables i, j, and k declared. Value initialized for k is 8, In this code, we are moving with nested for loop.

Here I value is 1, for loop will check the condition i<=1 condition gets true. Now, moving with inner for loop j value will be 1 condition gets true j<=1.so, it prints K+1. Then j value will be incremented by 1(2<+1) inner for loop condition gets false.

On the next iteration, i value will be incremented by 1, here the updated i value is 2 (2<=1) condition get false. So the answer is 9.

E.g. code to explain this:

```
int i, j, k;
    k = 8;
```

```
   for(i=1 ;i<=1;i++){ => i=1 True
       for(j=i;j<=1;j++){ => j=1 true
           cout<< k+1;  => print k+1 => 8+1 =>9
   }
}
```

Q6. What will be the output of the following pseudocode?

```
Integer a, b
Set a = 15, b = 7
a = a mod (a - 3)
b = b mod (b – 3)
a = a mod 1
b = b mod 1
Print a + b
```

A) 15

B) 7

C) 2

D) 0

[Note-mod finds the remainder after the division of one number by another. For example, the expression “5 mod 2” leaves a quotient of 2 and a remainder of 1]

Answer: 0

Explanation:

There are two variables a and b declared. Value initialized for a is 15 and b is 7. Taking mod operation for a by 12(a%12) and the answer is 3 will stored in a.

The next mod operation for b is 7 mod (7%4). The answer is 3 will be stored in b.

The next line takes the updated value of a and mods it by 1(3%1). Then the answer becomes 0 will be stored in a.

The next line takes the updated value of b mod by 1 (3%1) then the answer is 0. Finally adding all the updated values of a and b (0+0) and the output of Pseudocode is 0.

Q7. What will be the output of the following pseudocode?

```
Integer a, b, c
Set b = 5, a = 2, c = 2
if(b>a && a>c && c>b)
b = a + 1
```

```
Else
a = b + 1
End if
Print a + b + c
```

[Note-&&: Logical AND - The logical AND operator (&&) returns the Boolean value true(or 1) if both operands--. If (x) gets executed if the value if(), i.e., x is not zero]

A) 2

B) 13

C) 26

D) 5

Ans: B

Explanation:

There are three variables a, b and c declared. Value initialized for a is 2, b is 5 and c is 2.

Checking the condition using if, b >a and a>c and c>b here if conditions get false. Now else part will execute b value will be incremented by 1 and stored in a, Finally adding all the updated values of a, b, and c (6+5+2) and the output of Pseudocode is 13.

Q8. For which of the following applications can you use hashing?

1. To construct a message authentication code.

2. For Timestamping

3. For detecting a cycle in a graph

Choose the correct answer from the options given below.

A) Only 1 and 3

B) Only 2 and 3

C) Only 1

D) Only 1 and 2

Ans: D

Explanation:

Constructing a message authentication code and Timestamping are the real-time applications for hashing.

Q9. Consider an array of float. Calculate the difference between the address of the 1st and 4th element, assuming float occupies 4 bytes of memory.

A) 16

B) 4

C) 12

D) 8

Ans: C

Explanation:

Let's consider the address of elements:

1st element – 1000 - 1003 (4 bytes)

2nd element – 1004 - 1007 (4 bytes)

3rd element – 1008 – 1011 (4 bytes)

4th element - 1012 – 1015 (4 bytes)

The difference between the address of the 1st and 4th elements is 12.

Q10. What is the second part of a node in a linked list that contains the address of the next node called?

A. data

B. pointer

C. element

D. Link

Ans: D

Explanation:

The field of each node that contains the address of the next node is usually called the 'link'.

Q11. If you are using a Depth-first search (DFS) for traversing an unweighted graph, then which of the following will happen?

1. It produces the minimum spanning tree

2. It produces all pair shortest path tree

Choose the correct answer from the options given below.

A) Both 1 and 2 are true

B) Both 1 and 2 are false

C) Only 2 is true

D) Only 1 is true

Ans: D

Explanation:

Depth-first search (DFS) for traversing an **unweighted graph**, will produce the minimum spanning tree.

Only Depth-first search (DFS) for traversing a **weighted graph**, will produce all pairs shortest-path trees.

Q12. With the given information provided find out the address of Arr[17] in a 1-D array Arr[30].

- lower bound = 1

- starting base address = 1100

- size of each element is 2.

A) 1132

B) 1070

C) 1128

D) 1068

Ans: A

Explanation:

We need to find the address of Arr[17]. Starting base address is 1100.

Arr[1] – 1100 (2bytes)

Arr[2] – 1102 (2bytes)

Arr[3] – 1104 (2bytes)

Arr[4] – 1106 (2bytes)

Arr[5] – 1108 (2bytes)

Arr[6] – 1110 (2bytes)

Arr[7] – 1112 (2bytes)

Arr[8] – 1114 (2bytes)

Arr[9] – 1116 (2bytes)

Arr[10] – 1118 (2bytes)

Arr[11] – 1120 (2bytes)

Arr[12] – 1122 (2bytes)

Arr[13] – 1124 (2bytes)

Arr[14] – 1126 (2bytes)

Arr[15] – 1128 (2bytes)

Arr[16] – 1130 (2bytes)

Arr[17] – 1132 (2bytes)

Thus, the Answer for arr[17] is 1132

Q13. What will be the output of the following pseudocode?

```
Integer arr[]={10, 20, 30, 40, 5}
Integer a, s
Set s = 0
Set a = arr[1] + arr[2]
Print a
```

A) 25

B) 5

C) 50

D) 40

Ans: C

Explanation:

There is an array of integer arr[]={10,20,30,40,50}. There are two variables a and b declared. . The value initialized for s is 0. On the next line adding the 1st index value 20 and 2nd index value 30 arr[1] + arr[2](20+30), the answer is 50 will be stored in a. Finally printing the updated values of a is 50.

Q14. What will be the output of the following pseudocode?

```
Integer a, b, c
Set b = 2, a = 2
c = a ^ b
Print c
```

[Note- ^ is the bitwise exclusive OR operator that compares each bit of its first operand to the corresponding bit of its-- other bit is 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0]

A) 6

B) 4

C) 0

D) 2

Ans: C

Explanation:

There are three variables a, b and c declared. Value initialized for a is 2 and b is 2. When we do a bitwise exclusive OR of c i.e (2^2), the answer is 0. Finally, print the value of c.

Q15. Which of the following series will be printed by the given pseudocode?

```
Integer i, j, k, n
Set j=1, k=1
for(each i from 1 to 5)
  print k
  j=j+1
  k=k+j
end for
```

A) 1 3 6 10 15

B) 1 2 3 4 5

C) 2 4 6 8 10

D) 1 1 2 3 5

Ans: A

Explanation:

There are four variables i, j,k, and n declared. Value initialized for j is 1 and k is 1.

For loop, i value starts from 1 loop will run till the i<5, In the first iteration i value, is 1, printing k value is 1. Next line j value will be incremented by 1 (1+1) =>2. On the next line adding k and j (1+2), then the answer is 3.

2nd iteration i value will be incremented by 1, i=2. Print k, the updated k value is 3. on the next line j value will be incremented by 1 (2+1) =>3. On the next line adding k and j (3+3) , then the answer is 6.

3rd iteration i value will be incremented by 1, i=3. Print k, the updated k value is 6. on the next line j value will be incremented by 1 (3+1) =>4. On the next line adding k and j (6+4) , then the answer is 10.

4th iteration i value will be incremented by 1, i=4. Print k, the updated k value is 10. on the next line j value will be incremented by 1 (4+1) =>5. On the next line adding k and j (10+5) , then the answer is 15.

5th iteration i value will be incremented by 1, i=5. Print k, the updated k value is 15. Next line j value will be incremented by 1 (5+1) =>6. On the next line adding k and j (15+6), then the answer is 21.

Here for loop condition gets false, it comes out of the *for* loop. The output of Pseudocode is 1 3 6 10 15.