Multiple Choice Questions

https://csci-1301.github.io/about#authors

May 22, 2024 (01:30:39 PM)

Contents

1. Why are the instructors sharing most of the material in odt, docx, pdf, html and md?

   ☐ To insure compatibility across operating systems (Android, Linux, Windows, MacOs, …).
   ☐ To make it easier to access the resources in multiple ways (print, screen, etc.)
   ☐ All of the above.

2. What does "free" software means?

   ☐ That the software has no value.
   ☐ That the users can run the software for any purpose and study its source code.
   ☐ That it is not developed by a company.
   ☐ That the software can be downloaded at no cost.

3. In your IDE, the shortcut to compile your program is usually…

   ☐ "Build your solution", ctrl + shift + B or Cmd + B
   ☐ "Save", ctrl + S or Cmd + S
   ☐ "Exit", alt + F4 or Cmd + q
   ☐ "Start without debugging", Ctrl + F5 or Cmd + F5

4. To share or backup a project, you need to…

   ☐ share the .sln file.
   ☐ share the .cs file.
   ☐ share the .csproj file.
   ☐ zip the folder containing the .sln file and another folder with multiple files and folders in it.

5. If your IDE returns the message

   ```
   Program.cs(21,21): Error CS0117: 'Console' does not contain a definition
   ↪   for 'WiteLine' (CS0117) (Solution)
   ```

   This means that…

   ☐ That you misspelled the word "WriteLine".
   ☐ Your program successfully compiled and is ready to be executed.
   ☐ That the "Console" class does not exist.
   ☐ Your IDE was not properly installed and you should reboot your computer.

6. Consider the following code:

   ```
   int age, defaultChoice = 0;
   decimal averagePrice;
   ```

Which of the following is correct?

☐ It contains declaration and initialization statements.
☐ It declares variables of two different datatypes.
☐ Only the value of `defaultChoice` is set.
☐ All of the above.

7. Consider the following code:

```
int person = 12;
int pie = 5;
int piePerPerson = pie / person;
Console.WriteLine("Each guest gets " + piePerPerson + " pie(s).");
```

What will be displayed by it?

☐ Nothing: an error will prevent from compiling it successfully.
☐ "Each guest gets 2.4 pie(s)."
☐ "Each guest gets 0.4166666666666666666666…" (it will never ends, displaying 6 forever).
☐ "Each guest gets 0.416666666666667 pie(s)."
☐ "Each guest gets 0 pie(s)."

8. Consider the following statement:

```
decimal balance = 2.5M;
decimal price = 12;
decimal remainingBalance = balance - price;
```

Which of the following is correct?

☐ This program will not compile because the result of `balance - price` is not a decimal.
☐ This program will not compile because a decimal cannot be negative.
☐ This program will compile.
☐ This program will not compile because you cannot store an integer value (12) in a decimal.

9. The method used to read a string from the user is called…

☐ `ReadString`
☐ `ReadFrom`
☐ `ReadLine`
☐ `ReadInput`

10. Consider the following program:

```
Console.WriteLine("Enter your age.");
string fromUser = Console.ReadLine();
int age = _____ (fromUser);
```

To correctly be able to store the string in `fromUser` into age, you should replace _____ with…

☐ `(int)`
☐ `int.Parse`
☐ Nothing: as long as the user enters an integer value, we can store it into age just fine.
☐ None of the above.

11. What are, respectively, the return types of a constructor and of a `ToString` method?

☐ Constructors do not have a return type, and a `ToString` method returns a `string`.

☐ Constructors and `ToString` methods both return `string`s.
☐ Constructors returns a `string`, and a `ToString` method does not return anything (it simply displays a text).
☐ It is impossible to know ahead of time, as this depends of the class they are implemented in.

12. What is the name of a constructor method?

☐ Nothing: an error will prevent from compiling it successfully.
☐ Whatever the name of the class is.
☐ It does not have any.
☐ The name of the instance it creates.
☐ `Constructor`

13. What are the three logical connectives in C# (that we studied)?

☐ And (&&), or (||) and negation (!).
☐ Equality (==), greater than (>) and less than (<).
☐ And (`and`), or (`or`) and negation (`not`).

14. Which of the following will evaluate to true?

☐ `3 > 1 && 2`
☐ `(3 > 1) && 1 != 0`
☐ `!(3 > 1)`
☐ `3 > 1 || 2`

15. Will the following expression evaluates, and if so, what will it evaluate to?

`true == false || 2 / 1 > 0 && 3 - 1 != 2 * 0.5 + 0.5`

evaluates?

☐ It will evaluate to a number.
☐ It will evaluate to **false**.
☐ It will evaluate to **true**.
☐ It will not evaluate.
☐ None of the above.

16. What will be displayed by the following code?

```
int number = 10;
while (number <= 15)
{
    number+=2;
    Console.Write(number + " ");
}
```

☐ 12 14 16
☐ 10 11 12 13 14 15
☐ 10 11 12 13 14
☐ 10 12 16
☐ 10 12
☐ 10 12 14
☐ 12+14+16
☐ 10+11+12+13+14+15

17. What will be displayed by the following code?

```
int i = 0;
while(i < 10)
{
    Console.WriteLine(i);
}
```

☐ 0 followed by a new line, forever.
☐ 0 1 2 3 4 5 6 7 8 9
☐ 0 1 2 3 4 5 6 7 8 9 with a new line between each number
☐ Nothing

18. Consider the following code:

```
Console.WriteLine("Enter… something!");
int answer;
bool valid = int.TryParse(Console.ReadLine(), out answer);
Console.WriteLine($"returns: {valid}, value:{answer}");
```

If the user enters "Train", then it will display:

☐ returns: False, value: 0
☐ returns: True, value: 0
☐ returns: True, value: Train
☐ returns: False, value: Train
☐ Nothing: the program will crash.

19. Consider the following code:

```
string answer;
Console.WriteLine("Enter something");
answer = Console.ReadLine();
while (answer != "yes" || answer !="Yes"){
    Console.WriteLine("Enter something");
    answer = Console.ReadLine();
}
```

What can the user enters to *exit* this loop:

☐ There is nothing the user can enter to exit this loop
☐ Either "Yes" or "yes"
☐ Anything that is different from "Yes" and "yes"
☐ Anything

20. Consider the following code:

```
int answer;
Console.WriteLine("Enter something");
answer = int.Parse(Console.ReadLine());
while (answer > 10 && answer < 100){
    Console.WriteLine("Enter something");
    answer = int.Parse(Console.ReadLine());
}
```

What can the user enters to *exit* this loop?

☐ Any number not between 10 and 100 (both included)
☐ Any number between 10 and 100 (both included)
☐ Any number between 10 and 100 (both excluded)

□ Any number not between 10 and 100 (both excluded)

21. What is the correct way of creating an array of `int` of size 5 named `myArray`?

```csharp
□ int[] myArray = new int[5];
□ int[] myArray = int[5];
□ int[5] myArray = new int[];
□ int[4] myArray = new int[];
□ int myArray = new int[5];
□ int[] myArray = new int[4];
□ int[] myArray = new int(5);
□ int[] myArray = int[4];
```

22. Consider the following code:

```csharp
int[] grades = {10, 20, 5, 15};
Console.WriteLine(grades[2]);
```

What will it display?

□ 5
□ Nothing
□ 20
□ 15
□ grades
□ grades(2)
□ 10

23. Consider the following code:

```csharp
char[] grades = {'A', 'B', 'C', 'D', 'F'};
int i = 0;
while(i < grades.Length){
    i++;
    Console.WriteLine(grades[i]);
}
```

Something is wrong with it, can you tell what?

□ There will be an "Index was outside the bounds of the array." error.
□ The array is not properly initialized.
□ The loop is infinite
□ `grades.Length` is not declared.

24. What will be displayed by the following code?

```csharp
for (int e = -5; e <= 20; e += 5)
{
    Console.Write(e + " ");
}
```

□ -5 0 5 10 15 20
□ -5 0 5 10 15
□ 0 5 10 15
□ -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
□ Nothing
□ 0 5 10 15 20

25. What will be displayed by the following code?

```csharp
int variable = 0;
for (int e = 1; e <= 5; e += 1)
{
    variable += e;
}
Console.WriteLine(variable);
```

☐ 15
☐ 0
☐ Nothing
☐ 1 2 3 4 5