

Contents

1 Homework #1	1
1.1 Part I — Questions	1
1.2 Part II – Problems	2
2 Homework #2	2
2.1 Part I — Questions	2
2.2 Part II – Problems	5
3 Homework #4	6
3.1 Part I — Questions	6
3.2 Part II – Problems	7
4 Homework #5	8
4.1 Part I — Questions	8
4.2 Part II – Problems	9
5 Homework #6	9
5.1 Part I — Questions	9

Exercises

1 Homework #1

1.1 Part I — Questions

1. List five pieces of software, and three hardware components of a computer.
2. What is a GUI?
3. What, if any, is the difference between a compiler and an assembler?
4. Give a specific characteristic of C# compared to other programming languages.
5. Is the C# compiler case-sensitive?
6. Give three keywords.
7. Write a statement that would display, “Hi Mom!” (without the quotes) followed by a new line on the screen, once inserted in a proper method, compiled, and executed.
8. Write a statement that would display, “Hello!” (without the quotes) followed by a new line on the screen, once inserted in a proper method, compiled, and executed.
9. What is the limitation, if any, to the number of methods you can have per class? Why is the method called `Main` special?
10. What is a namespace?
11. Which of the following, if any, are keywords? `Welcome1 public apples int "I'm a string"`
12. Why are variables called “variables”?
13. What is the difference, if any, between `3` and `"3"`?
14. What is the difference, if any, between the `WriteLine` and `Write` methods?

15. Assume we have a variable whose name is `myVariable`, type is `string`, and value is `"My message"`. What would be displayed on the screen by the following statement? `Console.WriteLine($"Here is my variable: {myVariable}")`
16. Which of the following are correct identifier names? `$myHome3` `class` `my%variable` `ANewHope` `_train` `_ThisIsAVariable`
Is the name `myVariable` the same as `myvariable`? If not, why?
17. Circle the keywords: `myClass` `static` `Lab3-Exo1` `"Hello World"` `using` `Lab3_Part1`
18. Which one(s) of the following, if any, is a correct assignment (assuming that `variable`, `x` and `apples` have been declared as `int` variables)? `5 => variable;` `x=5;` `apples= 23` `x <= 23;` `variable =1,890;`
19. Cross out the wrong answer in the following sentences, [~~like this (incorrect)~~ | like this (correct)]:
 - “If the code does not obey the [rules | conventions] of C#, then the compiler will complain.”
 - “Every statement needs to end with [a forward slash / | a semi-colon ;].”
 - “C# is a [object-oriented | functional] programming language.”
 - “A class is made up of [a body and a header | multiple `using` statements].”
 - “An identifier can contain [only lower-case letters | letters and digits].”

1.2 Part II – Problems

1. There are 4 errors in the following code that will prevent it from compiling. Can you spot them all?

```
// My first attempt.
using System
class Wel
{
    static void Main();
{
    Console.WriteLine("Welcome \n to the lab!");
}
```

2 Homework #2

2.1 Part I — Questions

1. In C#, what is the “escape character”? Why is it useful?
2. Give the values of `a` and `b` after the following four instructions have been executed.

```
int a, b;
a = 2;
b = a * 2 + 1;
a -= 1;
```

3. Give the values of `c` and `d` after the following four instructions have been executed.

```
int c = 3, d;
d = 2 + c;
c = d * 2;
d += 2;
```

4. Is there an error in the following code? Explain the error or give the value of `b` after the second statement is executed.

```
decimal a = 1.6M;
int b = (int)a + a;
```

5. If one of the operator's operand is of type `float` and the other is of type `int`, what will be the type of the result of the operation?
6. Write an explicit conversion from a `double` variable `myDoubleVar` to an `int` variable called `myIntVar`. You don't need to re-declare those variables. Assuming `myDoubleVar`'s value is 5.89, what value would be stored in `myIntVar`?

7. Assuming that `myLastName` and `myFirstName` are two `string` variables that have been initialized, write a statement that *concatenates* them with a space and a comma in-between, and assign the resulting `string` to a variable named `fullName`. For instance, if the value of `myLastName` is "Holbertonand", and the value of `myFirstName` is "Betty", then the value of `fullName` after your operation should be "Holbertonand, Betty".
8. What is wrong with the following? Will the error(s) appear at compilation time or at execution time?

```
int age;
Console.WriteLine("Please enter your age:");
age = Console.ReadLine();
```

9. Fill in the blanks: "A class asserts that every objects created using it should have _____ (i.e., "data") and _____ (i.e., "operations")."
10. What, if any, is the difference between a parameter and an argument?
11. What is the purpose of the keyword `new`?
12. Briefly explain the difference between a local variable and an instance variable.
13. Indicate the order of evaluation of the operators in each of the following C# operations by adding parenthesis or developing the expression one step at a time, and compute the resulting value:

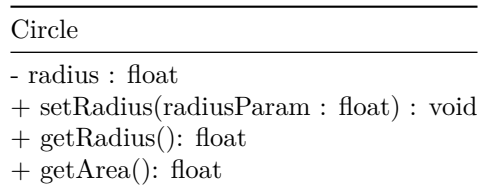
- a) `3 * 4 - 2`
- b) `3 % 2 + 3`
- c) `2 - 3 + 3 * 2`
- d) `2 + 2 * 1 - 4`

14. Write the complete implementation of a class that contains two attributes (with different data types), a setter for one attribute, a getter for the other attribute, a custom constructor, and a `ToString` method. You can re-use an example from a lecture or a lab, as long as it satisfies those conditions, or you can invent one. No need to write an application program.
15. What does the keyword `return` do?
16. Write a getter for an attribute of type `string` named `myName`.
17. Assuming `name` is a `string` instance variable, there is problem with the following setter. Fix it.

```
public int SetName(string val){
    name = val;
}
```

18. Briefly describe what a format specifier is. Write a statement that uses one.
19. Write a method for the `Rectangle` class that divides the length and width of the calling object by a factor given as a parameter.
20. Draw the UML diagram of a class named "Student" with a single attribute, "name", of type `string`, and two methods, `SetName` and `GetName`.

21. Write a `ToString` method for a `Account` class with two attributes, a `string` attribute called `name` and a `decimal` attribute called `amount`.
22. Consider the following UML diagram:



What is the name of the class, what are the methods and attributes of the class?

23. Is it possible to have more than one constructor defined for a class? If yes, how can C# know which one is called?
24. Write a constructor for a `Soda` class with one `string` attribute called `name`.
25. What is the “default” constructor? Do we always have the possibility of using it?
26. Consider the following partial class definition:

```
public class Book
{
    private string title;
    private string author;
    private string publisher;
    private int copiesSold;
}
```

- a) Write a statement that would create a `Book` object.
 - b) Write a “getter” and a “setter” for the `title` attribute.
 - c) Write a constructor for the `Book` class taking at least one argument (you are free to decide which one(s)).
27. Consider the following partial class definition:

```
class DVD
{
    private string title;
    private decimal price;
}
```

- a) Write a “setter” for the `title` attribute.
 - b) Write a constructor for the `DVD` class that takes two arguments.
 - c) Write a method called `Discount` that decreases the `price` attribute by 20.55%.
 - d) Write a (good, informative) `ToString` method for the class.
 - e) Write statements that ask the user to enter a price and then create a `DVD` object with a `price` attribute equal to the price the user entered. (The object’s `title` attribute can be anything you choose).
 - f) Draw the UML class diagram for the class you obtained by adding the above four methods to our original class definition.
28. Consider the following partial class definition:

```
class Book
{
    private string title;
    private decimal price;
}
```

- a) Write a “getter” for the `title` attribute.
- b) Write a constructor for the `Book` class that takes two arguments.
- c) Write a method called `AddTaxes` that increases the `price` attribute by 6.35%.
- d) Write a (good, informative) `ToString` method for that class.
- e) Write statements that ask the user to enter a price and then create a `Book` object with a `price` attribute equal to the price the user entered. (The object’s `title` attribute can be anything you choose).
- f) Draw the UML class diagram for the class you obtained by adding the above four methods to our original class definition.

29. Assume that my `Pet` class contains one custom constructor:

```
public Pet(string nameP, char genderP){
    name = nameP;
    gender = genderP;
}
```

What is the problem with the following statement?

```
Pet myPet = new Pet('M', "Bob");
```

2.2 Part II – Problems

There is only one problem this time, and it is harder than what you’ll be asked to do during the exam. Being able to solve it is an excellent sign that you are ready.

1. You are going to design a class named `Triangle`. A triangle has three angles, but knowing the value of only two angles is sufficient to determine the value of the third, since they always add up to 180°. Hence, it is sufficient to have only two `double` attributes, `angle1` and `angle2`. We want to define several methods:
 - a no-arg constructor that sets the value of `angle1` to 60.0 and the value of `angle2` to 60.0,
 - another constructor, that takes two arguments, and assigns to `angle1` the value of the first argument, and assigns to `angle2` the value of the second argument,
 - getters for `angle1` and `angle2`,
 - a method that computes and returns the value of the third angle, that we name `ComputeAngle3`,
 - a method that rotate the triangle: the value of the first angle should be replaced with the value of the second angle, and the value of the second angle should be replaced with the value of the third angle.
- a) Write the UML diagram for the `Triangle` class.
- b) Write the full, compilable implementation of the `Triangle` class.

3 Homework #4

3.1 Part I — Questions

1. What is sequential processing?
2. Decide if the following boolean expressions will evaluate to `true` or `false`:
 - `3 > 2.0 && false`
 - `(4 != 3) || false`
 - `'A' == 'b' && ! false`
 - `(! false) == (true || 4 == 3)`
3. Decide if the following Boolean expressions will evaluate to `true` or `false`:
 - `3 > 2.0 || true`
 - `(4 != 3) && false`
 - `'A' == 'b' && ! false`
 - `(! true) == (true || 4 != 3)`
4. For each of the following Boolean expressions, decide if it will evaluate to `true` or `false`:
 - a) `('y' == 'Y') && true`
 - b) `6 + 2 \< 8 || 3 > 4`
 - c) `(true && 4 == 3) == false`
 - d) `4 > 4 && !false`
5. For each of the following Boolean expressions, decide if it will evaluate to `true` or `false`:
 - a) `('y' != 'Y') && true`
 - b) `6 + 2 \< 12 || 3 > 4`
 - c) `(true && 4 >= 3) == false`
 - d) `13 \<= 4 * 3 || !false`
6. What is the relational operator used to determine whenever two values are different?
7. Give three relational operators, and then two logical operators.
8. Is there a simpler way to write the expression `over21 == true`, assuming that `over21` is a Boolean variable?
9. In C#, is there a difference between `=` and `==`? Write a statement that uses `=`.
10. Is the following statement correct, i.e., would it compile, assuming `myFlag` is a `bool` variable, and `myAge` is an initialized `int` variable?

```
if ( myAge > 20 )
{
    myFlag = true
};
```
11. Write an `if` statement that prints “Bonjour !” if the value of the `char` variable `lang` is `'f'`.
12. Write a `boolean` expression that evaluates to `true` if a variable `x` is between 3 (excluded) and 5 (included).
13. Write an `if-else` statement that assigns `"Minor"` to an already declared `string` variable `legalStatus` if `age` is strictly less than 18, and that assigns `"Major"` to `legalStatus` otherwise.
14. Assume we initialized an `int` variable called `courseNumber` and a `string` variable called `courseCode`. Write a series of statements that will display:

- a) "I'm taking this class!" if `courseNumber` is 1301 and `courseCode` is CSCI;
- b) "That's my major!" if `courseCode` is CSCI;
- c) "Is that an elective?" if `courseNumber` is greater than 3000; or
- d) "Is it a good class?" otherwise.

Your program should display exactly one message.

15. Assume we previously initialized an `int` variable called `graduationYear` and a `string` variable called `graduationSemester`. Write a series of statements that will display:
 - a) "I will graduate at the same time!" if `graduationYear` is 2023 and `graduationSemester` is Fall;
 - b) "I love this season." if `graduationSemester` is Spring;
 - c) "That is in a long time!" if `graduationYear` is greater than 2025; or
 - d) "I hope you'll have an in-person ceremony!" otherwise.

Your program should display exactly one message.

16. What will be displayed on the screen by the following program?

```
int x = 3, y = 2, z = 4;
if (x > y) {z += y;}
if (x > z) {y -= 4;}
Console.WriteLine($"x is {x}, y is {y}, and z is {z}.");
```

17. (*We'll use the 24-hour clock, sometimes called "military time".*) Assuming that an `int` variable `hour` has been initialized, write part of a program that would display on the screen "Good morning" if `hours` is less than or equal to 12, and "Hello" otherwise.
18. Assuming that `myString` is a string variable, write a statement that prints "Hello, Melody!" if the value of `myString` is equal to `Melody`, and nothing otherwise.
19. What will be displayed on the screen by the following program?

```
int x = 3, y = 2, z = 4;
if (y >= z) {z += y;}
else if (x != y) { if (false) {z -= 3;} else {z += x;}}
Console.WriteLine($"x is {x}, y is {y}, and z is {z}.");
```

20. Give an example of an `if` statement that could not be rewritten as a `switch`.

3.2 Part II – Problems

This time, the two exercises **do not** require a computer, and are here to craft on your problem-solving skills. Make sure you feel ready before starting them, try to do them with a limited amount of time and without notes, and check your answer using your IDE.

1. Write a program that asks the user to write a country name and stores the user's input into a string variable. Then, compare that string with "france": if it is equal, then display at the screen "Bienvenue en France !". Then, compare that string with "usa": if it is equal, then display at the screen "Welcome to the US!". If the string is different from both "france" and "usa", then display at the screen "Welcome to" followed by the name of the country the user typed in. Can you think of two ways to implement this program, one using `if-else-if` statements, the other using `switch`?
2. You want to write a small program for an on-line printing company. Your program should ask the user to chose a format (10 × 15 centimeters, or 8 × 11 inches), ask if it is the first time the customer order through your company, and a number of copies. Then, calculate the total cost of printing those pictures, knowing that
 - Printing a 10 × 15 centimeters picture costs \$0.20, printing a 8 × 11 inches picture costs \$0.25,

- A new customer gets a \$3 coupon if the order is more than \$5,
- A 10% discount is given if more than 50 copies were ordered,
- The two previous offers can be cumulated.

Display on the screen a message starting by “Welcome!”, then a new line, then “We cherish our new customers” if it is the first time the user uses your company, “, so we’re giving you a \$3 discount!” if the user is allowed to get the coupon, then print the total and “You had a 10% discount!” if the user ordered more than 50 copies. See below for examples of execution, where the user input is underlined, and hitting carriage return is represented by ↵.

```
Enter 'c' for 10x15cm, anything else for 8x11in.
c ↵
Is this your first time here? Type 'y' for 'yes'.
y ↵
Enter a number of copies.
90 ↵
Welcome!
We cherish our new customers, so we are giving you a $3 discount!
Your total is $13.50. You had a 10% discount!

Enter 'c' for 10x15cm, anything else for 8x11in.
p ↵
Is this your first time here? Type 'y' for 'yes'.
Not_at_all ↵
Enter a number of copies.
120 ↵
Your total is $27.00. You had a 10% discount!
```

4 Homework #5

4.1 Part I — Questions

1. Assume you are given an un-assigned `string` variable `letterGrade`, and an already assigned `float` variable `numberGrade`. Write a small program that assigns "A" to `letterGrade` if `numberGrade` is between 100 and 90 (both included), "B" to `letterGrade` if `numberGrade` is between 90 (excluded) and 80 (included), etc., and "Invalid data" if `numberGrade` is strictly lower than 0 or strictly greater than 100. Should you use a `switch` statement or a `if ...else if ...else`?
2. Given an `int` variable `counter`, write three statements to decrement its value by 1.
3. What will be displayed on the screen?

```
int x = 3, y = 7;
Console.WriteLine (x++ +" and "+ --y);
```

4. What will be displayed on the screen by the following program? Write the spaces and new line explicitly.

```
int counter = 10;
while (counter > 5)
{
    counter--;
    Console.Write(counter + "\n");
    if (counter == 7) {
        Console.WriteLine("Bingo");
    }
}
```


5. What will be displayed on the screen by the following program?

```
int counter = 10;
while (counter != 5) ;
Console.Write(counter + "\n");
counter--;
```

6. What is input validation? Name a control structure that can be used to perform it. Why is it important?
7. What is a sentinel value?
8. Write a small program that asks the user for an integer, and displays “It is positive” if the number entered is positive, “It is negative” if the number entered is negative, and “Not a number” if the user entered a string that is not an integer.
9. Write a program containing a **while** loop that would display the numbers between -100 and 100 (both included) with a space between them when executed.
10. Assume you are given an initialized **string** variable **name**, and a **string** variable **field**. Write a small program that assigns to **field**
- “CS” if **name** is “Turing” or “Liskov”,
 - “Math.” if **name** is “Aryabhata” or “Noether”,
 - “Unknown” otherwise.
11. Write a program that asks the user to enter a value between 1900 and 1999 (both included), and asks again as long as the user enters integers outside that range.

4.2 Part II – Problems

1. Write a **switch** statement that calculates the number of days in a particular month. You should assume that you are given already assigned **month** and **year** **int** variables, and that your program should set an already declared **int** **numberOfDays** variable to 28, 29, 30 or 31 depending on the month / year combination. Your program should start with a **switch** matching **month** against certain values, and, if **month** is 2, uses an **if** statement to decide whenever the number of days is 28 or 29. You can use something like

```
switch (month) {
    ...
    case (2):
        if ...
        ...
        break;
    ...
}
```

5 Homework #6

5.1 Part I — Questions

1. Write a statement that creates a 10-element **int** array named **numbers**.
2. What is wrong with the following array declaration?

```
int[] books = new int[-1];
```

3. What will be displayed on the screen by the following program?

```
for (int num = 3 ; num <= 5 ; num++)  
    Console.Write(num + " ");
```

4. Write a 'for' loop that displays on the screen the sequence "1, 2, 3, 4, 5, 6, 7, 8, 9, 10, ".
5. Write a 'for' loop that displays on the screen the sequence "1, 2, 3, 4, 5, 6, 7, 8, 9, 10", (note that there is no comma after 10).
6. Write a 'for' loop that displays on the screen the sequence "1 3 5 7 9 ".
7. Given an `int` variable `myVar` initialized with a positive value, write a loop that sums the integers between 0 and `myVar` (i.e., $0 + 1 + \dots + (\text{myVar} - 1) + \text{myVar}$).
8. Consider the following code:

```
for (int y = 1; y <= 3; y++)  
{  
    for (int z = 1; z < 5; z++)  
        Console.WriteLine("Scene " + y + ", take " + z + ". " );  
    Console.WriteLine();  
}
```

How many times does the outer loop iterate (i.e., how many scenes are shot)? How many times does the inner loop iterate (i.e., how many takes for each scene)? Finally, what is the total number of iterations of the nested loops (i.e., how many takes are made, total)?

9. What will be displayed on the screen by the following code?

```
int[] values = new int[6];  
for (int i = 0 ; i < 6 ; i++)  
    values[i] = (i*2);  
foreach (int j in values)  
    Console.WriteLine(j);
```

10. Suppose we are given an `int` array `dailyPushUp` with 7 elements. Write a piece of code that display the value of the elements stored in the array `dailyPushUp`.
11. What is "array bounds checking"? When does it happen?
12. What would be the size of the `test` array after the following statement has been executed?

```
int[] test = {3, 5, 7, 0, 9};
```

13. Write a statement that creates and initializes a `double` array with the values 12.5, 89.0 and 3.24.
14. What is the value of `count` and the content of `number` once the following has been executed?

```
int count=2;  
int[] number={3, 5, 7};  
number[count--] = 8;  
number[count]--;
```

15. Describe what the following code would do.

```
int[] record = { 3, 8, 11 };  
int accumulator = 0;  
foreach (int i in record)  
    accumulator += i;
```

16. Assuming we are given an `int` array named `arrayF`, write a program that adds one to each of its elements. That is, if `arrayF` contains 3, 5, 7 and -2 before your program is executed, it should then contain 4, 6, 8 and -1 after your program was executed.
17. Assuming we are given an `int` array named `arrayF`, write a program that displays the product of its elements. That is, if `arrayF` contains 2, 3 and -1, then your program should display -6.
18. Write a static method (header included) that takes as argument an `int` array, and display on the screen the value of each element of that array.
19. Write a static method (header included) that takes as argument an `int` array, and stores the value 10 in each element of that array.