

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?**

Solution. An **if...else if...else** is the right structure for this task:

```
float numberGrade;
string letterGrade;
numberGrade = -60; // This is just an example, feel free to change it.

if(numberGrade > 100 || numberGrade < 0){
    // It's actually easier to get rid of the "invalid" cases first.
    letterGrade = "Invalid Data";
}
else if (numberGrade >= 90){
    letterGrade = "A";
}
else if (numberGrade >= 80){
    letterGrade = "B";
}
else if (numberGrade >= 70){
    letterGrade = "C";
}
else if (numberGrade >= 60){
    letterGrade = "D";
}
else{
    // We know the value is greater than 0 but strictly lower than 60.
    letterGrade = "F";
}

Console.WriteLine(numberGrade + " corresponds to " + letterGrade);
```

2. Given an **int** variable `counter`, write three statements to decrement its value by 1.

Solution. We actually know four ways to do that:

```
counter = counter - 1;
counter -= 1;
counter--;
-- counter;
```

3. What will be displayed on the screen?

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

Solution. “3 and 6”

4. What will be displayed at the screen by the following program?

```
int counter = 2;
while (counter != 5)
{
    Console.Write(counter + "\n");
    counter++;
}
```

Solution.

2
3
4

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

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

Solution. Nothing, and it will loop.

6. What will be displayed at the screen by the following program?

```
int counter = 7;
while (counter != 2)
    Console.Write(counter + "\n");
counter--;
```

Solution. 7 infinitely many times.

7. What is input validation? Name a control structure that can be used to perform it. Why is it important?

Solution. Making sure the user's input is valid. The **while** loop. Because we can't trust the user.

8. What do we name a variable that is incremented at every iteration of a loop, i.e., that keeps the running total?

Solution. An accumulator.

9. What is a sentinel value?

Solution. It's a value that will trigger an exit from a loop. It is a value that was agreed on, and that signify “I now want to exit the loop.”.

10. Write a program that asks the user to enter a value between 0 and 10, and asks again as long as the user enters integers outside that range.

Solution.

```
int answer;
do{
    Console.WriteLine("Enter a value between 0 and 10 (both included).");
    answer = int.Parse(Console.ReadLine());
}while(answer > 10 || answer < 0);
```

11. 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.

Solution.

```
int answer;
Console.WriteLine("Enter an integer");
if(!int.TryParse(Console.ReadLine(), out answer)){
    Console.WriteLine("Not a number");
}
else if (answer > 0){
    Console.WriteLine("Positive");
}
else{
    Console.WriteLine("Negative");
}
```

12. 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.

Solution.

```
int counter = -100;
while(counter <= 100){
    Console.Write(counter++ + " ");
}
```

13. 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.

Solution.

```
string name;
name = "Turing"; // Value given as an example, change it to test.
string field;

switch(name){
    case("Turing"):
```

```
    case("Liskov"):
        field = "CS";
        break;
    case("Aryabhata"):
    case("Noether"):
        field = "Math.";
        break;
    default:
        field = "Unknown";
        break;
}
Console.WriteLine(name + " worked in " + field + ".");
```

14. 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.

Solution.

```
int answer;
do{
    Console.WriteLine("Enter a value between 1900 and 1999 (both included).");
    answer = int.Parse(Console.ReadLine());
}while(answer < 1900 || answer > 1999);
```

