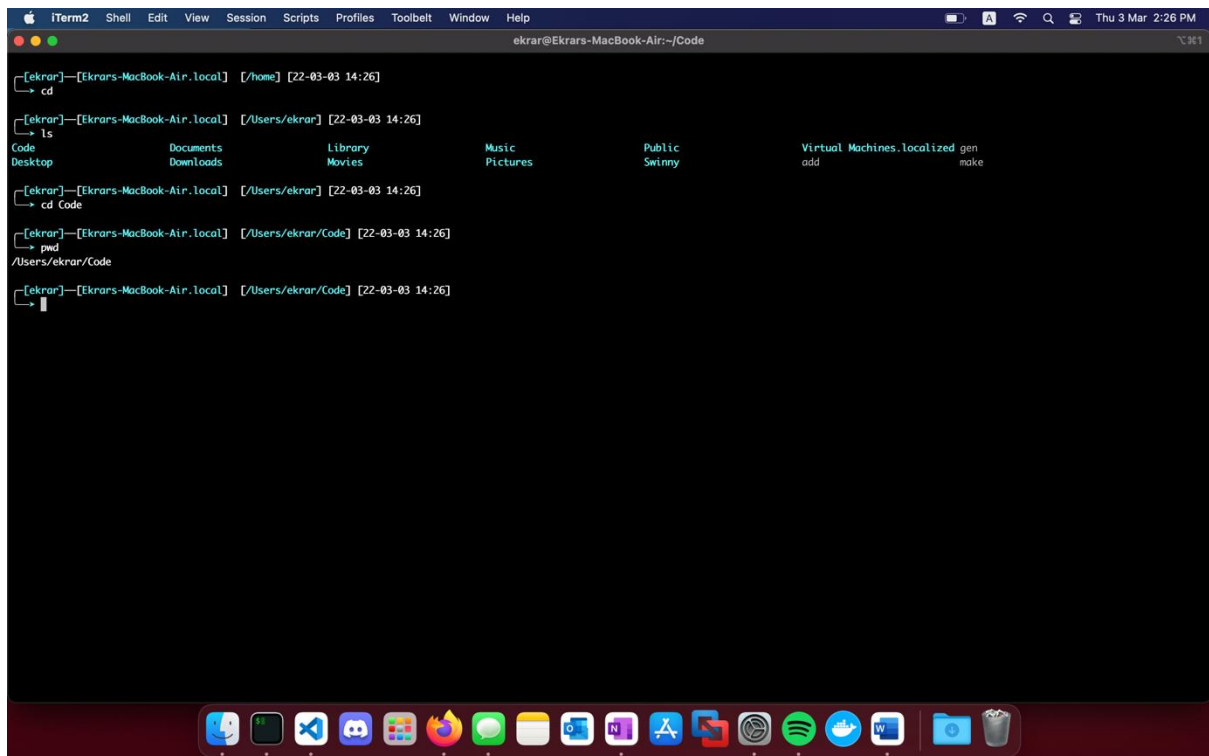


1.1P: Preparing for OOP – Answer Sheet

1. Explain the following terminal instructions:
 - a. `cd`: changes directory
 - b. `ls`: lists all the files in the current directory
 - c. `pwd`: prints the name of the working directory



The screenshot shows a macOS terminal window titled 'iTerm2'. The user 'ekrar' is logged into 'Ekrars-MacBook-Air'. The terminal session shows the following commands and outputs:

```
[ekrar]—[Ekrars-MacBook-Air.local] [/home] [22-03-03 14:26]
→ cd
[ekrar]—[Ekrars-MacBook-Air.local] [/Users/ekrar] [22-03-03 14:26]
→ ls
Code      Desktop  Documents  Library    Music      Public      Virtual Machines.localized  gen  make
Desktop  Downloads  Movies     Pictures   Swinny
[ekrar]—[Ekrars-MacBook-Air.local] [/Users/ekrar] [22-03-03 14:26]
→ cd Code
[ekrar]—[Ekrars-MacBook-Air.local] [/Users/ekrar/Code] [22-03-03 14:26]
→ pwd
/Users/ekrar/Code
[ekrar]—[Ekrars-MacBook-Air.local] [/Users/ekrar/Code] [22-03-03 14:26]
→
```

2. Consider the following kinds of information, and suggest the most appropriate data type to store or represent each:

Information	Suggested Data Type
A person's name	String
A person's age in years	Integer
A phone number	Integer
A temperature in Celsius	Float
The average age of a group of people	Array
Whether a person has eaten lunch	Boolean

3. Aside from the examples already given, come up with an example of information that could be stored as:

Data type	Suggested Information
String	State Name
Integer	Number of Pizza Slices
Float	Average Rainfall this Year
Boolean	Whether I eat like Pizza

4. Fill out the following table, evaluating the value of each expression and identifying the data type the value is most likely to be:

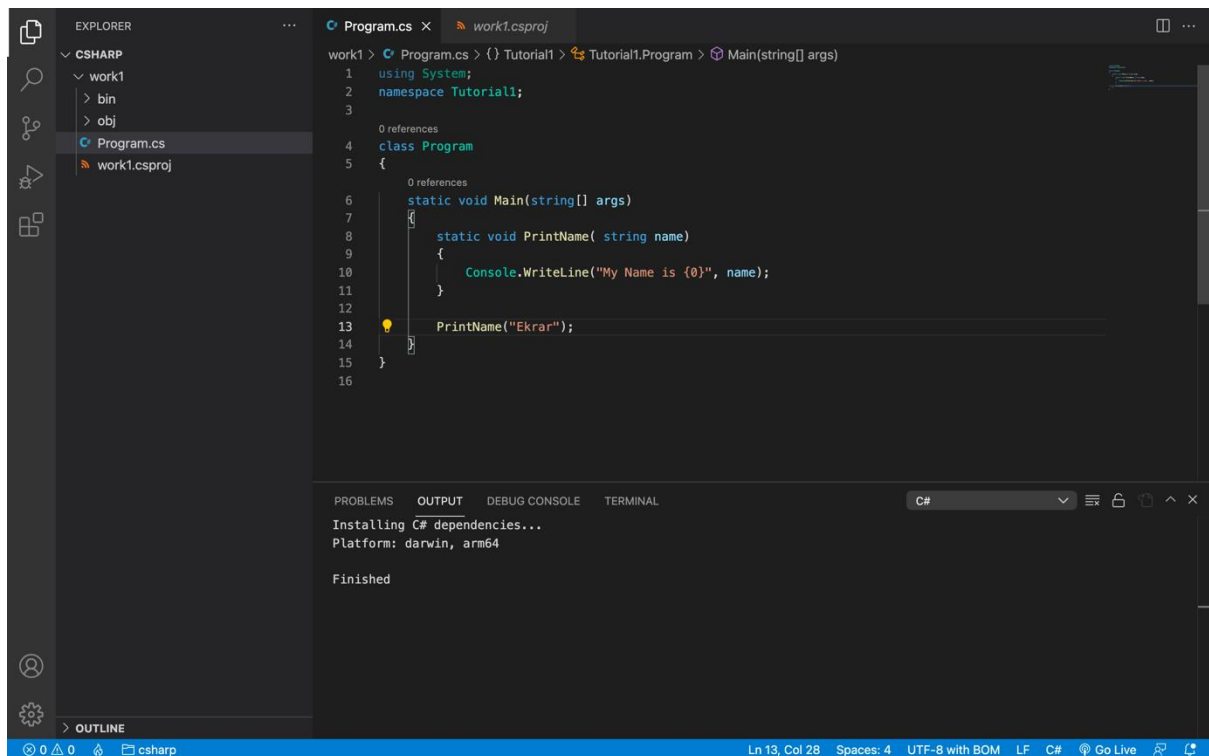
Expression	Given	Value	Data Type
5		5	Integer
True		True	Boolean
a	a = 2.5	2.5	Float
1 + 2 * 3		7	Integer
a and False	a = True	False	Boolean
a or False	a = True	True	Boolean
a + b	a = 1 b = 2	3	Integer
2 * a	a = 3	6	Integer
a * 2 + b	a = 1.5 b = 2	5	Integer
a + 2 * b	a = 1.5 b = 2	5.5	Float
(a + b) * c	a = 1 b = 1 c = 5	10	Integer
"Fred" + " Smith"		Fred Smith	String
a + " Smith"	a = "Wilma"	Wilma Smith	String

5. Explain the difference between **declaring** and **initialising** a variable.

The difference between the two is declaration only tells which data type the variable belongs to but initialization of a variable also assigns it a value.

6. Explain the term **parameter**. Write some code that demonstrates a simple use of a parameter.

A parameter is a kind of variable that the function takes in as an input.



The screenshot shows the Visual Studio Code editor with a C# project named 'work1'. The Explorer pane on the left shows the project structure with 'Program.cs' selected. The main editor displays the following code:

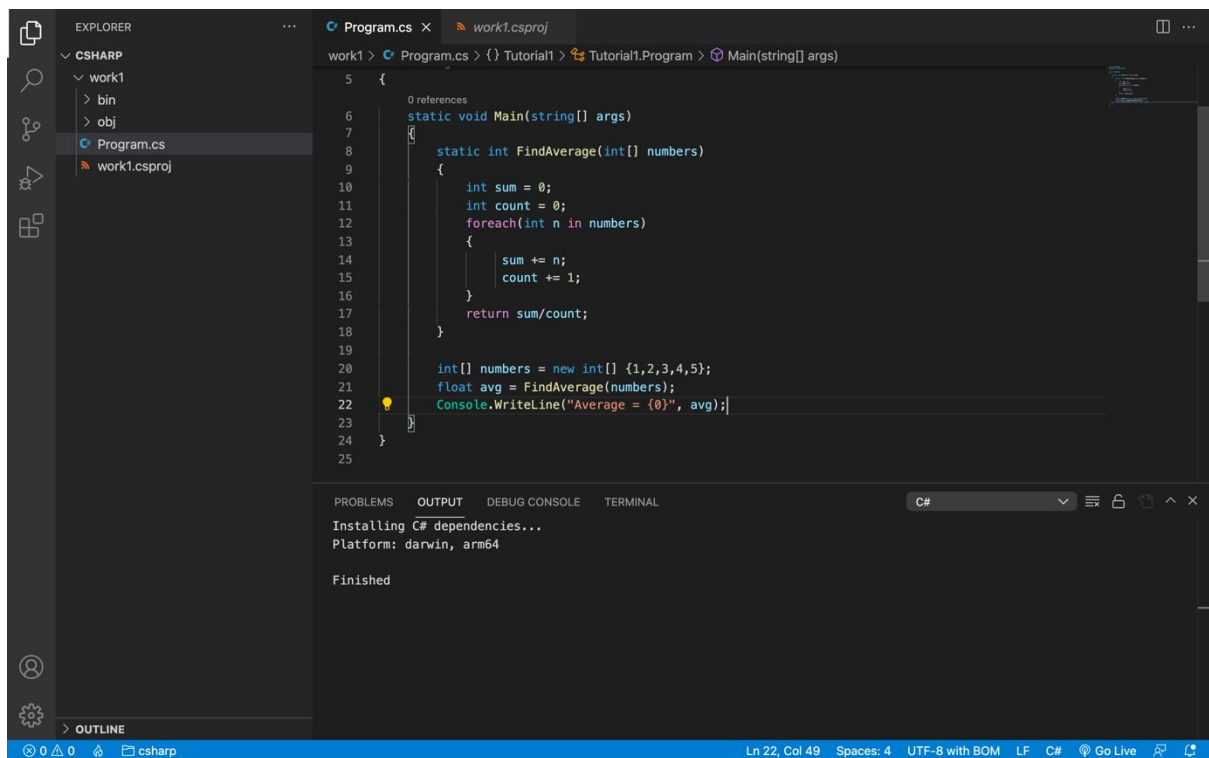
```
1 using System;
2 namespace Tutorial1;
3
4 class Program
5 {
6     static void Main(string[] args)
7     {
8         static void PrintName( string name)
9         {
10             Console.WriteLine("My Name is {0}", name);
11         }
12
13         PrintName("Ekrrar");
14     }
15 }
16
```

The bottom panel shows the OUTPUT window with the message: 'Installing C# dependencies... Platform: darwin, arm64 Finished'.

7. Using an example, describe the term **scope**.

Scope is a boundary in programming languages where variables can be accessed or referenced

8. In any procedural language you like, write a function called Average, which accepts an array of integers and returns the average of those integers.

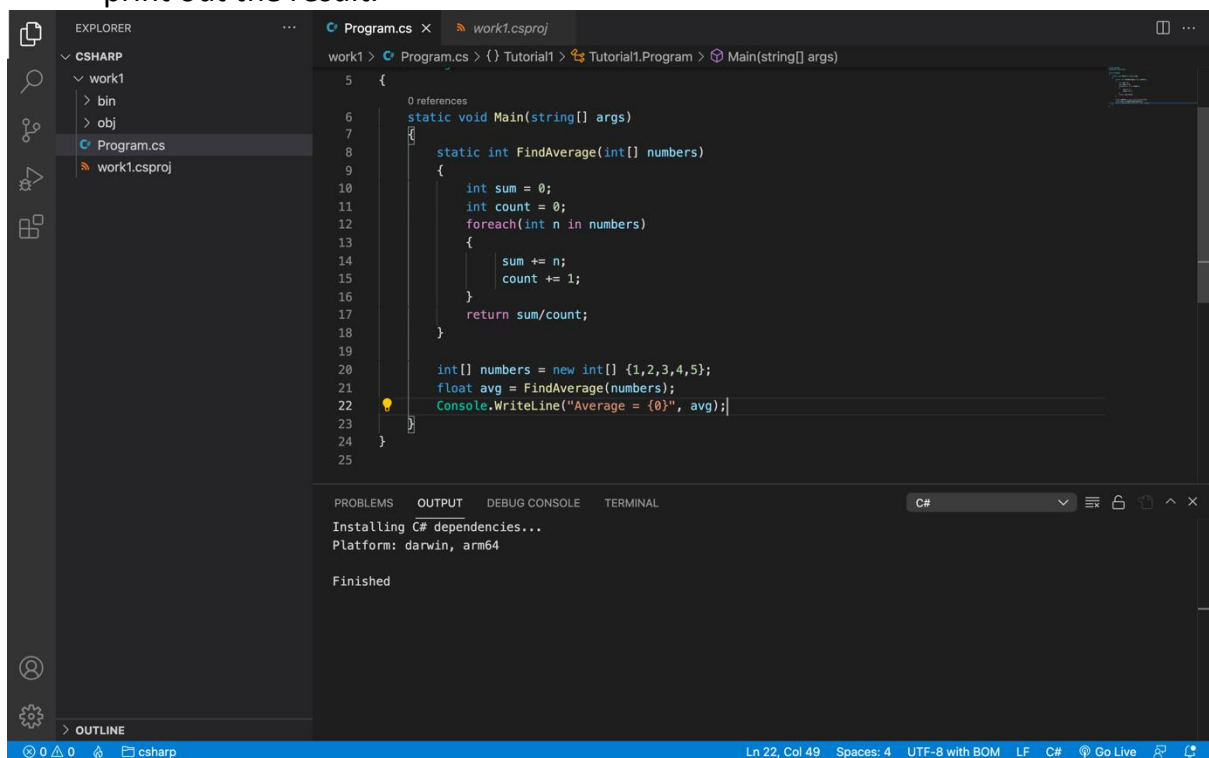


The screenshot shows the Visual Studio Code editor with a C# project named 'work1'. The Explorer pane on the left shows the project structure with 'Program.cs' selected. The main editor displays the following code:

```
5 {  
6     0 references  
7     static void Main(string[] args)  
8     {  
9         static int FindAverage(int[] numbers)  
10        {  
11            int sum = 0;  
12            int count = 0;  
13            foreach(int n in numbers)  
14            {  
15                sum += n;  
16                count += 1;  
17            }  
18            return sum/count;  
19        }  
20        int[] numbers = new int[] {1,2,3,4,5};  
21        float avg = FindAverage(numbers);  
22        Console.WriteLine("Average = {0}", avg);  
23    }  
24 }  
25
```

The bottom pane shows the 'OUTPUT' tab with the message: 'Installing C# dependencies... Platform: darwin, arm64 Finished'.

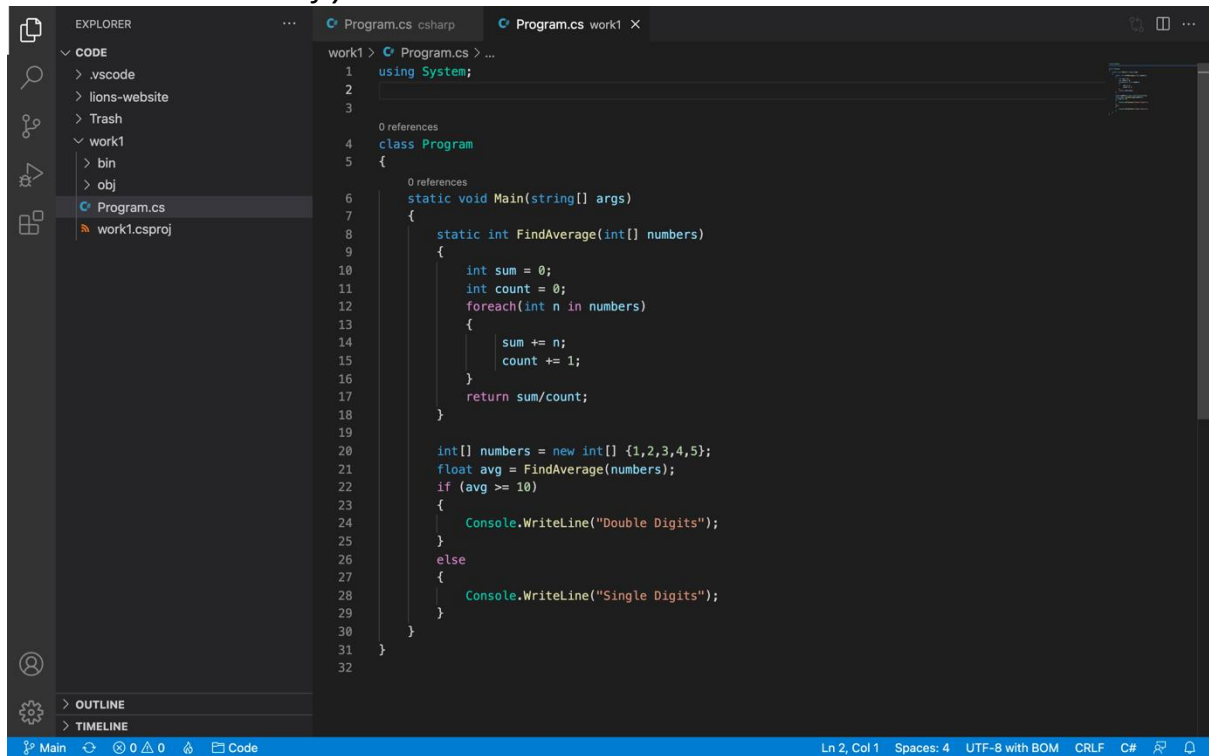
9. In the same language, write the code you would need to call that function and print out the result.



This screenshot is identical to the one above, showing the same C# code in Visual Studio Code. The code defines a static method `FindAverage` that calculates the average of an array of integers, and a `Main` method that calls `FindAverage` with an array `{1,2,3,4,5}` and prints the result to the console.

10. To the code from 9, add code to print the message “Double digits” if the average is above 10. Otherwise, print the message “Single digits”.

<insert a screenshot of your code here>



The screenshot shows the Visual Studio Code editor with a C# file named Program.cs. The code defines a class Program with a static Main method. Inside Main, a static method FindAverage is called with an array of numbers {1,2,3,4,5}. The FindAverage method calculates the sum of the numbers and divides it by the count to find the average. The Main method then checks if the average is greater than or equal to 10 and prints "Double Digits" or "Single Digits" accordingly.

```
1 using System;
2
3
4 class Program
5 {
6     static void Main(string[] args)
7     {
8         static int FindAverage(int[] numbers)
9         {
10             int sum = 0;
11             int count = 0;
12             foreach(int n in numbers)
13             {
14                 sum += n;
15                 count += 1;
16             }
17             return sum/count;
18         }
19
20         int[] numbers = new int[] {1,2,3,4,5};
21         float avg = FindAverage(numbers);
22         if (avg >= 10)
23         {
24             Console.WriteLine("Double Digits");
25         }
26         else
27         {
28             Console.WriteLine("Single Digits");
29         }
30     }
31 }
32
```

<insert a screenshot of your whole programming here>