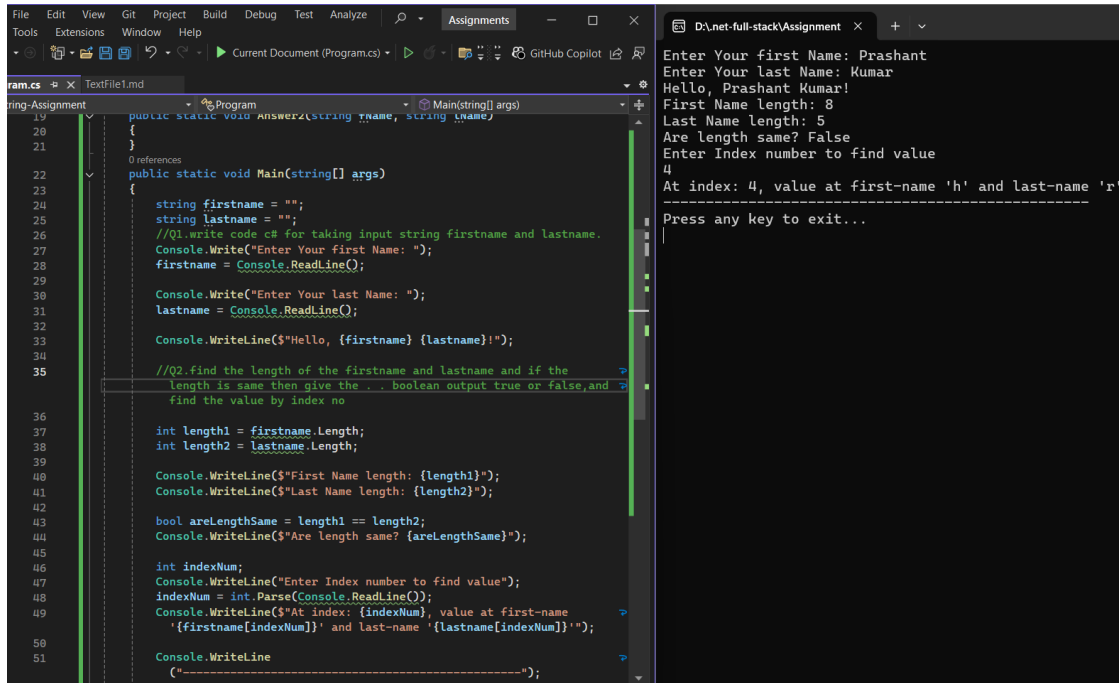


## # STRING QUESTIONS

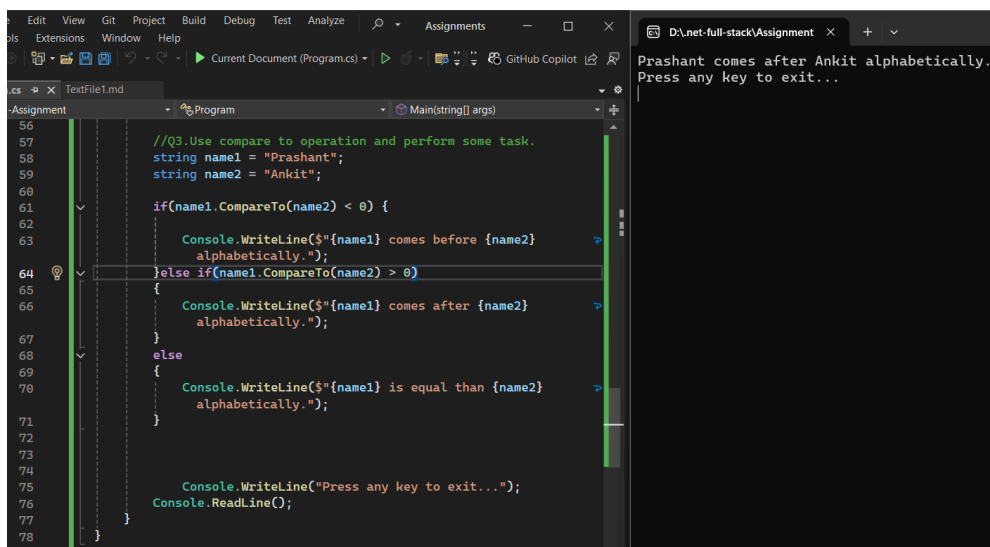
Q1.write code c# for taking input string firstname and lastname.

Q2.find the length of the firstname and lastname and if the length is same then give the . . boolean output true or false,and find the value by index no



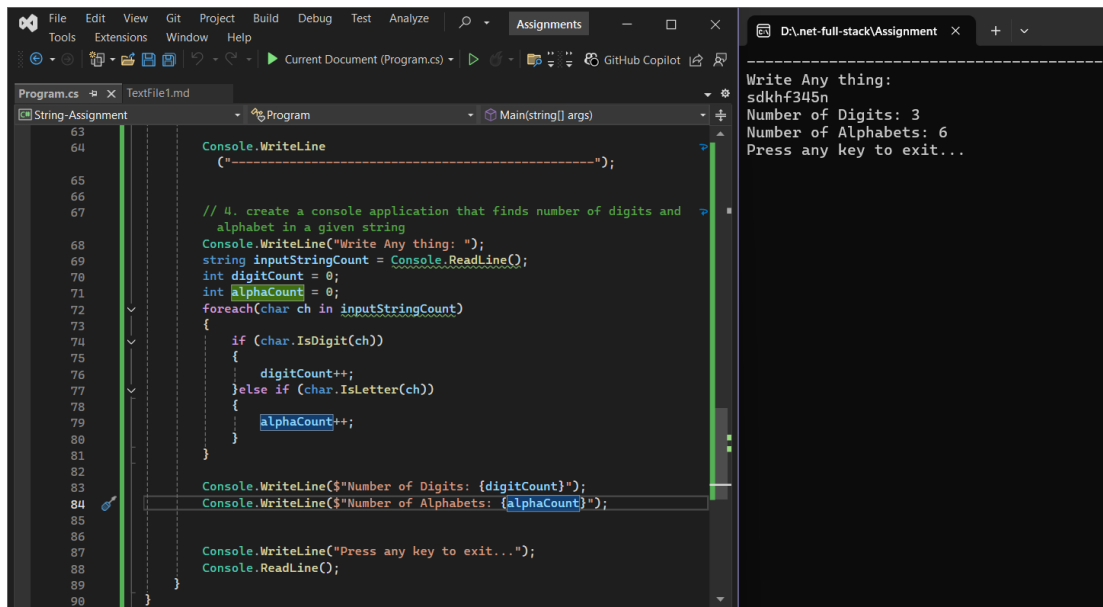
```
19 public static void Answer2(string fname, string lname)
20 {
21 }
22 public static void Main(string[] args)
23 {
24     string firstname = "";
25     string lastname = "";
26     //Q1.write code c# for taking input string firstname and lastname.
27     Console.WriteLine("Enter Your first Name: ");
28     firstname = Console.ReadLine();
29
30     Console.WriteLine("Enter Your last Name: ");
31     lastname = Console.ReadLine();
32
33     Console.WriteLine($"Hello, {firstname} {lastname}!");
34
35     //Q2.find the length of the firstname and lastname and if the
36     //length is same then give the . . boolean output true or false,and
37     //find the value by index no
38
39     int length1 = firstname.Length;
40     int length2 = lastname.Length;
41
42     Console.WriteLine($"First Name length: {length1}");
43     Console.WriteLine($"Last Name length: {length2}");
44
45     bool areLengthSame = length1 == length2;
46     Console.WriteLine($"Are length same? {areLengthSame}");
47
48     int indexNum;
49     Console.WriteLine("Enter Index number to find value");
50     indexNum = int.Parse(Console.ReadLine());
51     Console.WriteLine($"At index: {indexNum}, value at first-name
52     '{firstname[indexNum]}' and last-name '{lastname[indexNum]}';");
53
54     Console.WriteLine
55     ("-----");
```

Q3.Use compare to operation and perform some task.



```
56 //Q3.Use compare to operation and perform some task.
57 string name1 = "Prashant";
58 string name2 = "Ankit";
59
60 if(name1.CompareTo(name2) < 0) {
61     Console.WriteLine($"{{name1}} comes before {{name2}}
62     alphabetically.");
63 }
64 else if(name1.CompareTo(name2) > 0)
65 {
66     Console.WriteLine($"{{name1}} comes after {{name2}}
67     alphabetically.");
68 }
69 else
70 {
71     Console.WriteLine($"{{name1}} is equal than {{name2}}
72     alphabetically.");
73 }
74
75 Console.WriteLine("Press any key to exit...");
76 Console.ReadLine();
77 }
78 }
```

4. create a console application that finds number of digits and alphabet in a given string



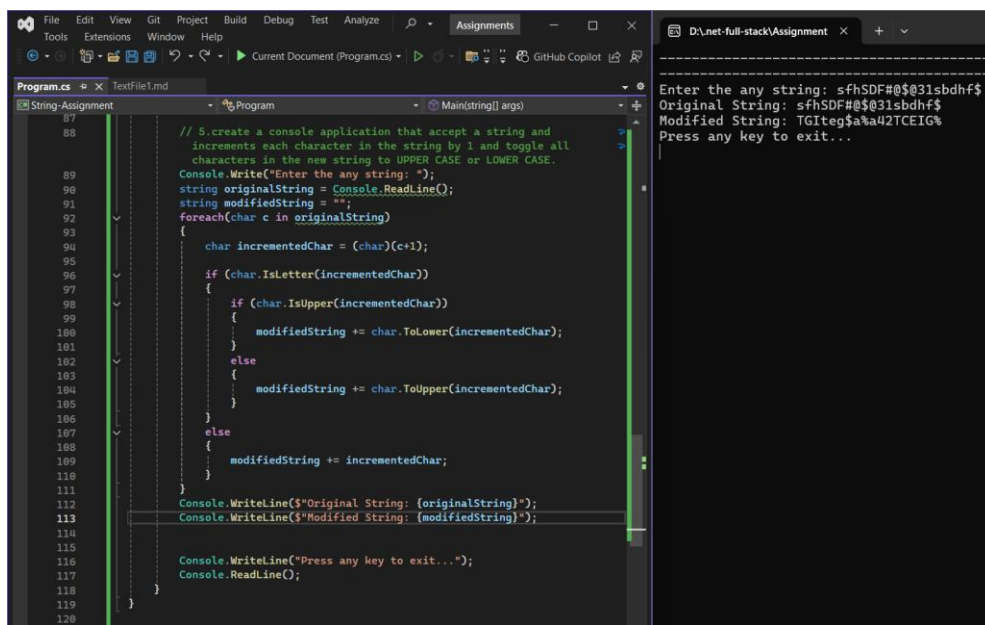
The screenshot shows a Visual Studio IDE with a C# console application. The code in Program.cs is as follows:

```
63 Console.WriteLine
64 ("-----");
65
66 // 4. create a console application that finds number of digits and
67 // alphabet in a given string
68 Console.WriteLine("Write Any thing: ");
69 string inputStringCount = Console.ReadLine();
70 int digitCount = 0;
71 int alphaCount = 0;
72 foreach(char ch in inputStringCount)
73 {
74     if (char.IsDigit(ch))
75     {
76         digitCount++;
77     } else if (char.IsLetter(ch))
78     {
79         alphaCount++;
80     }
81 }
82
83 Console.WriteLine($"Number of Digits: {digitCount}");
84 Console.WriteLine($"Number of Alphabets: {alphaCount}");
85
86 Console.WriteLine("Press any key to exit...");
87 Console.ReadLine();
88
89
90 }
```

The output window on the right shows the following text:

```
Write Any thing:
sdkhf345n
Number of Digits: 3
Number of Alphabets: 6
Press any key to exit...
```

5.create a console application that accept a string and increments each character in the string by 1 and toggle all characters in the new string to UPPER CASE or LOWER CASE.



The screenshot shows a Visual Studio IDE with a C# console application. The code in Program.cs is as follows:

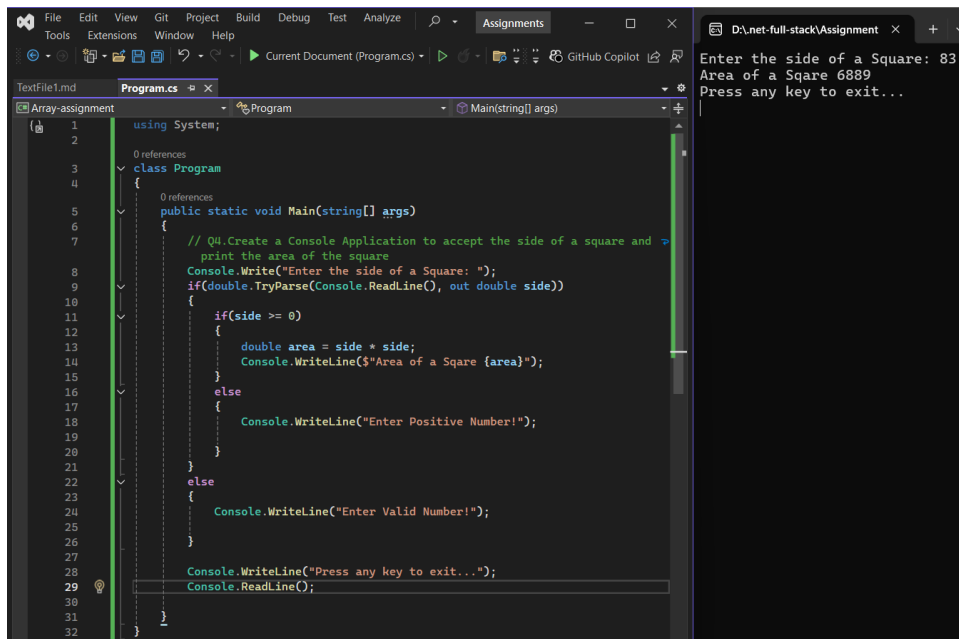
```
87
88 // 5.create a console application that accept a string and
89 // increments each character in the string by 1 and toggle all
90 // characters in the new string to UPPER CASE or LOWER CASE.
91 Console.WriteLine("Enter the any string: ");
92 string originalString = Console.ReadLine();
93 string modifiedString = "";
94 foreach(char c in originalString)
95 {
96     char incrementedChar = (char)(c+1);
97
98     if (char.IsLetter(incrementedChar))
99     {
100         if (char.IsUpper(incrementedChar))
101         {
102             modifiedString += char.ToLower(incrementedChar);
103         }
104         else
105         {
106             modifiedString += char.ToUpper(incrementedChar);
107         }
108     }
109     else
110     {
111         modifiedString += incrementedChar;
112     }
113 }
114 Console.WriteLine($"Original String: {originalString}");
115 Console.WriteLine($"Modified String: {modifiedString}");
116
117 Console.WriteLine("Press any key to exit...");
118 Console.ReadLine();
119
120 }
```

The output window on the right shows the following text:

```
Enter the any string: sfhSDF#@3lsbdhf$
Original String: sfhSDF#@3lsbdhf$
Modified String: TGiteg$a42TCEIG%
Press any key to exit...
```

## INTEGER

6. Create a Console Application to accept the side of a square and print the area of the square



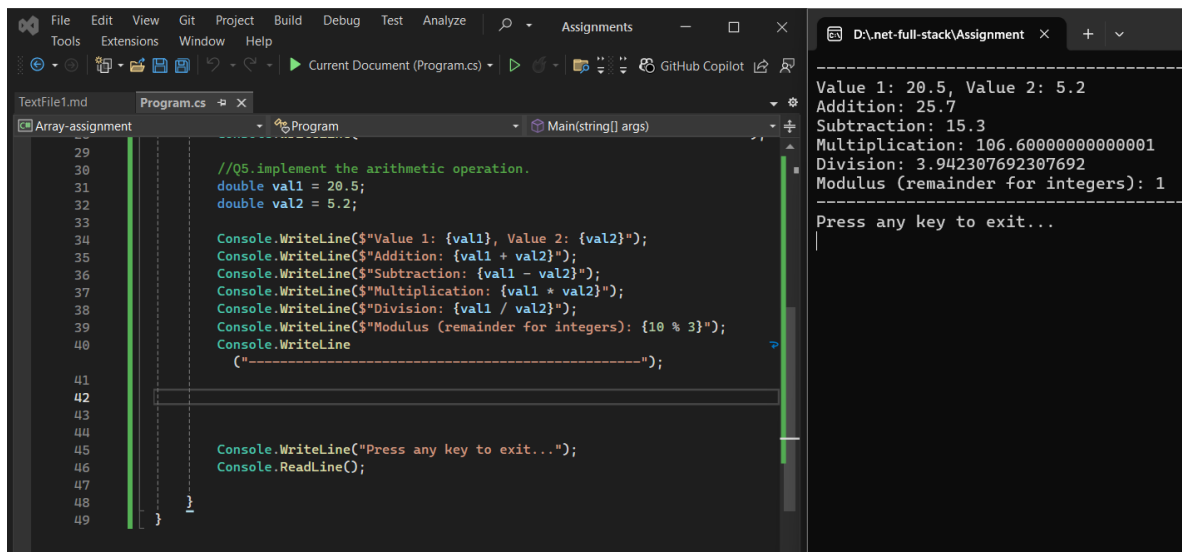
```
using System;

class Program
{
    public static void Main(string[] args)
    {
        // Q4.Create a Console Application to accept the side of a square and
        // print the area of the square
        Console.WriteLine("Enter the side of a Square: ");
        if(double.TryParse(Console.ReadLine(), out double side))
        {
            if(side >= 0)
            {
                double area = side * side;
                Console.WriteLine($"Area of a Square {area}");
            }
            else
            {
                Console.WriteLine("Enter Positive Number!");
            }
        }
        else
        {
            Console.WriteLine("Enter Valid Number!");
        }

        Console.WriteLine("Press any key to exit...");
        Console.ReadLine();
    }
}
```

Enter the side of a Square: 83  
Area of a Square 6889  
Press any key to exit...

7. implement the arithmetic operation.



```
//Q5.implement the arithmetic operation.
double val1 = 20.5;
double val2 = 5.2;

Console.WriteLine($"Value 1: {val1}, Value 2: {val2}");
Console.WriteLine($"Addition: {val1 + val2}");
Console.WriteLine($"Subtraction: {val1 - val2}");
Console.WriteLine($"Multiplication: {val1 * val2}");
Console.WriteLine($"Division: {val1 / val2}");
Console.WriteLine($"Modulus (remainder for integers): {10 % 3}");
Console.WriteLine("-----");

Console.WriteLine("Press any key to exit...");
Console.ReadLine();
```

Value 1: 20.5, Value 2: 5.2  
Addition: 25.7  
Subtraction: 15.3  
Multiplication: 106.60000000000001  
Division: 3.942307692307692  
Modulus (remainder for integers): 1  
-----  
Press any key to exit...

8. .Write a simple program that accepts two numbers num1 and num2.

Assign num1 to num2 in two scenario:

- i)Assign num1 to num2 by pre-incrementing num1 and observe the output
- ii)Assign num1 to num2 by post-incrementing num1 and observe the output
- iii) Swap both value

The screenshot shows a Visual Studio IDE with a C# program. The code is as follows:

```
//Write a simple program that accepts two numbers num1 and num2.
//Assign num1 to num2 in two scenario:
//i)Assign num1 to num2 by pre-incrementing num1 and observe the output
//ii)Assign num1 to num2 by post-incrementing num1 and observe the output
//iii) Swap both value
int num1 = 5;
int num2;
Console.WriteLine($"Initial values for pre-increment: num1 = {num1}");
num2 = ++num1; // num1 is incremented first, then assigned to num2
Console.WriteLine($"After pre-increment: num1 = {num1}, num2 = {num2}"); // num1=6, num2=6

int num3 = 5;
int num4;
Console.WriteLine($"Initial values for post-increment: num1 = {num3}");
num4 = num3++; // num3 is assigned to num4 first, then incremented
Console.WriteLine($"After post-increment: num1 = {num3}, num2 = {num4}"); // num1=6, num2=5

// iii) Swap both value
Console.WriteLine("\n--- Swapping Values ---");
int swapNum1 = 10;
int swapNum2 = 20;
Console.WriteLine($"Before swap: swapNum1 = {swapNum1}, swapNum2 = {swapNum2}");

int temp = swapNum1;
swapNum1 = swapNum2;
swapNum2 = temp;
Console.WriteLine($"After swap (using temp): swapNum1 = {swapNum1}, swapNum2 = {swapNum2}");
```

The output window on the right shows the following results:

```
Initial values for pre-increment: num1 = 5
After pre-increment: num1 = 6, num2 = 6
Initial values for post-increment: num1 = 5
After post-increment: num1 = 6, num2 = 5

--- Swapping Values ---
Before swap: swapNum1 = 10, swapNum2 = 20
After swap (using temp): swapNum1 = 20, swapNum2 = 10

Press any key to exit...
```

9. Use switch statement and perform some task.

The screenshot shows a Visual Studio IDE with a C# program using a switch statement. The code is as follows:

```
public static void Main(string[] args)
{
    //Q7. Use switch statement and perform some task.
    Console.WriteLine("Enter choice in number: ");
    if(int.TryParse(Console.ReadLine(), out int choice))
    {
        switch (choice)
        {
            case 1:
                Console.WriteLine("You chose One!");
                break;
            case 2:
                Console.WriteLine("You chose Two!");
                break;
            case 3:
                Console.WriteLine("You chose Three!");
                break;
            default:
                Console.WriteLine("Choice out of range (1-3).");
                break;
        }
    }
    else
    {
        Console.WriteLine("Invalid Choice!");
    }

    Console.WriteLine("-----");
    Console.WriteLine("Press Any Key to exit...");
    Console.ReadLine();
}
```

The output window on the right shows the following results:

```
Enter choice in number: 2
You chose Two!

Press Any Key to exit...
```

10. write a program to accept a number N and print whether it is positive , negative or zero.

```
File Edit View Git Project Build Debug Test Analyze
Tools Extensions Window Help
Current Document (Program.cs)
Process: [13436] Conditionals-Assignment
Program.cs
32 Console.WriteLine("-----");
33
34 Console.WriteLine("--- Conditionals ---");
35 //1) write a program to accept a number N and print whether it is
36 positive , negative or zero.
37 Console.WriteLine("Enter any number: ");
38 if(int.TryParse(Console.ReadLine(), out int value))
39 {
40     if(value < 0)
41     {
42         Console.WriteLine($"{value} is negative.");
43     }
44     else if(value > 0)
45     {
46         Console.WriteLine($"{value} is positive.");
47     }
48     else
49     {
50         Console.WriteLine($"{value} is Zero.");
51     }
52 }
53
54 Console.WriteLine("Press Any Key to exit...");
55 Console.ReadLine();
56
57
58
59
D:\net-full-stack\Assignment
--- Conditionals ---
Enter any number: -3
-3 is negative.
Press Any Key to exit...
```

11. Write a program to accept two number and print greater value of two

```
File Edit View Git Project Build Debug Test Analyze
Tools Extensions Window Help
Current Document (Program.cs)
D:\net-full-stack\Assignment
Program.cs
56
57 //2)Write a program to accept two number and print greater
58 value of two
59 Console.WriteLine("Enter first number: ");
60 if(float.TryParse(Console.ReadLine(), out float num1))
61 {
62     Console.WriteLine("Enter Second Number: ");
63     if(float.TryParse(Console.ReadLine(), out float num2)){
64         if(num1 > num2)
65         {
66             Console.WriteLine($"{num1} is greater");
67         }
68         else
69         {
70             Console.WriteLine($"{num2} is greater");
71         }
72     }
73 }
D:\net-full-stack\Assignment
--- Conditionals ---
Enter first number: 1
Enter Second Number: 2
1 is greater
```

12. write a program to accept a number N and print whether the number is Even or Odd

```
File Edit View Git Project Build Debug Test Analyze
Tools Extensions Window Help
Current Document (Program.cs)
D:\net-full-stack\Assignment
Program.cs
73
74 // 3) write a program to accept a number N and print whether
75 the number is Even or Odd
76 Console.WriteLine("Enter an integer to check Even/Odd: ");
77 if (int.TryParse(Console.ReadLine(), out int numEO))
78 {
79     if (numEO % 2 == 0)
80     {
81         Console.WriteLine($"{numEO} is Even.");
82     }
83     else
84     {
85         Console.WriteLine($"{numEO} is Odd.");
86     }
87 }
88 else
89 {
90     Console.WriteLine("Invalid input. Please enter an integer.");
91 }
D:\net-full-stack\Assignment
--- Conditionals ---
Enter an integer to check Even/Odd: 342956
342956 is Even.
Press Any Key to exit...
```

### 13. Return last digit of the given number.

The screenshot shows a Visual Studio IDE with a C# program in `Program.cs`. The program prompts the user to enter an integer to find its last digit. It uses `int.TryParse` to validate the input. If the input is a valid integer, it calculates the last digit using `Math.Abs(numberLastDigit % 10)` and prints the result. If the input is invalid, it prompts the user to enter a valid integer. The output window shows the program's execution for the input `-2334`, resulting in the last digit `4`.

```
//Console.WriteLine("\n--- Q9: Return last digit ---");
// Q9.Return last digit of the given number.
Console.WriteLine("Enter an integer to find its last digit: ");
if (int.TryParse(Console.ReadLine(), out int
numberLastDigit))
{
    int lastDigit = Math.Abs(numberLastDigit % 10); // Use
    Math.Abs to handle negative numbers correctly
    Console.WriteLine($"The last digit of {numberLastDigit}
    is: {lastDigit}");
}
else
{
    Console.WriteLine("Invalid input. Please enter an
    integer.");
}
```

### 14. Return second last digit of the given number.

The screenshot shows a Visual Studio IDE with a C# program in `Program.cs`. The program prompts the user to enter an integer to find its second last digit. It uses `int.TryParse` to validate the input. If the input is a valid integer, it checks if the number has at least two digits. If not, it prompts the user to enter a number with at least two digits. If the input is valid and has at least two digits, it calculates the second last digit using `Math.Abs((numberSecondLastDigit / 10) % 10)` and prints the result. If the input is invalid, it prompts the user to enter a valid integer. The output window shows the program's execution for the input `-82098`, resulting in the second last digit `9`.

```
// Q10.Return second last digit of the given number.
Console.WriteLine("Enter an integer to find its second last
digit: ");
if (int.TryParse(Console.ReadLine(), out int
numberSecondLastDigit))
{
    if (Math.Abs(numberSecondLastDigit) < 10) // If number
    has less than 2 digits
    {
        Console.WriteLine($"({numberSecondLastDigit}) does not
        have a second last digit.");
    }
    else
    {
        int secondLastDigit = Math.Abs
        ((numberSecondLastDigit / 10) % 10);
        Console.WriteLine($"The second last digit of
        {numberSecondLastDigit} is: {secondLastDigit}");
    }
}
else
{
    Console.WriteLine("Invalid input. Please enter an
    integer.");
}
```

The screenshot shows two Visual Studio IDE windows. The left window displays a C# program in `Program.cs` that prints all numbers from 1 to 100. The right window displays a C# program in `Program.cs` that prints alternate numbers from 1 to 100 (1, 3, 5, 7, 9, ...). Both programs use `for` loops and `Console.WriteLine` to output the numbers. The output windows show the results of the programs: the first prints all numbers from 1 to 100, and the second prints alternate numbers from 1 to 100.

```
class Program {
    public static void Main(string[] args)
    {
        // 1)Write a program to print all number from 1 to 100
        Console.WriteLine("print all number from 1 to 100");
        for(int i = 1; i <= 100; i++)
        {
            if (i < 100)
            {
                Console.WriteLine(i + ",");
            }
            else
            {
                Console.WriteLine(i);
            }
        }
        Console.WriteLine("\nPress any key to exit...");
        Console.ReadLine();
    }
}
```

```
//2)Write a program to print alternate number from 1 to 100
for(int i = 1; i < 100; i+=2)
{
    if(i < 99)
    {
        Console.WriteLine(i + ",");
    }
    else
    {
        Console.WriteLine(i);
    }
}
Console.WriteLine("\nPress any key to exit...");
Console.ReadLine();
```

15. Write a program to print all number from 1 to 100
16. Write a program to print alternate number from 1 to 99
17. Write a program to print all numbers backward from 100 to 1 i.e.100,99,98....
18. Write a program to print all numbers backward from 100 to 1 by skipping 2 number i.e.100,97,94....

```
//3)Write a program to print all numbers backward from
100 to 1 i.e.100,99,98....
for(int i = 100; i >= 1; i--)
{
    Console.Write(i);
    if(i > 1)
    {
        Console.Write(", ");
    }
}

Console.WriteLine("\nPress any key to exit...");
Console.ReadLine();
```

```
//4)Write a program to print all numbers
backward from 100 to 1 by skipping 2
number i.e.100,97,94....Console.WriteLine
("\nPress any key to exit...");

for(int i = 100; i > 0; i-=3)
{
    Console.Write(i);
    if(i -2> 0)
    {
        Console.Write(", ");
    }
}

Console.ReadLine();
```

3, 82, 81, 80, 79, 78, 77, 76, 75, 74, 73, 72, 71, 70, 69, 68, 67, 66, 65, 64, 63, 62, 61, 60, 59, 58, 57, 56, 55, 54, 53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38, 37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1  
Press any key to exit...

100, 97, 94, 91, 88, 85, 82, 79, 76, 73, 70, 67, 64, 61, 58, 55, 52, 49, 46, 43, 40, 37, 34, 31, 28, 25, 22, 19, 16, 13, 10, 7, 4, 1

## Array ArrayList

19. create an array and take input from user

```
--- ARRAY AND ARRAYLIST BASED QUESTIONS ---
Enter the Size of the array: 5
Enter 5 Elements of array:
Element 1: 23
Element 2: 45
Element 3: 86
Enter valid number!
Element 3: 2
Element 4: 89
Element 5: -45
Your Array Elements are: 23 45 2 89 -45
Press any key to exit...
```

```
Console.WriteLine("\n--- ARRAY AND ARRAYLIST BASED QUESTIONS ---");
//1. create an array and take input from user
int arrSize;
Console.Write("Enter the Size of the array: ");
if (int.TryParse(Console.ReadLine(), out arrSize))
{
    Console.WriteLine("Array size must be a positive integer.");
    return;
}
int[] myArr = new int[arrSize];
Console.WriteLine($"Enter {arrSize} Elements of array: ");
for (int i = 0; i < arrSize; i++)
{
    bool isValid = false;
    while (!isValid)
    {
        Console.Write($"Element {i + 1}: ");
        if (int.TryParse(Console.ReadLine(), out int elem))
        {
            myArr[i] = elem;
            isValid = true;
        }
        else
        {
            Console.WriteLine("Enter valid number!");
        }
    }
}
Console.Write("Your Array Elements are: ");
foreach (int i in myArr)
{
    Console.Write(i + " ");
}
Console.WriteLine();
else
{
    Console.WriteLine("Invalid input. Please enter a valid number!");
}
Console.WriteLine("\nPress any key to exit...");
```

20. Write a program that finds a number of element in an array with out using built in function.



The screenshot shows a Visual Studio window with a C# file named `Program.cs`. The code is as follows:

```

120
121 //Q12. Write a program that finds a number
122 //of element in an array with out using
123 //built in function.
124 int[] myArr2 = { 23, 12, -47, 0, 23, 85 };
125 int count = 0;
126 foreach(int i in myArr2)
127 {
128     count++;
129 }
130 Console.WriteLine($"Lenth of Array: {count}");
131 Console.WriteLine("\n");
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

The console output shows:

```

--- ARRAY AND ARRAYLIST BASED QUESTIONS ---
Lenth of Array: 6
Press any key to exit...

```

Q13. Write a program to accept 10 integers to an array and perform the below action:

- 1) print the element in descending order
- 2) find the min value, max value entered
- 3) Print the sum we get after adding all the number in the array

The screenshot shows a Visual Studio window with a C# file named `Program.cs`. The code is as follows:

```

111
112 //Q13. Write a program to accept 10 integers to an array
113 //and perform the below action:
114 //1) print the element in descending order
115 //2) find the min value, max value entered
116 //3) Print the sum we get after adding all the number in
117 //the array
118 int sizeOfMyArr3 = 10;
119 int[] myArr3 = new int[sizeOfMyArr3];
120 Console.WriteLine($"Enter {sizeOfMyArr3} Elements of
121 Array: ");
122 for(int i = 0; i < 10; i++)
123 {
124     bool isValid = false;
125     while (!isValid)
126     {
127         Console.WriteLine($"Enter Element-{i + 1}: ");
128         if (int.TryParse(Console.ReadLine(), out int el))
129         {
130             myArr3[i] = el;
131             isValid = true;
132         }
133         else
134         {
135             Console.WriteLine("Enter Valid Integer!");
136         }
137     }
138 }
139 Console.WriteLine("\nOriginal Q13 Array: " + string.Join
140 (" ", myArr3));
141 Console.WriteLine("Descending order of Array: ");
142 //1) print the element in descending order
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

The console output shows:

```

--- ARRAY AND ARRAYLIST BASED QUESTIONS ---
Enter 10 Elements of Array:
Enter Element-1: 1
Enter Element-2: -7
Enter Element-3: 24
Enter Element-4: 78
Enter Element-5: -5
Enter Element-6:
Enter Valid Integer!
Enter Element-6: 45
Enter Element-7: 8
Enter Element-8: 78
Enter Element-9: 6
Enter Element-10: +75

Original Q13 Array: 1, -7, 24, 78, -5, 45, 8, 78, 6, 75
Descending order of Array:
78, 78, 75, 45, 24, 8, 6, 1, -5, -7

Min Value : -7
Max Value : 78

Sum of the array is 303
Press any key to exit...

```

The screenshot shows a Visual Studio window with a C# file named `Program.cs`. The code is as follows:

```

158
159 Console.WriteLine($"Descending order of Array: ");
160 //1) print the element in descending order
161 for (int i = 0; i < myArr3.Length; i++){
162     for(int j = 0; j < myArr3.Length - 1 - i; j++){
163         if (myArr3[j] < myArr3[j + 1]){
164             int temp = myArr3[j];
165             myArr3[j] = myArr3[j + 1];
166             myArr3[j + 1] = temp;
167         }
168     }
169 }
170 Console.WriteLine(String.Join(" ", myArr3));
171 //2) find the min value, max value entered
172 int minVal = myArr3[0];
173 int maxVal = myArr3[0];
174 for(int i = 0; i < myArr3.Length; i++){
175     if (myArr3[i] < minVal){
176         minVal = myArr3[i];
177     } else if (myArr3[i] > maxVal){
178         maxVal = myArr3[i];
179     }
180 }
181 Console.WriteLine($"Min Value : {minVal}");
182 Console.WriteLine($"Max Value : {maxVal}");
183
184 //3) Print the sum we get after adding all the number in
185 //the array
186 int sum = 0;
187 foreach(int i in myArr3){
188     sum += i;
189 }
190 Console.WriteLine($"Sum of the array is {sum}");
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

The console output shows:

```

--- ARRAY AND ARRAYLIST BASED QUESTIONS ---
Enter 10 Elements of Array:
Enter Element-1: 1
Enter Element-2: -7
Enter Element-3: 24
Enter Element-4: 78
Enter Element-5: -5
Enter Element-6:
Enter Valid Integer!
Enter Element-6: 45
Enter Element-7: 8
Enter Element-8: 78
Enter Element-9: 6
Enter Element-10: +75

Original Q13 Array: 1, -7, 24, 78, -5, 45, 8, 78, 6, 75
Descending order of Array:
78, 78, 75, 45, 24, 8, 6, 1, -5, -7

Min Value : -7
Max Value : 78

Sum of the array is 303
Press any key to exit...

```