

Data Programming - Problem Set 1

Prepared By: Avhimanhu Sapkota

Student ID: 200547612

Task 1

What data type is each of the following?

5: Integer

5.0: Float

$5 > 1$: Boolean

"5": String

$5 * 2$: Integer

"5" * 2: String

"5" + "2": String

$5 / 2$: Float

$5 \% 2$: Integer

{5, 2, 1}: Array

$5 == 3$: Boolean

Pi (the number): Double

Task 2

A. How many letters are there in 'Supercalifragilisticexpialidocious'?

```
In [ ]: using System;
        namespace Problem_Set_1
        {
            public class Task2
            {
                public static void Main(string[] args)
                {
                    // Assigns the given word to the string variable.
                    string givenWord = "Supercalifragilisticexpialidocious";

                    // Counts the number of letter using Length method and stores in integer variable.
                    int noOfLetters = givenWord.Length;

                    Console.WriteLine($"The total number of letters in {givenWord} is {noOfLetters}");
                }
            }
        }
```

Output:

The total number of letters in Supercalifragilisticexpialidocious is 34

B. Does 'Supercalifragilisticexpialidocious' contain 'ice' as a substring?

```
In [ ]: using System;
        namespace Problem_Set_1
        {
            public class Task2
            {
                public static void Main(string[] args)
                {
                    // Assigns the given word to the string variable.
                    string givenWord = "Supercalifragilisticexpialidocious";

                    // Checks if givenWord contains ice using Contains method and displays the output
                    if (givenWord.Contains("ice"))
                        Console.WriteLine($"Yes, {givenWord} contains ice as substring");
                    else
                        Console.WriteLine($"No, {givenWord} does not contain ice as substring");
                }
            }
        }
```

Output:

Yes, Supercalifragilisticexpialidocious contains ice as substring

C. Which of the following words is the longest: Supercalifragilisticexpialidocious, Honorificabilitudinitatibus, or Bababadalgharaghtakamminarronnkonn?

```

In [ ]: using System;
        namespace Problem_Set_1
        {
            public class Task2
            {
                public static void Main(string[] args)
                {
                    // Assigns all the given words to the string variables respectively.
                    string firstWord = "Supercalifragilisticexpialidocious";
                    string secondWord = "Honorificabilitudinitatibus";
                    string thirdWord = "Bababadalgharaghtakamminarronkonn";

                    // Checks if length of first word is greater than second and third word. If true, displays 1
                    if (firstWord.Length > secondWord.Length && firstWord.Length > thirdWord.Length)
                    {
                        Console.WriteLine($"The Longest word is {firstWord}");
                    }
                    // Compares if length of second word is greater than first and third word. If true, displays
                    else if (secondWord.Length > firstWord.Length && secondWord.Length > thirdWord.Length)
                    {
                        Console.WriteLine($"The Longest word is {secondWord}");
                    }
                    // If both conditions were not true displays third word as longest
                    else
                    {
                        Console.WriteLine($"The Longest word is {thirdWord}");
                    }
                }
            }
        }

```

Output:

The Longest word is Bababadalgharaghtakamminarronkonn

D. Which composer comes first in the dictionary: 'Berlioz', 'Borodin', 'Brian', 'Bartok', 'Bellini', 'Buxtehude', 'Bernstein'. Which one comes last?

```

In [ ]: using System;
        namespace Problem_Set_1
        {
            public class Task2
            {
                public static void Main(string[] args)
                {
                    // Puts all composers in a string array
                    string[] composersArray = { "Berlioz", "Borodin", "Brian", "Bartok", "Bellini", "Buxtehude",

                    // Sorts the provided list of composers within the array using Sort function
                    Array.Sort(composersArray);

                    // Counts the number of strings in the array and decreased it by 1 to identify last index of
                    int lastIndex = composersArray.Length - 1;

                    // Displays the composers who came first and last on the basis of sorting
                    Console.WriteLine($"The composer who came first after Sorting is {composersArray[0]}");
                    Console.WriteLine($"The composer who was last after Sorting is {composersArray[lastIndex]}")
                }
            }
        }

```

Output:

The composer who came first after Sorting is Bartok
 The composer who was last after Sorting is Buxtehude

Task 3

Implement function `triangleArea(a,b,c)` that takes as input the lengths of the 3 sides of a triangle and returns the area of the triangle.

```

In [ ]: using System;
        namespace Problem_Set_1
        {
            public class Task3
            {
                public static void Main(string[] args)
                {
                    // Takes length of each side of the triangle and stores it in double variables.
                    Console.Write ("Enter length of first side of the triangle: ");
                    double firstSide = Double.Parse(Console.ReadLine());
                    Console.Write ("Enter length of second side of the triangle: ");
                    double secondSide = Double.Parse(Console.ReadLine());
                    Console.Write ("Enter length of third side of the triangle: ");
                    double thirdSide = Double.Parse(Console.ReadLine());

                    // calls function triangleArea which takes length of sides as parameters to calculate the area
                    double area = triangleArea(firstSide, secondSide, thirdSide);

                    // Displays the area of triangle
                    Console.WriteLine($"The area of Triange with sides {firstSide}, {secondSide} and {thirdSide} is {area}");
                }

                // Function that calculates area of triangle and returns the calculated value.
                static double triangleArea(double a, double b, double c)
                {
                    // Calculates value of s with respect to given formula.
                    double s = (a + b + c) / 2;

                    // Calculates area of triangle with respect to given formula.
                    double area = Math.Sqrt(s * (s - a) * (s - b) * (s - c));

                    return area;
                }
            }
        }

```

Output:

Enter length of first side of the triangle: 2

Enter length of second side of the triangle: 2

Task 4

Write a program in C# Sharp to separate odd and even integers in separate arrays.


```

In [ ]: using System;
        namespace Problem_Set_1
        {
            public class Task4
            {
                public static void Main(string[] args)
                {
                    // Creates array to store for even and odd input from user.
                    // Here, Input limit is 10 numbers as array needs to be predefined.
                    int[] oddArray = new int[10];
                    int[] evenArray = new int[10];

                    // Integer index for even and odd array
                    int evenIndex = 0, oddIndex = 0;

                    // Prompts user for total number of elements desired and stores it in integer variable after
                    Console.WriteLine("Enter the number of elements to be stored in the array: ");
                    int numElements = Convert.ToInt32(Console.ReadLine());

                    Console.WriteLine($"Enter {numElements} elements below: ");
                    // Loop that prompts user for numbers, check it is odd or even and stores in different array
                    for (int i = 0; i < numElements; i++)
                    {
                        Console.WriteLine($"Element - {i}: ");
                        int inputNumber = Convert.ToInt32(Console.ReadLine());

                        // Checks if input number is even by dividing it with 2, if yes stores it in even array
                        if (inputNumber % 2 == 0)
                        {
                            evenArray[evenIndex] = inputNumber;
                            evenIndex++;
                        }
                        else // If not divisible by 2, stores the number in odd array and increases the index
                        {
                            oddArray[oddIndex] = inputNumber;
                            oddIndex++;
                        }
                    }

                    // Iterates through the array and prints even numbers
                    Console.WriteLine("\nThe elements in even array are: ");
                    for (int i = 0; i < evenIndex; i++)
                        Console.Write(evenArray[i] + " ");
                }
            }
        }

```

```

        // Iterates through the array and prints odd numbers
        Console.WriteLine("\nThe elements in even array are: ");
        for (int i = 0; i < oddIndex; i++)
            Console.Write(oddArray[i] + " ");
    }
}

```

Output:

Enter the number of elements to be stored in the array: 5

Enter 5 elements below:

Element - 0 : 25

Element - 1 : 47

Element - 2 : 42

Element - 3 : 56

Element - 4 : 32

The elements in even array are:

42 56 32

The elements in odd array are:

25 47

Task 5

A. Write a function `inside(x,y,x1,y1,x2,y2)` that returns True or False depending on whether the point(x,y) lies in the rectangle with lower left corner (x1,y1) and upper right corner (x2,y2).

B. Use function `inside()` from part a. to write an expression that tests whether the point (1,1) lies in both of the following rectangles: one with lower left corner (0.3, 0.5) and upper right corner (1.1, 0.7) and the other with lower left corner (0.5, 0.2) and upper right corner (1.1, 2)


```

In [ ]: using System;
        namespace Problem_Set_1
        {
            public class Task4
            {
                public static void Main(string[] args)
                {
                    // Test case scenario for Part - A
                    // Calls the Function inside and stores returned value in boolean variable
                    bool checkCoordinate1 = inside(1, 1, 0, 0, 2, 3);
                    bool checkCoordinate2 = inside(-1, -1, 0, 0, 2, 3);
                    Console.WriteLine("Does the co-ordinate (1,1) lie inside the rectangle (0,0) to (2,3)?\n" +
                        Console.WriteLine("\nDoes the co-ordinate (-1,-1) lie inside the rectangle (0,0) to (2,3)?\n");

                    // Test case scenario for Part - B

                    // Part B - Calls the function with the provided coordinates from the question
                    bool checkRectangle1 = inside(1, 1, 0.3, 0.5, 1.1, 0.7);
                    bool checkRectangle2 = inside(1, 1, 0.5, 0.2, 1.1, 2);

                    // Checks if (1,1) lies under both rectangles with the provided coordinates and displays it
                    Console.WriteLine("\nDoes the point (1,1) lie within rectangle 1: (0.3, 0.5) and (1.1, 0.7)");

                    if (checkRectangle1 && checkRectangle2)
                        Console.WriteLine("Yes, (1,1) lies in both of the rectangles at once");
                    else
                        Console.WriteLine("No, (1,1) doesnot lie in both of the rectangles at once");
                }

                // Part A - Function that checks if the point (x,y) lies in the rectangle given the lower left a
                static bool inside(double x, double y, double x1, double y1, double x2, double y2)
                {
                    // Checks if the coordinates lie within the coordinates of rectangle
                    if (x >= x1 && x <= x2 && y >= y1 && y <= y2)
                        return true;
                    else
                        return false;
                }
            }
        }

```

```
}
```

Output:

Does the co-ordinate (1,1) lie inside the rectangle (0,0) to (2,3)?

True

Does the co-ordinate (-1,-1) lie inside the rectangle (0,0) to (2,3)?

False

Does the point (1,1) lie within rectangle 1: (0.3, 0.5) and (1.1, 0.7) and rectangle 2: (0.5, 0.2) and (1.1, 2)?

No, (1,1) doesnot lie in both of the rectangles at once

Task 6

Write a function `pig()` that takes a word (i.e., a string) as input and returns its pig-Latin form. Your function should still work if the input word contains upper case characters. Your output should always be lower case however.

```

In [16]: # Function Pig that converts words to pig-Latin Format.
# Takes string as parameter and returns converted string.
def pig(word):
    # List of vowel letters
    vowels = ['a', 'e', 'i', 'o', 'u']
    # Converts the word to lowercase
    word = word.lower()

    # Checks if first letter has vowel and converts on the basis of given technique
    if word[0] in vowels:
        return word + 'way'
    else:
        return word[1:] + word[0] + 'ay'

# Takes input from User and stores it as string
word = input("Enter a string to turn it into pig-Latin: ")

# Calls the function Pig, gets the converted string and displays it.
print(f"The pig-Latin of {word} is {pig(word)}");

```

```

Enter a string to turn it into pig-Latin:Yatch
The pig-Latin of Yatch is atchyay

```

Task 7

File `bloodtype1.txt` records blood-types of patients (A, B, AB, O or OO) at a clinic. Write a function `bldcount()` that reads the file with name `name` and reports (i.e., prints) how many patients there are in each bloodtype

```

In [34]: # Function that counts the patients of each blood group from the file
def bldcount(filename):

    # Dictionary to store the number of patient of each blood group type
    bloodCounts = {'A': 0, 'B': 0, 'AB': 0, 'O': 0, 'OO': 0}

    # Opens the file using open function in read-only mode.
    with open(filename, 'r') as file:
        # Reads and splits the data in the file using space
        fileData = file.read().split()

        # Loop that iterates to read each blood type mentioned in file
        for bloodType in fileData:
            # Checks each blood type and updates the count in the dictionary
            if bloodType in bloodCounts:
                bloodCounts[bloodType] = bloodCounts[bloodType] + 1

        # Print the blood type counts
        for bloodType, count in bloodCounts.items():
            print(f"There are {count} patients of blood group {bloodType}.")

# Calls the function with the filename
bldcount('bloodtype.txt')

```

There are 15 patients of blood group A.
 There are 1 patients of blood group B.
 There are 13 patients of blood group AB.
 There are 15 patients of blood group O.
 There are 0 patients of blood group OO.

Task 8

Write a function `curconv()` that takes as input:

1. a currency represented using a string (e.g., 'JPY' for the Japanese Yen or 'EUR' for the Euro)
2. an amount

and then converts and returns the amount in US dollars.

The currency rates you will need are stored in file `currencies.txt`

```

In [40]: # Function that converts the currency to US dollars
# Takes file that has conversion rate as parameter 1 and amount
def curconv(currency, amount):

    # A dictionary to store conversion rates from the file
    conversionRate = {}

    # Opens the file using open function in read-only mode.
    with open("currencies.txt", 'r') as file:
        for eachLine in file:
            # Splits the data of each line using tab space and stores as array
            splittedData = eachLine.split("\t")

            # Stores the splitted data into conversionRate dictionary
            conversionRate[splittedData[0]] = splittedData[1]

    # Converts the entered amount of specific currency to USD equivalent
    usdEquivalent = float(conversionRate[currency]) * amount
    print(f"The USD equivalent of {currency} {amount} is USD {usdEquivalent}")

# Call the Function with desired Currency and
curconv("EUR", 100)
curconv("JPY", 100)
curconv("NOK", 100)
curconv("SGD", 100)

```

```

The USD equivalent of EUR 100 is USD 122.96544
The USD equivalent of JPY 100 is USD 1.241401
The USD equivalent of NOK 100 is USD 16.77063
The USD equivalent of SGD 100 is USD 78.8871

```

Task 9

Each of the following will cause an exception (an error). Identify what type of exception each will cause.

Trying to add incompatible variables, as in adding 6 + 'a' gives the following error:

```
TypeError: unsupported operand type(s) for +: 'int' and 'str'
```



```
In [48]: 6 + 'a'
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[48], line 1  
----> 1 6 + 'a'  
  
TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

Referring to the 12th item of a list that has only 10 items gives the following error:

IndexError: list index out of range

```
In [50]: testList = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
print(testList[12])
```

```
-----  
IndexError                                Traceback (most recent call last)  
Cell In[50], line 2  
      1 testList = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
----> 2 print(testList[12])  
  
IndexError: list index out of range
```

Using a value that is out of range for a function's input, such as calling `math.sqrt(-1.0)` gives the following error:

ValueError: math domain error

```
In [52]: import math
result = math.sqrt(-1.0)
print(result)
```

```
-----
ValueError                                Traceback (most recent call last)
Cell In[52], line 2
      1 import math
----> 2 result = math.sqrt(-1.0)
      3 print(result)

ValueError: math domain error
```

Using an undeclared variable, such as print(x) when x has not been defined gives the following error:

NameError: name 'x' is not defined

```
In [53]: print(x)
```

```
-----
NameError                                Traceback (most recent call last)
Cell In[53], line 1
----> 1 print(x)

NameError: name 'x' is not defined
```

Trying to open a file that does not exist, such as mistyping the file name or looking in the wrong directory gives the following error:

FileNotFoundError: [Errno 2] No such file or directory: 'currencies1.txt'

```
In [59]: with open("currencies1.txt") as file:
        for eachLine in file:
            splittedData = eachLine.split("\t")
            print(splittedData)
```

```
-----
FileNotFoundError                                Traceback (most recent call last)
Cell In[59], line 1
----> 1 with open("currencies1.txt") as file:
      2     for eachLine in file:
      3         splittedData = eachLine.split("\t")

File ~/anaconda3/lib/python3.10/site-packages/IPython/core/interactiveshell.py:282, in _modified_open
(file, *args, **kwargs)
    275 if file in {0, 1, 2}:
    276     raise ValueError(
    277         f"IPython won't let you open fd={file} by default "
    278         "as it is likely to crash IPython. If you know what you are doing, "
    279         "you can use builtins' open."
    280     )
--> 282 return io_open(file, *args, **kwargs)

FileNotFoundError: [Errno 2] No such file or directory: 'currencies1.txt'
```

Task 10

Assume that the string `letters` is already defined as `'abcdefghijklmnopqrstuvwxyz'`. Write a function called `frequencies()` that takes a string as its only parameter, and returns a list of integers, showing the number of times each character appears in the text. Your function may ignore any characters that are not in letters.

```

In [70]: # Function that counts letters and returns their frequencies
def frequencies(stringText):
    # declares string letters with every letter of the alphabet
    letters = 'abcdefghijklmnopqrstuvwxyz'

    # List to store the frequency of letters
    totalCount = []

    # Loop that iterates through each letter from the a-z
    for eachLetter in letters:
        #Counts the frequency of each letter in the given text and stores it
        letterFrequency = stringText.count(eachLetter)
        # Appends frequency of each letter to the list
        totalCount.append(letterFrequency)
    # Returns the list of frequency of letters
    return totalCount

# Sample texts to test the code
sampleText1 = "The quick red fox got bored and went home."
sampleText2 = "apple"

# Calls the function frequencies and prints the returned value
print(frequencies(sampleText1))
print(frequencies(sampleText2))

[1, 1, 1, 3, 5, 1, 1, 2, 1, 0, 1, 0, 1, 2, 4, 0, 1, 2, 0, 2, 1, 0, 1, 1, 0, 0]
[1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0]

```

In []: