Netaji Subhas University of Technology

Practical File
On
Computer Programming

Submitted To:

Mrs Anamika Rajput

Submitted By:

Nipul Rajput
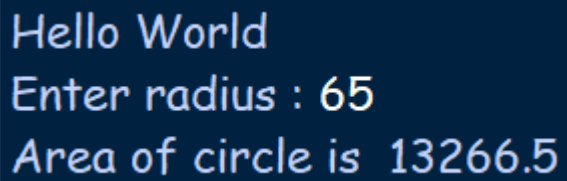
2024UCA1883

CSAI – 2

Semester – 1

INDEX

Sno.	Program	Remarks
1	Write a Python program to print "Hello, World!"; Write a Python program to calculate the area of a circle given the radius.	
2	Write a Python program to check if a number is even or odd.; Implement a simple calculator using conditional statements; Write a Python program to print the Fibonacci series using a for loop.	
3	Implement a function to check if a given string is a palindrome.; Perform various operations on lists (e.g., sorting, slicing).; Use dictionaries to store and retrieve student grades.	
4	Create a class to represent a book with attributes and methods.; Implement inheritance by creating subclasses for different types of books.; Write a generator function to generate the Fibonacci series.	
5	Use lambda functions, map, and filter to perform operations on a list.; Create a module that contains functions for mathematical operations.; Import and use functions from external packages (e.g., math, random).	
6	Create and manipulate NumPy arrays.; Perform basic operations and indexing on arrays.	
7	Implement string operations (e.g., concatenation, slicing).; Use regular expressions to validate email addresses.	
8	Read data from a text file and perform operations.; Handle exceptions for file operations and input validation.	

PROGRAM 1

Write a Python program to print "Hello, World!"; Write a Python program to calculate the area of a circle given the radius.

print('Hello World')

n=int(input("Enter radius : "))

a=3.14*n*n
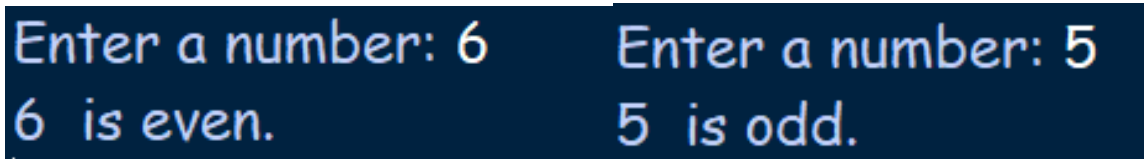
print("Area of circle is ",a)

```
Hello World
Enter radius : 65
Area of circle is  13266.5
```

PROGRAM 2

Write a Python program to check if a number is even or odd.; Implement a simple calculator using conditional statements; Write a Python program to print the Fibonacci series using a for loop.

To find if even or odd:

```python
def check_even_odd(number):
    if number % 2 == 0:
        print(number," is even.")
    else:
        print(number," is odd.")
num = int(input("Enter a number: "))
check_even_odd(num)
```

```
Enter a number: 6      Enter a number: 5
6  is even.            5  is odd.
```

To implement a calculator:

```python
print("1. Add")
print("2. Subtract")
print("3. Multiply")
print("4. Divide")
while True:
    choice = input("Enter choice (1/2/3/4): ")
    if choice in ('1', '2', '3', '4'):
        num1 = float(input("Enter first number: "))
        num2 = float(input("Enter second number: "))
        if choice == '1':
```

```python
        result = num1 + num2
        print(f"The result is: {result}")
    elif choice == '2':
        result = num1 - num2
        print(f"The result is: {result}")
    elif choice == '3':
        result = num1 * num2
        print(f"The result is: {result}")
    elif choice == '4':
        if num2 == 0:
            print("Error! Division by zero.")
        else:
            result = num1 / num2
            print(f"The result is: {result}")
    else:
        print("Invalid Input")
    next_calculation = input("Do you want to perform another calculation? (yes/no): ")
    if next_calculation.lower() != 'yes':
        break
print("Thank you for using the calculator!")
```

```
1. Add
2. Subtract
3. Multiply
4. Divide
Enter choice (1/2/3/4): 1
Enter first number: 1
Enter second number: 5
The result is: 6.0
Do you want to perform another calculation? (yes/no): yes
Enter choice (1/2/3/4): 2
Enter first number: 3
Enter second number: 4
The result is: -1.0
Do you want to perform another calculation? (yes/no): yes
Enter choice (1/2/3/4): 3
Enter first number: 5
Enter second number: 8
The result is: 40.0
Do you want to perform another calculation? (yes/no): yes
Enter choice (1/2/3/4): 9
Invalid Input
Do you want to perform another calculation? (yes/no): yes
Enter choice (1/2/3/4): 4
Enter first number: 6
Enter second number: 9
The result is: 0.6666666666666666
Do you want to perform another calculation? (yes/no): no
Thank you for using the calculator!
```

Fibonacci series using for loop:

n= int(input("Enter the number of terms: "))

a, b = 0, 1

print("Fibonacci series:")

```
for i in range(n):
    print(a, end=" ")
    a, b = b, a + b
```
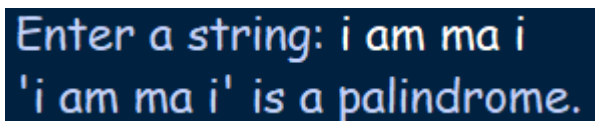
```
Enter the number of terms: 10
Fibonacci series:
0 1 1 2 3 5 8 13 21 34
```

PROGRAM 3

Implement a function to check if a given string is a palindrome.; Perform various operations on lists (e.g., sorting, slicing).; Use dictionaries to store and retrieve student grades.

Function to Check if a Given String is a Palindrome:

```python
def is_palindrome(s):
    s = s.replace(" ", "").lower()
    return s == s[::-1]

string = input("Enter a string: ")
if is_palindrome(string):
    print(f"'{string}' is a palindrome.")
else:
    print(f"'{string}' is not a palindrome.")
```

```
Enter a string: i am ma i
'i am ma i' is a palindrome.
```

Various Operations on Lists:

```python
numbers = [16,23,13,84,65,36,97]
sorted_numbers = sorted(numbers)
reverse_sorted_numbers = sorted(numbers, reverse=True)
first_three = numbers[:3]
last_three = numbers[-3:]
middle_elements = numbers[1:5]
print("Original list:", numbers)
print("Sorted list (ascending):", sorted_numbers)
print("Sorted list (descending):", reverse_sorted_numbers)
print("First three elements:", first_three)
print("Last three elements:", last_three)
```

print("Middle elements (index 1 to 4):", middle_elements)

```
Original list: [16, 23, 13, 84, 65, 36, 97]
Sorted list (ascending): [13, 16, 23, 36, 65, 84, 97]
Sorted list (descending): [97, 84, 65, 36, 23, 16, 13]
First three elements: [16, 23, 13]
Last three elements: [65, 36, 97]
Middle elements (index 1 to 4): [23, 13, 84, 65]
```

Using Dictionaries to Store and Retrieve Student Grade:

marks = {"Nipul": 85, "Roshan": 78, "Prankush": 92, "Bhavishya": 88, "Snehil": 76}

stu_name = input("Enter the student's name to get their grade: ")

if stu_name in marks:

   print(f"{stu_name}'s grade is {marks[stu_name]}")

else:

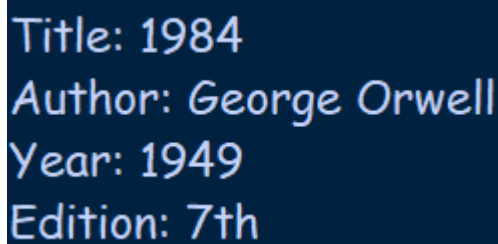   print(f"No grade found for {stu_name}")

```
Enter the student's name to get their grade: Prankush
Prankush's grade is 92
```

PROGRAM 4

Create a class to represent a book with attributes and methods.; Implement inheritance by creating subclasses for different types of books.; Write a generator function to generate the Fibonacci series.

Create a Class to Represent a Book with Attributes and Methods:

```python
class Book:

    def __init__(self, title, author, year,edition):
        self.title = title
        self.author = author
        self.year = year
        self.edition=edition

    def display_info(self):
        print(f"Title: {self.title}")
        print(f"Author: {self.author}")
        print(f"Year: {self.year}")
        print(f"Edition: {self.edition}")

book1 = Book("1984", "George Orwell", 1949,'7th')

book1.display_info()
```

```
Title: 1984
Author: George Orwell
Year: 1949
Edition: 7th
```

Implement Inheritance by Creating Subclasses for Different Types of Books:

```python
class EBook(Book):

    def __init__(self, title, author, year, file_size):
        super().__init__(title, author, year)
```

```python
        self.file_size = file_size
    def display_info(self):
        super().display_info()
        print(f"File Size: {self.file_size} MB")
class AudioBook(Book):
    def __init__(self, title, author, year, duration):
        super().__init__(title, author, year)
        self.duration = duration
    def display_info(self):
        super().display_info()
        print(f"Duration: {self.duration} hours")
ebook = EBook("Digital Fortress", "Dan Brown", 1998, 5)
audiobook = AudioBook("The Hobbit", "J.R.R. Tolkien", 1937, 11.5)
print("EBook Information:")
ebook.display_info()
print("\nAudioBook Information:")
audiobook.display_info()
```

```
EBook Information:
Title: The Guide
Author: R.K Narayan
Year: 2008
Edition: 6
File Size: 3 MB

AudioBook Information:
Title: David Copperfields
Author: Charles Dicken
Year: 1849
Edition: 2
Duration: 4 hours
```

Generator Function to Generate the Fibonacci Series:

```python
def fibonacci_series(n):
    a, b = 0, 1
    for _ in range(n):
        yield a
        a, b = b, a + b

n_terms = int(input("Enter the number of terms in the Fibonacci series: "))
print("Fibonacci Series:")
for number in fibonacci_series(n_terms):
    print(number, end=" ")
```

```
Enter the number of terms in the Fibonacci series: 6
Fibonacci Series:
0 1 1 2 3 5
```

PROGRAM 5

Use lambda functions, map, and filter to perform operations on a list.; Create a module that contains functions for mathematical operations.; Import and use functions from external packages (e.g., math, random).

Use Lambda Functions, map, and filter to Perform Operations on a List:

numbers = [11,22, 63, 24, 15, 46, 77]

squared_numbers = list(map(lambda x: x ** 2, numbers))

print("Squared Numbers:", squared_numbers)

even_numbers = list(filter(lambda x: x % 2 == 0, numbers))

print("Even Numbers:", even_numbers)

doubled_numbers = list(map(lambda x: x * 2, numbers))

print("Doubled Numbers:", doubled_numbers)

```
Squared Numbers: [121, 484, 3969, 576, 225, 2116, 5929]
Even Numbers: [22, 24, 46]
Doubled Numbers: [22, 44, 126, 48, 30, 92, 154]
```
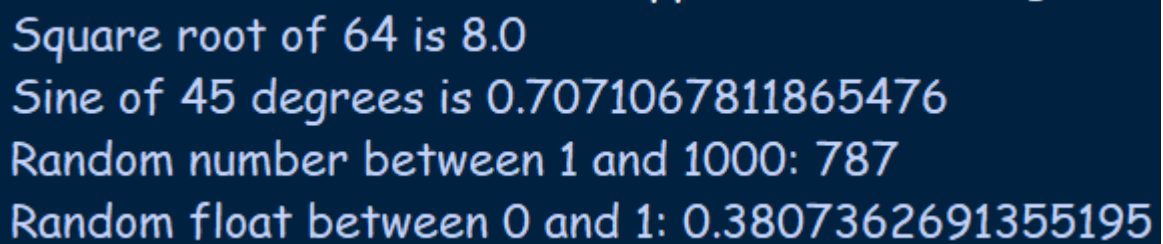
Create a Module that Contains Functions for Mathematical Operations:

import matho as mo

x = 1

y = 35

print(f"Addition of {x} and {y}: {mo.add(x, y)}")

print(f"Subtraction of {x} and {y}: {mo.subtract(x, y)}")

print(f"Multiplication of {x} and {y}: {mo.multiply(x, y)}")

print(f"Division of {x} and {y}: {mo.divide(x, y)}")

```
Addition of 1 and 35: 36
Subtraction of 1 and 35: -34
Multiplication of 1 and 35: 35
Division of 1 and 35: 0.02857142857142857
```

Import and Use Functions from External Packages (math and random):

```python
import math

import random

number = 64

square_root = math.sqrt(number)

print(f"Square root of {number} is {square_root}")

angle = math.radians(45)

sine_value = math.sin(angle)

print(f"Sine of 45 degrees is {sine_value}")

random_number = random.randint(1, 1000)

print(f"Random number between 1 and 1000: {random_number}")

random_float = random.random()

print(f"Random float between 0 and 1: {random_float}")
```

```
Square root of 64 is 8.0
Sine of 45 degrees is 0.7071067811865476
Random number between 1 and 1000: 787
Random float between 0 and 1: 0.3807362691355195
```

PROGRAM 6

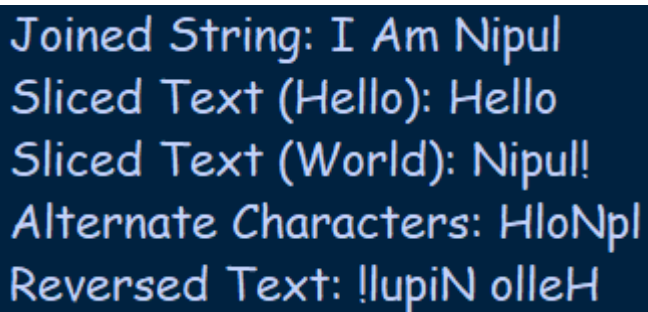Create and manipulate NumPy arrays.; Perform basic operations and indexing on arrays.

```python
import numpy as np

import random

array1 = np.array([1, 2, 3, 4, 5])

print("Array 1:", array1)

array2 = np.arange(0, 10, 2)

print("Array 2:", array2)

array3 = np.array([10, 20, 30, 40, 50])

add_result = array1 + array3

print("Addition Result:", add_result)

sub_result = array3 - array1

print("Subtraction Result:", sub_result)

mul_result = array1 * 2

print("Multiplication Result:", mul_result)

div_result = array3 / array1

print("Division Result:", div_result)

elementwise_mul = array1 * array3

print("Element-wise Multiplication:", elementwise_mul)
```

```
Array 1: [1 2 3 4 5]
Array 2: [0 2 4 6 8]
Addition Result: [11 22 33 44 55]
Subtraction Result: [ 9 18 27 36 45]
Multiplication Result: [ 2  4  6  8 10]
Division Result: [10. 10. 10. 10. 10.]
Element-wise Multiplication: [ 10  40  90 160 250]
```

PROGRAM 7

Implement string operations (e.g., concatenation, slicing).; Use regular expressions to validate email addresses.

```
words = ["I","Am","Nipul"]

sentence = " ".join(words)

print("Joined String:", sentence)

text = "Hello Nipul!"

hello = text[:5]

print("Sliced Text (Hello):", hello)

nipu = text[6:12]

print("Sliced Text (World):", nipu)

alternate_chars = text[::2]

print("Alternate Characters:", alternate_chars)

reversed_text = text[::-1]

print("Reversed Text:", reversed_text)
```

```
Joined String: I Am Nipul
Sliced Text (Hello): Hello
Sliced Text (World): Nipul!
Alternate Characters: HloNpl
Reversed Text: !lupiN olleH
```

PROGRAM 8

Read data from a text file and perform operations.; Handle exceptions for file operations and input validation.

```python
with open('data.txt', 'w') as file:

    file.write("Hello World\n")

    file.write("Python is great\n")

    file.write("Data Science is fascinating\n")

    file.write("File handling in Python\n")

def read_file(file_name):

    try:

        with open(file_name, 'r') as file:

            content = file.readlines()  # Read all lines from the file

            return content

    except FileNotFoundError:

        print(f"Error: The file '{file_name}' does not exist.")

        return []

    except IOError:

        print("Error: An I/O error occurred.")

        return []


def perform_operations(lines):

    if not lines:

        print("No lines to process.")

        return

    line_count = len(lines)

    word_count = sum(len(line.split()) for line in lines)

    print(f"Total number of lines: {line_count}")

    print(f"Total number of words: {word_count}")
```

```python
file_name = input("Enter the name of the file to read (e.g., data.txt): ")
while not file_name:
    file_name = input("Invalid input. Please enter the file name: ")
lines = read_file(file_name)
perform_operations(lines)
```

```
Enter the name of the file to read (e.g., data.txt):
Invalid input. Please enter the file name: data.txt
Total number of lines: 4
Total number of words: 13
```