

**IT Workshop (Python)
LAB MANUAL**

Subject Code: B23-CSE-213

Class: 2nd Year (3rd Semester)

**Bachelor of Technology
in
Computer Engineering**



**Prepared By: Dr. Rama Rani and Mr. Ashish Kharb
Assistant Professor, Department of Computer Engineering**

**DEPARTMENT OF COMPUTER ENGINEERING
STATE INSTITUTE OF ENGINEERING & TECHNOLOGY,
NILOKHERI
(AFFILIATED TO KURUKSHETRA UNIVERSITY, KURUKSHETRA)**

1. Course Purpose and Objectives

The purpose of this laboratory course is to provide basic knowledge of Python programming. Students will gain hands-on experience in implementing fundamental programs, mathematical computations, searching techniques (Linear and Binary), sorting techniques (Selection and Insertion), and matrix multiplication. The objective is to develop algorithmic thinking and problem-solving skills using Python, emphasizing the writing of foundational logic from scratch suitable for beginners.

2. Course Outcomes (CO)

- **CO1:** To study fundamentals of python programming and implement basic programs.
- **CO2:** To implement the searching technique using python.
- **CO3:** To implement sorting techniques using python.
- **CO4:** To implement matrix multiplication using python.

3. Lab Rules, Safety Instructions & Evaluation Scheme

Lab Rules

1. Students must carry this manual and their observation notebook to every lab session.
2. Ensure all Python scripts are saved in the designated user workspace and backed up regularly.
3. Focus on writing core logic; avoid using advanced built-in functions or complex exception handling to ensure fundamental understanding.
4. Maintain silence and professional decorum.

Hardware & Software Requirements

- **Hardware:** PC with Intel i3/i5 or higher, 4GB RAM, 50GB HDD/SSD free space.
- **Software:**
 - **Operating System:** Windows 10/11, Ubuntu/Linux, or macOS.
 - **Compilers/Tools:** Python 3.x Interpreter.
 - **Editors/IDEs:** IDLE (Integrated Development and Learning Environment), Visual Studio Code (VS Code), PyCharm, or Jupyter Notebook.

INDEX

<u>S No.</u>	<u>TITLE</u>	<u>DATE</u>	<u>SIGN</u>
1.	Write a program to compute the GCD of two numbers.		
2.	Write a program to find the square root of a number.		
3.	Write a program to find the Exponentiation (power of a number).		
4.	Write a program to find the maximum of a list of numbers.		
5.	Write a program for Linear search and Binary search.		
6.	Write a program for Selection sort.		
7.	Write a program for Insertion sort.		
8.	Write a program to find first n prime numbers.		
9.	Write a program to multiply matrices.		
10.	Write a program that take command line arguments (word count).		
11.	Write a program to find the most frequent words in a text read from a file.		

PROGRAM-1

Aim: Write a program to compute the GCD of two numbers.

!--

{Program 1}

```
print("Himanshu,CSE-A,2024117536")
num1=int(input("Enter 1st no.:"))
num2=int(input("Enter 2nd no.:"))
if(num1>num2):
    small=num2
else:
    small=num1
for i in range(1,small+1):
    if(num1%i==0 and num2%i==0):
        gcd=i
    else:
        continue
print(f"GCD of {num1} & {num2}: {gcd}")
```

Output:

```
PS C:\Users\DELL> python -u "c:\college\IT File\IT file.py"
Himanshu,CSE-A,2024117536
Enter 1st no.:23
Enter 2nd no.:69
GCD of 23 & 69: 23
PS C:\Users\DELL> █
```

PROGRAM-2

Aim: Write a program to find the square root of a number.

:--

{Program:2}

```
print("Himanshu,CSE-A,2024117536")
num=int(input("Enter number:"))
if(num<0):
    print(f"Squareroot of {num} : Complex")
elif(num==0):
    print(f"Squareroot of {num} : {num}")
else:
    res=num**(1/2)
    if(res==int(res)):
        print(f"Perfect Squareroot of {num} : {res}")
    else:
        print(f"Squareroot of {num} : {res} \n{num} donot have perfect squareroot")
```

Output:

```
PS C:\Users\DELL> python -u "c:\college\IT File\ITfile.py"
Himanshu,CSE-A,2024117536
Enter number:36
Perfect Squareroot of 36 : 6.0
PS C:\Users\DELL> █
```

PROGRAM-3

Aim: Write a program to find the Exponentiation (power of a number).

!--

{Program-3}

```
print("Himanshu,CSE-A,2024117536")
num=int(input("Enter number:"))
pow=int(input("Enter power:"))
res=num**pow
print(f'{num} raised to {pow}: {res}')
```

Output:

```
PS C:\Users\DELL> python -u "c:\college\IT File\IT file.py"
Himanshu,CSE-A,2024117536
Enter number:24
Enter power:2
24 raised to 2: 576
PS C:\Users\DELL> █
```

PROGRAM-4

Aim: Write a program to find the maximum of a list of numbers.

{Program-4}

```
print("Himanshu,CSE-A,2024117536")
list=[]
n=int(input("Enter size of list:"))
for i in range(n):
    a=int(input("Enter value:"))
    list.append(a)

max=list[0]
for i in range(n):
    if(list[i]>max):
        max=list[i]
print(list)
print(f"Max in above list: {max}")
```

Output:

```
PS C:\Users\DELL> python -u "c:\college\IT File\IT file.py"
Himanshu,CSE-A,2024117536
Enter size of list:5
Enter value:24
Enter value:54
Enter value:22
Enter value:45
Enter value:67
[24, 54, 22, 45, 67]
Max in above list: 67
PS C:\Users\DELL> █
```

PROGRAM-5

Aim: Write a program for Linear search and Binary search.

{Program-5}

```
def linear(list,key):
    n=len(list)
    found=0
    for i in range(n):
        if(list[i]==key):
            print(f'{key} is found at index {i}')
            found=1
    if(found==0):
        print("Search is Unsuccesfull")
```

```
def binary(list,key):
    low=0
    high=len(list)-1
    mid=int((high+low)/2)
    found=0
    while(low<=high and list[mid]!=key):
        mid=int((high+low)/2)
        if(list[mid]>key):
            high=mid-1
        elif(list[mid]<key):
            low=mid+1
        else:
            print(f'{key} is found at index {i}')
            found=1
    if(found==0):
        print("Search is Unsuccesfull")
```

```
print("Himanshu,CSE-A,2024117536")
```

```
list=[]
```

```
n=int(input("Enter size of list:"))
```

```
for i in range(n):
```

```
    a=int(input("Enter value:"))
```

```
    list.append(a)
```

```
key=int(input("Enter value to be found:"))
```

```
print("1.Linear Search \t 2.Binary Search")
```

```
ch=int(input("Enter choice:"))
```

```
if(ch==1):
```

```
    linear(list,key)
```

```
elif(ch==2):
```

```
    binary(list,key)
```

```
else:
```

```
    print("Wrong Choice")
```


Output:

```
PS C:\Users\DELL> python -u "c:\college\IT File\IT file.py"
Himanshu,CSE-A,2024117536
Enter size of list:5
Enter value:24
Enter value:65
Enter value:76
Enter value:32
Enter value:11
Enter value to be found:32
1.Linear Search          2.Binary Search
Enter choice:1
32 is found at index 3
PS C:\Users\DELL> 
```

PROGRAM-6

Aim: Write a program for Selection sort.

!--

{Program-6}

```
print("Himanshu,CSE-A,2024117536")
```

```
arr=[]
```

```
n=int(input("Enter size of list:"))
```

```
for i in range(n):
```

```
    a=int(input("Enter value:"))
```

```
    arr.append(a)
```

```
print(f'Before sorting: {arr}')
```

```
for i in range(len(arr)):
```

```
    minpos = i
```

```
    for j in range(i + 1, len(arr)):
```

```
        if arr[j] < arr[minpos]:
```

```
            minpos = j
```

```
    arr[i], arr[minpos] = arr[minpos], arr[i]
```

```
print("Sorted array after selection sort=", arr)
```

Output:

```
PS C:\Users\DELL> python -u "c:\college\IT File\IT file.py"
Himanshu,CSE-A,2024117536
Enter size of list:5
Enter value:34
Enter value:21
Enter value:12
Enter value:4
Enter value:5
Before sorting:[34, 21, 12, 4, 5]
Sorted array after selection sort= [4, 5, 12, 21, 34]
PS C:\Users\DELL> █
```

PROGRAM-7

Aim: Write a program for Insertion sort.

:--

{Program-7}

```
print("Himanshu,CSE-A,2024117536")
arr=[]
n=int(input("Enter size of list:"))
for i in range(n):
    a=int(input("Enter value:"))
    arr.append(a)
print(f"Before sorting: {arr}")

for i in range(1, len(arr)):
    key = arr[i]
    j = i - 1
    while j >= 0 and arr[j] > key:
        arr[j + 1] = arr[j]
        j -= 1
    arr[j + 1] = key

print("Sorted array after insertion sort =", arr)
```

Output:

```
PS C:\Users\DELL> python -u "c:\college\IT File\IT file.py"
Himanshu,CSE-A,2024117536
Enter size of list:5
Enter value:23
Enter value:54
Enter value:65
Enter value:21
Enter value:12
Before sorting:[23, 54, 65, 21, 12]
Sorted array after insertion sort = [12, 21, 23, 54, 65]
PS C:\Users\DELL> █
```

PROGRAM-8

Aim: Write a program to find first n prime numbers.

:--

{Program-8}

```
print("Himanshu,CSE-A,2024117536")
n=int(input("Enter no. upto which u want prime no.'s:"))
found=0
for i in range(2,n):
    for j in range(2,i):
        if(i%j==0):
            found=1
            break
    else:
        found=0

    if(found!=1):
        print(i,end=" ")
```

Output

```
PS C:\Users\DELL> python -u "c:\college\IT File\IT file.py"
Himanshu,CSE-A,2024117536
Enter no. upto which u want prime no.'s:25
2 3 5 7 11 13 17 19 23
PS C:\Users\DELL> █
```

PROGRAM-9

Aim: Write a program to multiply matrices.

:--

{Program-9}

```
print("Himanshu,CSE-A,2024117536")
r1 = int(input("Enter rows of Matrix A: "))
c1 = int(input("Enter columns of Matrix A: "))
r2 = int(input("Enter rows of Matrix B: "))
c2 = int(input("Enter columns of Matrix B: "))

if c1 != r2:
    print("Matrix multiplication NOT possible")
else:
    A = []
    print("Enter Matrix A:")
    for i in range(r1):
        row = [int(x) for x in input().split()]
        A.append(row)

    B = []
    print("Enter Matrix B:")
    for i in range(r2):
        row = [int(x) for x in input().split()]
        B.append(row)

    C = []
    for i in range(r1):
        row = []
        for j in range(c2):
            s = 0
            for k in range(c1):
                s += A[i][k] * B[k][j]
            row.append(s)
        C.append(row)

    print("Result Matrix C:")
    for row in C:
        print(row)
```

Output:

```
PS C:\Users\DELL> python -u "c:\college\IT File\IT file.py"
Himanshu,CSE-A,2024117536
Enter rows of Matrix A: 2
Enter columns of Matrix A: 2
Enter columns of Matrix A: 2
Enter rows of Matrix B: 2
Enter columns of Matrix B: 2
Enter Matrix A:
1 2
3 4
Enter Matrix B:
4 5
1 2
Result Matrix C:
[6, 9]
[16, 23]
PS C:\Users\DELL> 
```

PROGRAM-10

Aim: Write a program that take command line arguments (word count).

--

{Program-10}

```
import sys
```

```
if len(sys.argv) < 2:
```

```
    print("Please give file name.")
```

```
else:
```

```
    f = open(sys.argv[1], "r")
```

```
    text = f.read()
```

```
    f.close()
```

```
    words = text.split()
```

```
    print("Word count =", len(words))
```

Output:

```
Microsoft Windows [Version 10.0.26100.6899]
(c) Microsoft Corporation. All rights reserved.

C:\college\IT File>python ITfile.py student.txt
Word count = 4

C:\college\IT File>
```

PROGRAM-11

Aim: Write a program to find the most frequent words in a text read from a file.

!--

{Program-11}

```
print("Himanshu,CSE-  
A,2024117536") filename  
= input("Enter file name: ")  
with open(filename, "w")  
as f:  
    a=input("Enter text in the  
    file:") f.write(a)
```

```
with  
    open("filename",  
    "r") as f: text =  
    f.read()  
    text=text.lower()  
    words =
```

```
text.split
```

```
() freq =
```

```
{}
```

```
for
```

```
    w
```

```
    in
```

```
    w
```

```
    o
```

```
    r
```

```
    d
```

```
    s:
```

```
    if
```

```
    w
```

```
    in
```

```
    fr
```

```
    e
```

```
    q:
```

```
        fr
```

```
        eq[
```

```
        w]
```

```
        +=
```

```
        1
```

```
    else
```

```
    :
```

```
        f
```

```
req[
```

```
w]=
```

```
1
```



```

print(
freq)
w
or
d
=
"""
co
un
t =
0

for w in freq:
    if
        freq
        [w]
        >
        cou
        nt:
        cou
        nt =
        freq
        [w]
        wor
        d =
        w

print("Most frequent word:",
word) print("Frequency:",
count

```

Output:

```

PS C:\Users\DELL> python -u "c:\college\IT File\IT file.py"
Himanshu,CSE-A,2024117536
Enter file name: stud.txt
Enter text in the file:What is the topic today? what does ur instinct say?
{'what': 2, 'the': 1, 'topic?': 1, 'is': 1, 'your': 1, 'fav': 1, 'color?': 1}
Most frequent word: what
Frequency: 2
PS C:\Users\DELL> █

```