REPORTFILE

[DOCUMENT SUBTITLE]

Mayank Anand

[COMPANY NAME] | [COMPANY ADDRESS]

INDEX

PARACTICAL FILE- COMPUTER SCIENCE (083)

LIST OF PARCTICALS (2023-24)

CLASS-XII

S.No	Name of Practical	Date	Teacher's Signature
1	Write a python program to input any number from user and calculate factorial of a number.		
2	Write a python program to read a line and calculate total number of uppercase letters, lowercase letters, spaces, and digits in the given string.		
3	Write a python program to search for a given word in a string.		
4	Write a python program to input integer values in a list and calculate sum of them.		
5	Write a python program to input list of elements and search the given element in the list.		
6	Write a python program to input integer values in a tuple and find largest and smallest number.		
7	Write a python program to create a dictionary with roll number, name and marks of n students in a class and display the name of the students who have scored marks above 85.		
8	Write a python program to perform the basic arithmetic operations in a menu driven program using functions.		
9	Write a python program to read a text file line by line and display each word separated by a **#.**		
10	Write a python program to read a text file and display number of vowels, consonants, uppercase, lowercase characters in a file.		
11	Write a python program to remove all the lines that contain the character 'a' in a file and write it to another file.		

```
input_file_name = input("Enter the input file name: ")
output_file_name = input("Enter the output file name: ")

try:
    with open(input_file_name, 'r') as input_file:
        with open(output_file_name, 'w') as output_file:
            for line in input_file:
                if 'a' not in line:
                    output_file.write(line)
    print("Lines containing 'a' removed and written to", output_file_name)
except FileNotFoundError:
    print("File not found.")
```

```
file_name = input("Enter the file name: ")

try:
    with open(file_name, 'r') as file:
        content = file.read()

        vowels = 0
        consonants = 0
        uppercase = 0
        lowercase = 0

        for char in content:
            if char.isalpha():
                if char.lower() in 'aeiou':
                    vowels += 1
                else:
                    consonants += 1
                if char.islower():
                    lowercase += 1
                else:
                    uppercase += 1

        print("Number of vowels:", vowels)
        print("Number of consonants:", consonants)
```

```
        print("Number of uppercase characters:", uppercase)
        print("Number of lowercase characters:", lowercase)

except FileNotFoundError:
    print("File not found.")
```

```python
import random

def roll_dice():
    return random.randint(1, 6)

num_rolls = int(input("Enter the number of times to roll the dice: "))

for i in range(num_rolls):
    result = roll_dice()
    print("Roll", i+1, ":", result)
```

```python
n = int(input("Enter the number of students: "))
student_data = {}

for i in range(n):
    roll_no = input("Enter Roll Number: ")
    name = input("Enter Name: ")
    marks = int(input("Enter Marks: "))
    student_data[roll_no] = {"Name": name, "Marks": marks}

print("Students who scored above 85:")
for roll_no, i in student_data.items():
    if i["Marks"] > 85:
        print(i["Name"])
```

```python
import math
n = int (input("Enter the number whose factorial you want to find: "))
print ("The factorial of the number is: ")
print (math.factorial(n))
```

```python
import csv

def write_to_csv(file_name, data):
    with open(file_name, 'a', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(data)

def read_csv(file_name):
    records = []
    with open(file_name, 'r') as file:
        reader = csv.reader(file)
        for row in reader:
            records.append(row)
    return records

def search_employee(records, employee_number):
    for record in records:
        if record[0] == employee_number:
            return record[1], record[2]
    return None, None

file_name = "employee_data.csv"

while True:
    print("Menu:")
    print("1. Add employee")
    print("2. Search employee")
    print("3. Exit")

    choice = input("Enter your choice (1-3): ")

    if choice == "1":
        employee_number = input("Enter employee number: ")
        name = input("Enter name: ")
        salary = input("Enter salary: ")
        write_to_csv(file_name, [employee_number, name, salary])
        print("Employee data saved.")
    elif choice == "2":
        employee_number = input("Enter employee number to search: ")
        records = read_csv(file_name)
```

```
        found_name, found_salary = search_employee(records, employee_number)
        if found_name and found_salary:
            print("Name:", found_name)
            print("Salary:", found_salary)
        else:
            print("Employee not found.")
    elif choice == "3":
        break
    else:
        print("Invalid choice.")

def add(x, y):
    return x + y

def subtract(x, y):
    return x - y

def multiply(x, y):
    return x * y

def divide(x, y):
    if y != 0:
        return x / y
    else:
        return "Cannot divide by zero"

while True:
    print("Menu:")
    print("1. Add")
    print("2. Subtract")
    print("3. Multiply")
    print("4. Divide")
    print("5. Exit")

    choice = input("Enter your choice (1-5): ")

    if choice == "5":
        break

    num1 = float(input("Enter first number: "))
```

```
num2 = float(input("Enter second number: "))

if choice == "1":
    print("Result:", add(num1, num2))
elif choice == "2":
    print("Result:", subtract(num1, num2))
elif choice == "3":
    print("Result:", multiply(num1, num2))
elif choice == "4":
    print("Result:", divide(num1, num2))
else:
    print("Invalid choice")
```

```python
l=eval(input("enter the elements in list form: "))
t=tuple(l)
a=max(t)
b=min(t)
print(a, "is the max", b, "is the min")
```

```python
file_name = input("Enter the file name: ")

try:
    with open(file_name, 'r') as file:
        for line in file:
            words = line.split()
            formatted_line = "#".join(words)
            print(formatted_line)
except FileNotFoundError:
    print("File not found.")
```

```python
line = input("Enter a line: ")

uppercase = 0
lowercase = 0
space = 0
digit = 0

for char in line:
    if char.isupper():
        uppercase += 1
    elif char.islower():
        lowercase += 1
    elif char.isspace():
        space += 1
    elif char.isdigit():
        digit += 1

print("Uppercase letters:", uppercase)
print("Lowercase letters:", lowercase)
print("Spaces:", space)
```

```
print("Digits:", digit)
```

```python
import pickle

def write_records(file_name, records):
    with open(file_name, 'wb') as file:
        pickle.dump(records, file)

def read_records(file_name):
    with open(file_name, 'rb') as file:
        records = pickle.load(file)
        return records

def search_roll_number(records, roll_number):
    for record in records:
        if record["roll_number"] == roll_number:
            return record["name"]
    return None

file_name = "student_records.dat"

records = [
    {"roll_number": 101, "name": "Alice"},
    {"roll_number": 102, "name": "Bob"},
    {"roll_number": 103, "name": "Charlie"}
]
write_records(file_name, records)

search_roll = int(input("Enter the roll number to search: "))
found_name = search_roll_number(read_records(file_name), search_roll)

if found_name:
    print("Name:", found_name)
else:
    print("Roll number not found.")
```

```python
l=eval(input("enter the list: "))
n=input("enter the search word: ")
if n in l:
    print("yes")
```

```
else:
    print("no")
```

```
string=input("enter the string: ")
n=input("enter the word you want to search for: ")
if n in string:
    print("yes the word is present")
else:
    print("no the word is not present")
```

```python
class Stack:
    def __init__(self):
        self.items = []

    def push(self, item):
        self.items.append(item)

    def pop(self):
        if not self.is_empty():
            return self.items.pop()
        else:
            return "Stack is empty"

    def peek(self):
        if not self.is_empty():
            return self.items[-1]
        else:
            return "Stack is empty"

    def is_empty(self):
        return len(self.items) == 0

    def size(self):
        return len(self.items)

stack = Stack()

stack.push(5)
stack.push(10)
stack.push(15)

print("Peek:", stack.peek())
print("Size:", stack.size())

print("Popped:", stack.pop())
print("Popped:", stack.pop())

print("Is empty:", stack.is_empty())
```

```
l=eval(input("enter the integers in the list"))
s=sum(l)
print(s)
```

```python
import csv

def write_to_csv(file_name, data):
    with open(file_name, 'a', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(data)

def read_csv(file_name):
    records = []
    with open(file_name, 'r') as file:
        reader = csv.reader(file)
        for row in reader:
            records.append(row)
    return records

def search_password(records, user_id):
    for record in records:
        if record == user_id:
            return record
    return None

file_name = "user_data.csv"

while True:
    print("Menu:")
    print("1. Create user")
    print("2. Search password")
    print("3. Exit")

    choice = input("Enter your choice (1-3): ")

    if choice == "1":
        user_id = input("Enter user ID: ")
        password = input("Enter password: ")
        write_to_csv(file_name, [user_id, password])
        print("User data saved.")
    elif choice == "2":
        user_id = input("Enter user ID to search password: ")
        records = read_csv(file_name)
        found_password = search_password(records, user_id)
```

```
        if found_password:
            print("Password:", found_password)
        else:
            print("User ID not found.")
    elif choice == "3":
        break
    else:
        print("Invalid choice.")
```

```python
import pickle

def write_records(file_name, records):
    with open(file_name, 'wb') as file:
        pickle.dump(records, file)

def read_records(file_name):
    with open(file_name, 'rb') as file:
        records = pickle.load(file)
        return records

def update_marks(records, roll_number, new_marks):
    for record in records:
        if record["roll_number"] == roll_number:
            record["marks"] = new_marks
            return True
    return False

file_name = "student_records.dat"

records = [
    {"roll_number": 101, "name": "Alice", "marks": 85},
    {"roll_number": 102, "name": "Bob", "marks": 92},
    {"roll_number": 103, "name": "Charlie", "marks": 78}
]
write_records(file_name, records)

search_roll = int(input("Enter the roll number to update marks: "))
new_marks = int(input("Enter the new marks: "))

updated = update_marks(read_records(file_name), search_roll, new_marks)

if updated:
    write_records(file_name, records)
    print("Marks updated successfully.")
else:
    print("Roll number not found.")
```