PRACTICAL-6 DATE:_____

AIM:.

i. Create a class named 'String' with one data member of type char *, which stores a string. Include default, parameterized and copy constructor to initialize the data member. Write a program to test this class.

ii. Write a base class named Employee and derive classes Male employee and Female Employee from it. Every employee has an id, name and a scale of salary. Make a function ComputePay (in hours) to compute the weekly payment of every employee. A male employee is paid on the number of days and hours he works. The female employee gets paid the wages for 40 hours a week, no matter what the actual hours are. Test this program to calculate the pay of employee.

iii. Create a class called scheme with scheme_id, scheme_name, outgoing_rate, and message charge. Derive customer class form scheme and include cust_id, name and mobile_no data.Define necessary functions to read and display data. Create a menu driven program to read call and message information for a customer and display the detail bill.

INPUT:-

i).

```
#include <iostream>
#include <cstring>
using namespace std;

class String {
  private:
    char * str;
  public:
    String() {
      str = new char[1];
      str[0] = '\0';
    }
    String (const char * inputstr) {
      str = new char [strlen (inputstr) + 1];
      strcpy (str, inputStr);
    }
    String (const String& other) {
      str = new char[strlen (other.str) + 1];
```

```
            strcpy(str, other.str);
        }

        ~String() {
            delete[] str;
        }
        void display() const {
            cout << str << endl;
        }
        int length() const {
            return strlen(str);
        }
};

int main() {
    String s1;
    cout << "String s1 (default constructor)";
    s1.display();
    String s2("Hello, World!");
    cout << "String s2 (parameterized constructor)";
    s2.display();
    String s3(s2);
    cout << "String s3 (copy constructor, copy of s2)";
    s3.display();
    cout << "Length of s2: " << s2.length() << endl;
    return 0;
}
```

Output:-

```
String s1 (default constructor)
String s2 (parameterized constructor): Hello, World!
String s3 (copy constructor, copy of s2): Hello, World!
Length of s2: 13
```

ii).

```
#include <iostream>
#include <string>
using namespace std;

class Employee {
    protected :
        int id ;
        string name ;
        double salaryScale;
    public :
        Employee (int empId , string empName, double empsalaryScale)
            : id(empId), name(empName), salaryScale (empsalary)
            {}
        virtual double computepay (int hours) = 0;
        void display () {
            cout << "ID :" << id << ", Name :" << name << ", Salary Scale :"
                                << salaryScale << endl;
    }
};

class MaleEmployee :
        public Employee {
            public :
                MaleEmployee (int empId, string empName, double
            empSalaryScale) : Employee (empId, empName, empSalaryScale)
                {}

                double computePay (int hours) override {
                    return hours * salaryScale * 10;
                }
    };

int main() {
        MaleEmployee maleEmp (101, "John", 15.5);
        FemaleEmployee femaleEmp (102, "Jane", 14.0); }
```

Output :-

```
ID : 101 , Name : John , Salary Scale : 15.5
Male Weekly Pay : $ 697.5
ID : 102 , Name : Jane , Salary Scale : 14
Female Weekly Pay : $560
```

iii).

```
# include <iostream>
# include <string>
using namespace std ;

class Scheme {
    protected :
        int scheme_id ;
        string scheme_name ;
        double outgoing_rate ;
        double message_charge ;
    public :
        void readscheme() {
            cout << " Enter Scheme ID : " ;
            cin >> scheme_id ;
            cout << " Enter Scheme Name : " ;
            cin. ignore() ;
            getline (cin, scheme_name) ;
            cout << " Enter Outgoing Rate per minute : " ;
            cin >> outgoing_rate ;
            cout << " Enter Message Charge per message : " ;
            cin >> message_charge ;
    }
    void displayScheme () {
        cout << " Scheme ID : " << scheme_id << endl ;
        cout << " Scheme Name : " << scheme_name << endl ;
        cout << " Outgoing rate : " << outgoing_rate << " per minute " << endl ;
    }
```

```
int main() {
    Customer cust;
    int choice;

    while (true) {
        cout << "\n Menu: \n";
        cout << "1. Enter Scheme Details \n";
        cout << "2. Enter Customer Details \n";
        cout << "3. Generate and Display Bill \n";
        cout << "4. Exit \n";
        cout << "Enter your choice: ";
        cin >> choice;

        switch (choice) {
            case 1:
                cust.readscheme();
                break;
            case 2:
                cust.readcustomer();
                break;
            case 3:
                cust.generateBill();
                break;
            case 4:
                cout << "Exiting the program...." << endl;
                return 0;
            default:
                cout << "Invalid choice! Please try again." << endl;
        }
    }
    return 0;
}
```

Output :-