SLOT- 2 ASSIGNMENT

```
// 1) Sum of Natural Numbers up to N
#include<iostream>
using namespace std;
int sumNaturalNumbers(int n) {
    return n * (n + 1) / 2;
}

// 2) Count Digits in a Number
int countDigits(int num) {
    int count = 0;
    while (num > 0) {
        num /= 10;
        ++count;
    }
    return count;
}

// 3) Function Overloading for Calculating Area
#include <cmath>
double area(double radius) {
    return M_PI * radius * radius;
}
double area(double length, double breadth) {
    return length * breadth;
}
double area(double base, double height, bool isTriangle) {
    return 0.5 * base * height;
}

// 4) Implement Polymorphism for Banking Transactions
class BankTransaction {
public:
    virtual void performTransaction() const = 0;
};
class Deposit : public BankTransaction {
public:
    void performTransaction() const override {
        cout << "Depositing money..." << endl;
    }
};
class Withdraw : public BankTransaction {
public:
    void performTransaction() const override {
        cout << "Withdrawing money..." << endl;
    }
};

// 5) Hierarchical Inheritance for Employee Management System
class Person {
protected:
    string name;
public:
    void setName(string n) { name = n; }
};
class Manager : public Person {
public:
    void display() { cout << name << " is a Manager." << endl; }
};
class Worker : public Person {
public:
    void display() { cout << name << " is a Worker." << endl; }
};

int main() {
    // Test all functions/classes here as needed

    // 1) Sum of Natural Numbers up to N
    int n = 10;
    cout << "Sum of natural numbers up to " << n << " is: " << sumNaturalNumbers(n) << endl;

    // 2) Count Digits in a Number
    int number = 12345;
    cout << "Number of digits in " << number << " is: " << countDigits(number) << endl;

    // 3) Function Overloading for Calculating Area
    cout << "Area of circle with radius 5: " << area(5.0) << endl;
    cout << "Area of rectangle with length 4 and breadth 6: " << area(4.0, 6.0) << endl;
    cout << "Area of triangle with base 5 and height 8: " << area(5.0, 8.0, true) << endl;

    // 4) Implement Polymorphism for Banking Transactions
    BankTransaction* transaction;
    Deposit deposit;
    Withdraw withdraw;

    transaction = &deposit;
    transaction->performTransaction();
    transaction = &withdraw;
    transaction->performTransaction();

    // 5) Hierarchical Inheritance for Employee Management System
    Manager manager;
    Worker worker;

    manager.setName("Alice");
    worker.setName("Bob");

    manager.display();
    worker.display();

    return 0;
}
```