

WWC

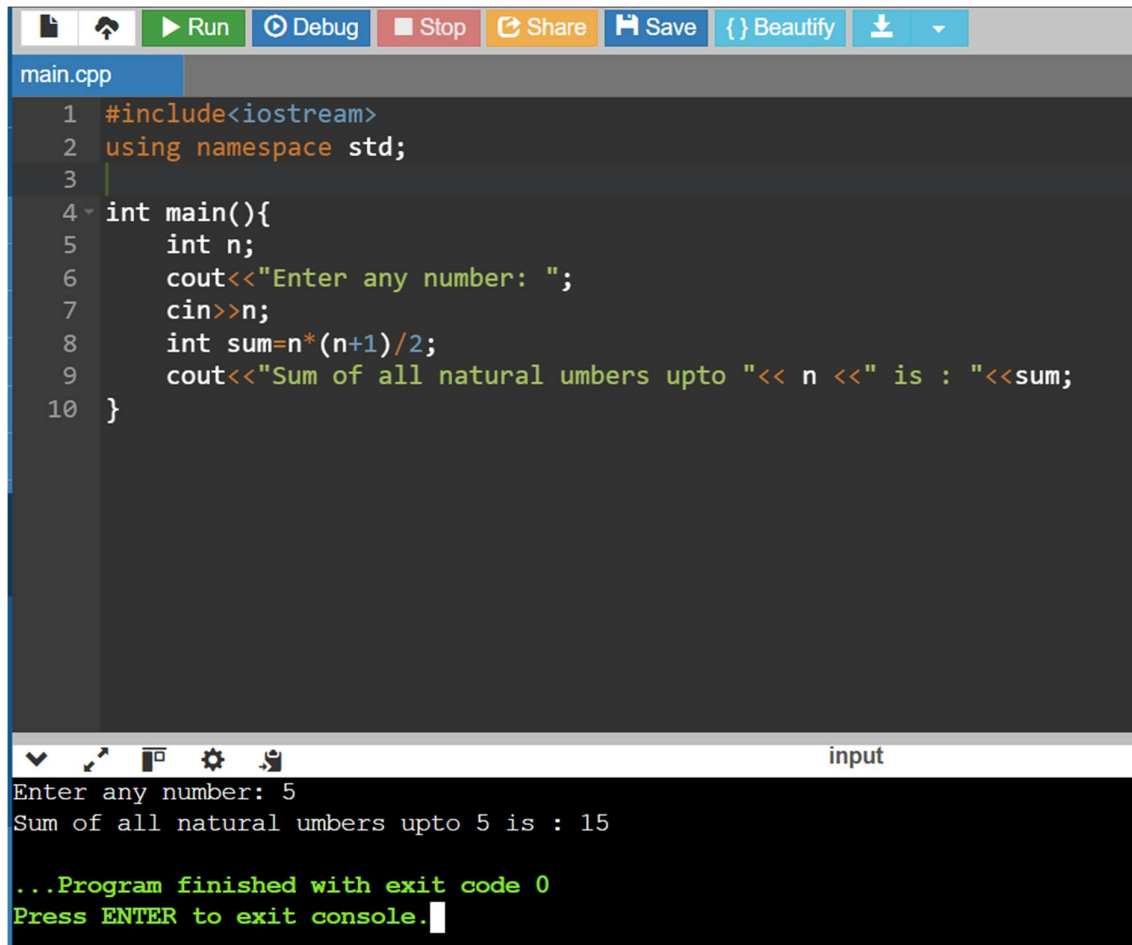
Name: Siddharth Lal

Sec: KPIT-901(B)

UID: 22bcs13410

DAY 1

Problem 1



The image shows a screenshot of a C++ IDE. The top toolbar includes buttons for Run, Debug, Stop, Share, Save, Beautify, and a dropdown menu. The main editor window displays a C++ program in a file named main.cpp. The code calculates the sum of natural numbers from 1 to n using the formula $\text{sum} = n \times (n + 1) / 2$. The program prompts the user to enter a number and then displays the sum. Below the editor, a console window shows the program's execution with input 5 and output 15. The console also displays the message "...Program finished with exit code 0" and "Press ENTER to exit console."

```
main.cpp
1  #include<iostream>
2  using namespace std;
3
4  int main(){
5      int n;
6      cout<<"Enter any number: ";
7      cin>>n;
8      int sum=n*(n+1)/2;
9      cout<<"Sum of all natural umbers upto "<< n <<" is : "<<sum;
10 }
```

input

Enter any number: 5
Sum of all natural umbers upto 5 is : 15
...Program finished with exit code 0
Press ENTER to exit console.

Problem 2

```
main.cpp
1  #include<iostream>
2  using namespace std;
3
4  int area(int a, int b){
5      return a*b;
6  }
7  int area(double base, double height){
8      return (0.5*base*height);
9  }
10 int area(int r){
11     return 3.14*r*r;
12 }
13 int main(){
14     cout<<"Area of a rectangle :"<<area(5,10)<<endl;
15     cout<<"Area of a triangle :"<<area(5.0,10.0)<<endl;
16     cout<<"Area of a circle :"<<area(5);
17 }
```

input

```
Area of a rectangle :50
Area of a triangle :25
Area of a circle :78

...Program finished with exit code 0
```

Problem 3

```
main.cpp
1  #include<iostream>
2  using namespace std;
3
4  int main(){
5      int n;
6      cout<<"Enter any number :";
7      cin>>n;
8      for(int i=1; i<=10;i++){
9          cout<<n<<" * "<<i<<" = "<<n*i<<endl;
10     }
11 }
```

inp

Enter any number :3

3 * 1 = 3

3 * 2 = 6

3 * 3 = 9

3 * 4 = 12

3 * 5 = 15

3 * 6 = 18

3 * 7 = 21

3 * 8 = 24

3 * 9 = 27

3 * 10 = 30

Problem 4

main.cpp

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  class Employee {
6      int employeeID;
7      string employeeName;
8      float employeeSalary;
9
10 public:
11     void setEmployeeID(int id) {
12         if (id >= 1 && id <= 1000000) {
13             employeeID = id;
14         } else {
15             cout << "Invalid Employee ID." << endl;
16         }
17     }
18
19     int getEmployeeID() const {
20         return employeeID;
21     }
22
23     void setEmployeeName(const string& name) {
24         if (name.length() <= 50) {
25             employeeName = name;
26         } else {
```

main.cpp

```
27         cout << "Invalid Employee Name." << endl;
28     }
29
30
31     string getEmployeeName() const {
32         return employeeName;
33     }
34
35     void setEmployeeSalary(float salary) {
36         if (salary >= 1.0 && salary <= 10000000.0) {
37             employeeSalary = salary;
38         } else {
39             cout << "Invalid Employee Salary." << endl;
40         }
41     }
42
43     float getEmployeeSalary() const {
44         return employeeSalary;
45     }
46
47     void displayDetails() const {
48         cout << "Employee ID: " << employeeID << endl;
```

main.cpp

```
46
47 void displayDetails() const {
48     cout << "Employee ID: " << employeeID << endl;
49     cout << "Employee Name: " << employeeName << endl;
50     cout << "Employee Salary: " << employeeSalary << endl;
51 }
52 };
53
54 int main() {
55     Employee emp;
56
57     int id;
58     string name;
59     float salary;
60
61     cout << "Enter Employee ID: ";
62     cin >> id;
63     cin.ignore();
64
65     cout << "Enter Employee Name: ";
66     getline(cin, name);
67
68     cout << "Enter Employee Salary: ";
69     cin >> salary;
70
71     emp.setEmployeeID(id);
72     emp.setEmployeeName(name);
```

main.cpp

```
73     cout << "Enter Employee Salary: ";
74     cin >> salary;
75
76     emp.setEmployeeID(id);
77     emp.setEmployeeName(name);
78     emp.setEmployeeSalary(salary);
79
80     cout << endl;
81     emp.displayDetails();
82
83     return 0;
84 }
```

input

```
Enter Employee ID: 01
Enter Employee Name: A
Enter Employee Salary: 99
```

```
Employee ID: 1
Employee Name: A
Employee Salary: 99
```

...Program finished with exit code 0

Problem 5

```
main.cpp
1  #include<iostream>
2  #include<iomanip>
3  using namespace std;
4
5  class Account{
6  protected:
7      double balance;
8  public:
9      Account(double bal) : balance(bal) {}
10     virtual void calculateInterest() = 0;
11     virtual ~Account(){}
12 };
13 class SavingsAccount : public Account {
14     double rate;
15     int time;
16 public:
17     SavingsAccount(double bal, double r, int t) : Account(bal), rate(r), time(t) {}
18
19     void calculateInterest() override {
20         double interest = (balance * rate * time) / 100.0;
21         cout << "Savings Account Interest: " << fixed << setprecision(2) << interest << endl;
22     }
23 };
24 class CurrentAccount : public Account {
25     double maintenanceFee;
26 public:
27     CurrentAccount(double bal, double fee) : Account(bal), maintenanceFee(fee) {}
28
29     void calculateInterest() override {
30         balance -= maintenanceFee;
31         cout << "Balance after fee deduction: " << fixed << setprecision(2) << balance << endl;
32     }
33 };
34
35 int main() {
36     int accountType;
37     cout << "Enter Account Type (1 for Savings, 2 for Current): ";
38     cin >> accountType;
39
40     if (accountType == 1) {
41         double balance, rate;
42         int time;
43         cout << "Enter Balance: ";
44         cin >> balance;
45         cout << "Enter Interest Rate (%): ";
46         cin >> rate;
47         cout << "Enter Time (years): ";
48         cin >> time;
```

```

main.cpp
39
40 - if (accountType == 1) {
41     double balance, rate;
42     int time;
43     cout << "Enter Balance: ";
44     cin >> balance;
45     cout << "Enter Interest Rate (%): ";
46     cin >> rate;
47     cout << "Enter Time (years): ";
48     cin >> time;
49
50 -     if (balance < 1000 || balance > 1000000 || rate < 1 || rate > 15 || time < 1 || time > 10) {
51         cout << "Invalid input values." << endl;
52         return 1;
53     }
54
55     SavingsAccount sa(balance, rate, time);
56     sa.calculateInterest();
57
58 - } else if (accountType == 2) {
59     double balance, fee;
60     cout << "Enter Balance: ";
61     cin >> balance;
62     cout << "Enter Monthly Maintenance Fee: ";
63     cin >> fee;
64
65 -     if (balance < 1000 || balance > 1000000 || fee < 50 || fee > 500) {
66         cout << "Invalid input values." << endl;
67         return 1;
68     }
69
70     CurrentAccount ca(balance, fee);
71     ca.calculateInterest();
72
73 - } else {
74     cout << "Invalid account type." << endl;
75 }
76
77 return 0;
78 }
79
80 double balance, fee;
60 cout << "Enter Balance: ";

```

input

```

Enter Account Type (1 for Savings, 2 for Current): 1
Enter Balance: 20000
Enter Interest Rate (%): 4
Enter Time (years): 3
Savings Account Interest: 2400.00

```

```

...Program finished with exit code 0
Press ENTER to exit console.

```