

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

class student
{
private:
    string name;
    int roll;

public:
    void getdata() {
        string ch;
        int n;
        cout << "Enter Name: ";
        cin >> ch;
        ch = name;
        cout << "Enter Roll number: ";
        cin >> n;
        n = roll;
    }
    void display() {
        cout << "Name: " << name;
        cout << "Roll Number: " << roll;
    }
};

class Parent
{
public:
    Parent()
    {
        cout << "Inside base class" << endl;
    }
    ~Parent()
    {
        cout << "Destructor evoked" << endl;
    }
};

class Child : public Parent
{
public:
    Child() {
        cout << "Inside sub class" << endl;
    }
};

float area(int r) {
    float a;
    float pi = 3.14;
    a = pi * r * r;
    return a;
}
```

```
}
int area(int l, int b) {
    float a1;
    a1 = l * b;
    return a1;
}
float area(int n, int bs, int h) {
    float a2;
    a2 = n * bs * h;
    return a2;
}
template <class t>
t small (t a[],int n) {
    t x=a[0];
    for (int i=0;i<n;i++) {
        if (x>a[i]) {
            x=a[i];
        }
    }
    return x;
}
class Bank {
    int c;

public:
    Bank() {
        c = 0;
    }
    void operator++(int)
    {
        c++;
    }
    int get_c()
    {
        return c;
    }
    void operator--(int) {
        c--;
    }
};
double division(int a, int b) {
    if (b == 0)
    {
        throw "Division by zero condition!";
    }
    return (a / b);
}
int main()
{
    int c, n;
    do
```

```

{
    cout << "1. Class & Object\n";
    cout << "2. Inheritance and Constructor Destructor\n";
    cout << "3. Polymorphism\n";
    cout << "4. Template\n";
    cout << "5. Opertor Overloading\n";
    cout << "6. File Handling\n";
    cout << "7. Exception Handling\n";
    cout << "8. Exit\n";
    cin >> n;
    switch
    {
    case 1: {
        student st;
        st.getdata();
        st.display();
        break;
    }
    case 2: {
        Parent P;
        Child Ch;
        break;
    }
    case 3: {
        int b, bs, h, r, l;
        float are;
        cout << "\nEnter the Radius of Circle: \n";
        cin >> r;
        are = area(r);
        cout << "\nArea of Circle: " << are << endl;
        cout << "Enter the Base & Hieght of Triangle:\n";
        cin >> bs;
        cin >> h;
        are = area(0.5, b, h);
        cout << "\nArea of Triangle: " << are << endl;
        cout << "\nEnter the Length & Bredth of Rectangle: \n";
        cin >> l >> b;
        are = area(l, b);
        cout << "\nArea of Rectangle: " << are << endl;
        break;
    }
    case 4: {
        int n,i,c;
        cout<<"Enter size: ";
        cin>>n;
        int a[n];
        cout<<"Enter Array : ";
        for (i=0;i<n;i++) {
            cin>>a[i];
        }
        cout<<"\nSmallest value in array is = "<<small(a,n);
    }
}

```

```
}
case 5: {
    Bank b;
    cout << "Initial No Of People " << b.get_c() << endl;
    b++;
    b++;
    b++;
    cout << "Present No Of People " << b.get_c() << endl;
    b--;
    b--;
    b--;
    cout << "Present No Of People " << b.get_c() << endl;
    break;
}
case 6: {
    ofstream MyFile("file.txt");
    MyFile << "Hello, My name is Adam.";
    string myText;
    ifstream MyFileRead("file.txt");
    while (getline(MyFileRead, myText))
    {
        cout << myText;
    }
    MyFile.close();
    MyFileRead.close();
    break;
}
case 7: {
    int x = 50;
    int y = 0;
    double z = 0;
    try
    {
        z = division(x, y);
        cout << z << endl;
    }
    catch (const char *msg)
    {
        cerr << msg << endl;
    }
    break;
}
case 8: {
    exit(0);
    break;
}
}
} while (c != 9);
return 0;
}
```