Object Oriented Programming CAT 2

Name: Brian karanja

REG NO: SCM212-1375/2023

Answers

Question 1

```cpp
#include <iostream>

using namespace std;

int sum(int num1, int num2)

{
    return num1 + num2;
}

int main()
{
    int x;
    int y;
    cout << "enter the value of x ";
    cin >> x;
    cout << "enter the value of y  ";
    cin >> y;
    int results = sum(x, y);

    cout << "The sum of " << x << " and " << y << " is " << results;

    return 0;
}
```

Question 2

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

// Function to calculate factorial
long fact(int n)
{
    if (n <=0)
    {
        cout << "Factorial is invalid";
        return 0;
    }
    else
    {
        long factorial = 1;
        for (int i = 2; i <= n; ++i)
        {
            factorial *= i;
        }
        return factorial;
    }
}

int main()
{
    int x;
    cout << "Enter an integer: ";
    cin >> x;

     long result = fact(x);
    cout << "Factorial of " << x<< " = " << result;

    return 0;
}
```

Question 3

```cpp
#include <iostream>

using namespace std;

// Function to calculate power of a number without using math library
double power(double n, int p = 2) {
    double result = 1.0;
    for (int i = 0; i < p; ++i) {
        result *= n;
    }
    return result;
}

int main() {
    double x;
    int y;

    cout << "Enter the value of x: ";
    cin >> x;
    cout << "Enter the exponent (default is 2 if omitted): ";
    cin >> y;

    double result = power(x, y);

    cout << "The result of " << x << " raised to the power of " << y << " is " <<
result << endl;

    return 0;
}
```

Question 4

```
18,4
```

Question 5

```
4,4
```

Question 6

```
190 280 100
280 10
570 10
```

Question 7

```
3, 30
7, 3
4, 40
4, 3
```

Question 8

```cpp
#include <iostream>
#include <string>

using namespace std;

class student {
private:
    int admno;
    string sname;
    float eng, math, science;
    float total;

    float ctotal() {
        return eng + math + science;
    }

public:
    void Takedata() {
        cout << "Enter admission number: ";
        cin >> admno;
        cout << "Enter student name: ";
        cin.ignore(); // to clear the input buffer
        getline(cin, sname);
        cout << "Enter marks for English: ";
        cin >> eng;
        cout << "Enter marks for Math: ";
        cin >> math;
        cout << "Enter marks for Science: ";
        cin >> science;

        // Calculate total
        total = ctotal();
    }

    void Showdata() {
        cout << "Admission Number: " << admno << endl;
        cout << "Student Name: " << sname << endl;
        cout << "Marks in English: " << eng << endl;
        cout << "Marks in Math: " << math << endl;
        cout << "Marks in Science: " << science << endl;
        cout << "Total Marks: " << total << endl;
    }
};
```

```cpp
int main() {
    student s;

    // Taking data from user
    s.Takedata();

    // Displaying data
    s.Showdata();

    return 0;
}
```

Question 9

```cpp
#include <iostream>

using namespace std;

class Rectangle {
private:
    float length;
    float width;

public:
    void setlength(float len) {
        length = len;
    }

    void setwidth(float wid) {
        width = wid;
    }

    float perimeter() {
        return 2 * (length + width);
    }

    float area() {
        return length * width;
    }

    void show() {
        cout << "Length: " << length << endl;
        cout << "Width: " << width << endl;
    }

    int sameArea(Rectangle other) {
        float this_area = area();
        float other_area = other.area();
        if (this_area == other_area)
            return 1;
        else
            return 0;
    }
};
```

```cpp
int main() {
    Rectangle r1, r2;

    // Setting dimensions for first rectangle
    r1.setlength(5.0);
    r1.setwidth(3.0);

    // Setting dimensions for second rectangle
    r2.setlength(4.0);
    r2.setwidth(6.0);

    // Displaying dimensions of rectangles
    cout << "Rectangle 1:" << endl;
    r1.show();
    cout << "Rectangle 2:" << endl;
    r2.show();

    // Calculating and displaying perimeter and area of rectangles
    cout << "Perimeter of Rectangle 1: " << r1.perimeter() << endl;
    cout << "Area of Rectangle 1: " << r1.area() << endl;

    cout << "Perimeter of Rectangle 2: " << r2.perimeter() << endl;
    cout << "Area of Rectangle 2: " << r2.area() << endl;

    // Checking if rectangles have the same area
    if (r1.sameArea(r2))
        cout << "Both rectangles have the same area." << endl;
    else
        cout << "Rectangles do not have the same area." << endl;

    return 0;
}
```

Question 10

```cpp
#include <iostream>

using namespace std;

class complex {
private:
    float real;
    float imaginary;

public:
    // Function to set the specified value in object
    void set(float r, float i) {
        real = r;
        imaginary = i;
    }

    // Function to display complex number object
    void disp() {
        cout << "Complex Number: " << real << " + " << imaginary << "i" << endl;
    }

    // Function to sum two complex numbers & return complex number
    complex sum(complex num2) {
        complex result;
        result.real = real + num2.real;
        result.imaginary = imaginary + num2.imaginary;
        return result;
    }
};

int main() {
    // Create three complex number objects
    complex c1, c2, c3;

    // Set the value in the first two objects
    c1.set(2.5, 3.7);
    c2.set(1.8, 4.2);
    // Calculate sum and assign it to the third object
    c3 = c1.sum(c2);
    // Display all complex numbers
    cout << "First complex number: ";
    c1.disp();
    cout << "Second complex number: ";
    c2.disp();
    cout << "Sum of the two complex numbers: ";
    c3.disp();
    return 0;
}
```

Question 11

```cpp
#include <iostream>

using namespace std;

class Distance {
private:
    int feet;
    float inches;

public:
    // Function to give value to object
    void set(int ft, float in) {
        feet = ft;
        inches = in;
    }
    // Function to display distance in feet and inches
    void disp() {
        cout << "Distance: " << feet << " feet " << inches << " inches" << endl;
    }
    // Function to sum two distances & return distance
    Distance add(Distance d2) {
        Distance result;
        result.feet = feet + d2.feet;
        result.inches = inches + d2.inches;
        if (result.inches >= 12) {
            result.feet += static_cast<int>(result.inches / 12);
            result.inches = fmod(result.inches, 12);
        }
        return result;
    }
};

int main() {
    // Create three Distance objects
    Distance d1, d2, d3;
    // Set the value in the first two objects
    d1.set(3, 6.5);
    d2.set(2, 10.3);
    // Calculate sum and assign it to the third object
    d3 = d1.add(d2);
    // Display all distances
    cout << "First distance: ";
    d1.disp();
    cout << "Second distance: ";
    d2.disp();
    cout << "Sum of the two distances: ";
    d3.disp();
    return 0;
}
```

Question 12

```cpp
#include <iostream>

using namespace std;

class time {
private:
    int hours;
    int minutes;
public:
    // Function to set the specified value in object
    void settime(int h, int m) {
        hours = h;
        minutes = m;
    }
    // Function to display time object
    void showtime() {
        cout << "Time: " << hours << " hours and " << minutes << " minutes" <<
endl;
    }
    // Function to sum two time objects & return time
    time sum(time t2) {
        time result;
        result.hours = hours + t2.hours;
        result.minutes = minutes + t2.minutes;
        if (result.minutes >= 60) {
            result.hours += result.minutes / 60;
            result.minutes %= 60;
        }
        return result;
    }
};
int main() {
    // Create three time objects
    time t1, t2, t3;
    // Set the value in the first two objects
    t1.settime(2, 30);
    t2.settime(1, 45);
    // Calculate sum and assign it to the third object
    t3 = t1.sum(t2);
    // Display all time objects
    cout << "First time object: ";
    t1.showtime();
    cout << "Second time object: ";
    t2.showtime();
    cout << "Sum of the two time objects: ";
    t3.showtime();
    return 0;
}
```

Question 13 i. Data members accessible from objects belonging to class Author:

- acode
- aname
- amount
- pub
- turnover

ii. Member functions accessible from objects belonging to class Branch:

- Branch()
- haveit()
- giveit()

iii. Members accessible from member functions of class Author:

- acode
- aname
- amount
- register()
- Publisher()
- enter()
- display()