

Lab Title: C++ Functions

Student Name Surname: Zeynep Yağmur Çalta

Student Number: 220201006

LAB Number: 3

1. Objective

Task 1: Write a program that defines an int function to calculate and return the sum of two numbers.

2. Source Code

Insert your program for each task with proper formatting with the outputs.

```
#include <iostream>
using namespace std;
int sum(int a, int b) {
    int sum = a + b;
    cout << "sum of the numbers are" << sum << endl;
    return sum;
}

int main()
{
    int a, b;
    cout << "please enter a value for sum";
    cin >> a;
    cout << "please enter b value for sum";
    cin >> b;
    sum(a, b);
}
```

Task 2: Write a program that uses a function prototype and defines the function after the main() function.

2. Source Code

Insert your program for each task with proper formatting with the outputs.

```
int mult(int a, int b);

int main() {
    int a, b;

    cout << "please enter the two number to divide" << endl;

    cout << "first number:" << endl;
    cin >> a;

    cout << "second number:" << endl;
    cin >> b;

    mult(a, b);
}

int mult(int a, int b) {
    double mult = a * b;

    cout << "Answer is :" << mult;

    return mult;
}
```

Task 3: Write a program that defines a function to calculate the area of a circle using a user-entered

radius.

2. Source Code

Insert your program for each task with proper formatting with the outputs.

```
double areaOfCircle(double r) {
    const double pi = 3.14;

    double area = pi * r * r;

    cout << "area of the circle is" << area << endl;

    return area;
```

```

}

int main() {
    double r;
    cout << "please enter the r value for calculate :" << endl;
    cin >> r;

    areaOfCircle(r);
}

```

Task 4: Write a void function that takes one integer parameter and prints its square.

2. Source Code

Insert your program for each task with proper formatting with the outputs.

```

void square(int a) {
    int square = a * a;

    cout << "square of the number is:" << square << endl;
}

int main() {
    int a;
    cout << "please enter the number for calculate square:";
    cin >> a;

    square(a);

}

```

Task 5: Write a program that demonstrates call by value, showing that the original variable does not

change. 2. Source Code

Insert your program for each task with proper formatting with the outputs.

QUESDTION-5
<pre> int main() { int a = 10; int *p = &a; p = &a; cout << "value of x:" << a << endl; cout << "address of x :" << p << endl; } </pre>

```

cout << "value trought pointer: "<void change(int a) {
    a = 20;
    cout << "the value inside of the function is:" << a << endl;
}

int main() {
    int number = 10;

    cout << "inside of the functuon value is:" << number << endl;
    change(number);

    cout << "after call the functuon value is:" << number << endl;

}

```

Task 6: Write a program that demonstrates call by reference, showing that the original variable changes.

2. Source Code

Insert your program for each task with proper formatting with the outputs.

```

void changeValue(int& x) {
    x = x + 10;
}

int main() {
    int num = 5;
    cout << "Before function call: " << num << endl;

    changeValue(num);

    cout << "After function call: " << num << endl;

    return 0;
}

cout << "please enter the double number for call int function" << endl;
cin >> x;
cin >> y;

add(a, b);
add(x, y);
}

```

Task 7 : Write two overloaded functions named add() — one that adds two integers and another that adds two doubles.

2. Source Code

Insert your program for each task with proper formatting with the outputs.

```
int display(int a, int b) {
    int sum = a + b;
    cout << "displays for with parameters function" << sum << endl;

    return 1;
}

void display() {

    cout << "displays for no parameters function" << endl;
}

int main() {

    int a, b;
    cout << "please enter a and b value:" << endl;
    cin >> a;
    cin >> b;

    display(a, b);

    display();

}
```

Task 8 :

Write two overloaded functions named display() — one with no parameters and one with a string parameter.

2. Source Code

Insert your program for each task with proper formatting with the outputs.

```
int display(int a, int b) {
    int sum = a + b;
```

```

        cout << "displays for with parameters function" << sum << endl;
    }
}

void display() {
    cout << "displays for no parameters function" << endl;
}

int main() {
    int a, b;
    cout << "please enter a and b value:" << endl;
    cin >> a;
    cin >> b;

    display(a, b);

    display();
}

```

Task 9: Write a simple program that overrides a function by redefining it with the same name and

parameters in a new scope.

2. Source Code

```

class Animal {
public:
    virtual void speak() {
        cout << "The animal makes a sound." << endl;
    }
};

class Dog : public Animal {
public:
    void speak() override {
        cout << "The dog barks." << endl;
    }
};

int main() {
    Animal a;
    Dog d;

    a.speak();
    d.speak();
    return 0;
}

```

Task 10: Write a program that defines multiple overloaded max() functions to find the larger value among integers and floats.

2. Source Code

```
int max(int a, int b) {  
    int sum = a + b;  
    cout << "this is integer part and sum is" << endl;  
    return sum;  
}  
  
float max(float a, float b) {  
    float sum = a + b;  
    cout << "this is float part and sum is" << endl;  
  
    return sum;  
}  
  
int main() {  
    cout << max(3, 7) << endl;  
    cout << max(2.5f, 4.8f) << endl;  
}
```

Task 11: Write a program that uses the same function name for different parameter types to demonstrate function overloading.

2. Source Code

```
void display(int num) {  
    cout << "Integer: " << num << endl;  
}  
  
void display(double num) {  
    cout << "Double: " << num << endl;  
}  
  
void display(string text) {  
    cout << "String: " << text << endl;  
}  
  
int main() {  
  
    display(10);  
    display(3.14);  
    display("Hello");  
  
    return 0;  
}
```