**Lab 16**

# Task 1

#include<iostream>

void swap(int& x, int& y);//Declare function

int main()

{

int num1, num2; //taking two Numbers

std::cout << "Enter First Number: ";

std::cin >> num1;

std::cout << "Enter Second Number: ";

std::cin >> num2;

swap(num1, num2);//calling Function

std::cout << "Values After Swaping \n\n";//showing after swap

std::cout << "Num1 = " << num1 << std::endl;

std::cout << "Num2 = " << num2 << std::endl;

system("pause");

}

void swap(int& x, int& y)//defining swap function

{

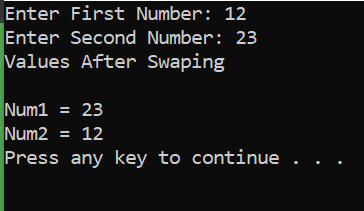
int z = x;

x = y;

y = z;

}

# Output



# Task 2

#include<iostream>

void line(double& x0, double& y0);//declaring Function

int main()

{

double x0, y0;//Taking two Numbers

std::cout << "Enter X0 : ";

std::cin >> x0;

std::cout << "Enter Y0 : ";

std::cin >> y0;

line(x0, y0);

system("pause");

}

void line(double& x0, double& y0)//defining function

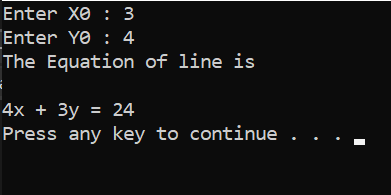
{

std::cout << "The Equation of line is \n\n";

std::cout << y0 << "x" << " + " << x0 << "y" << " = " << 2 \* x0 \* y0 << "\n";//evaluating the expression

}

# Output



# Task 3

#include<iostream>

void input(); //declaring input function

double MaxDistance(int& n, int& fuel); //declare fuction

int main()

{

input();//callling function

system("pause");

}

void input()

{

int bikes, fuel\_amount;//taking bikes and fuel

double max;

std::cout << "Enter Number of bikes : ";

std::cin >> bikes;

std::cout << "Enter Amount Of Fuel : ";

std::cin >> fuel\_amount;

max = MaxDistance(bikes, fuel\_amount);//calling function for calculating maximum distance

std::cout << "The Maximum Distance is : " << max << std::endl;

}

double MaxDistance(int& n, int& fuel)//defining function

{

double distance = 0;

for (int i = 1; i <= n; i++)

{

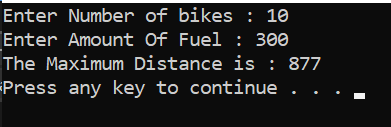
distance += fuel / i;

}

return distance;

}

# Output



# Task 4

#include<iostream>

int largestx(int& n, int& p);//declare function

int main()

{

int n, p, x, orignal\_num;

std::cout << "Enter n : ";//taking inputs

std::cin >> n;

std::cout << "Enter p : ";

std::cin >> p;

orignal\_num = n;//saving orignal number

x = largestx(n, p);//calling function

std::cout << "x = " << x << "\n";

std::cout << p << "^" << x << " divides " << orignal\_num << "! and " << x << " is largest such power of " << p << "\n";//giving output

system("pause");

}

int largestx(int& n, int& p)//defining

{

int largest\_pow = 0;

while (n > 0)//loop till n becomes zero

{

n = n / p;

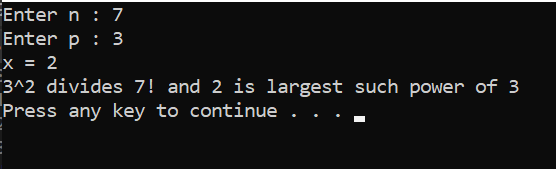
largest\_pow += n;

}

return largest\_pow;

}

# Output



# Task 5

#include <iostream>

#include <cstdlib>

#include <ctime>

int genratingQuiz(int difficultyLevel);//declaring functions

void showQuestion(int difficultyLevel, int& correctAnswers, int& incorrectAnswers);

void CAI();

int main()

{

srand(time(0));

int choice;

do { //taking choice from user

std::cout << "Choose an option:\n";

std::cout << "1. Computer-assisted instruction\n";

std::cout << "2. Exit\n";

std::cin >> choice;

switch (choice) {

case 1:

CAI();

break;

case 2:

std::cout << "Exiting program.\n";

break;

default:

std::cout << "Invalid choice. Please try again.\n";

}

} while (choice != 2);

return 0;

}

int genratingQuiz(int difficultyLevel)

{

int max = 9;

int min = 1;

for (int i = 1; i < difficultyLevel; ++i) {

max \*= 10;

min \*= 10;

}

return rand() % (max - min + 1) + min;

}

void showQuestion(int difficult, int& correct, int& wrong) //defining function

{

int num1 = genratingQuiz(difficult);

int num2 = genratingQuiz(difficult);

int answer, response;

do

{

std::cout << "How much is " << num1 << " times " << num2 << "? ";

std::cin >> answer;

if (answer == num1 \* num2) //checking answer

{

correct++;

response = rand() % 4 + 1; //genrating random response

switch (response)

{

case 1:

std::cout << "Very good!\n";

break;

case 2:

std::cout << "Excellent!\n";

break;

case 3:

std::cout << "Nice work!\n";

break;

case 4:

std::cout << "Keep up the good work!\n";

break;

}

}

else {

wrong++;

response = rand() % 4 + 1; //genrating random response for wrong answer

switch (response) {

case 1:

std::cout << "No. Please try again.\n";

break;

case 2:

std::cout << "Wrong. Try once more.\n";

break;

case 3:

std::cout << "Don't give up!\n";

break;

case 4:

std::cout << "No. Keep trying.\n";

break;

}

}

} while((answer != num1 \* num2));

}

void CAI() //checkingg difficulty level

{

int difficult, correct = 0, wrong = 0;

std::cout << "Enter difficulty level : ";

std::cin >> difficult;

for (int i = 0; i < 10; ++i)

{

showQuestion(difficult, correct, wrong);

}

double percent = static\_cast<double>(correct) / 10 \* 100;

if (percent < 75)

{

std::cout << "Please ask your teacher for extra help.\n";

}

else

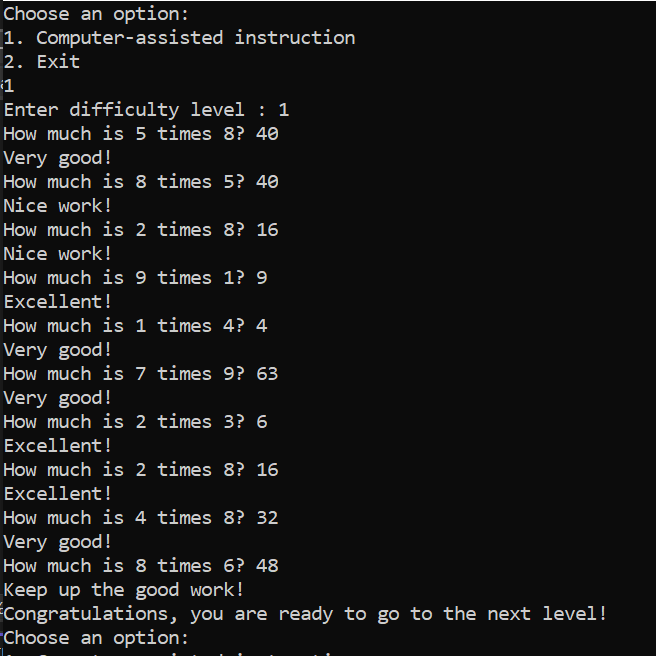
{

std::cout << "Congratulations, you are ready to go to the next level!\n";

}

}

# Output



# Task 6

#include <iostream>

void sum(int num1, int num2, int& sum);//declare

int main()

{

int num1, num2, solution;

std::cout << "Enter the first number: "; //taking inputs

std::cin >> num1;

std::cout << "Enter the second number: ";

std::cin >> num2;

sum(num1, num2, solution); //function call

std::cout << "The sum is: " << solution << std::endl; //output

return 0;

}

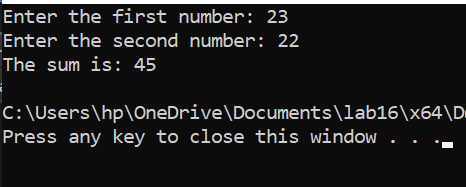
void sum(int num1, int num2, int& sum)//defining

{

sum = num1 + num2;

}

# Output



# Task 7

#include <iostream>

#include <cmath>

int main()

{

std::cout << "1. abs(-4): " << abs(-4) << std::endl; //writing all functions and giving their answers

std::cout << "2. fabs(10.8): " << fabs(10.8) << std::endl;

std::cout << "3. fabs(-2.5): " << fabs(-2.5) << std::endl;

std::cout << "4. pow(3.2, 2): " << pow(3.2, 2) << std::endl;

std::cout << "5. pow(2.5, 3): " << pow(2.5, 3) << std::endl;

std::cout << "6. sqrt(25.0): " << sqrt(25.0) << std::endl;

std::cout << "7. sqrt(6.25): " << sqrt(6.25) << std::endl;

std::cout << "8. pow(3.0, 4.0) / abs(-9): " << pow(3.0, 4.0) / abs(-9) << std::endl;

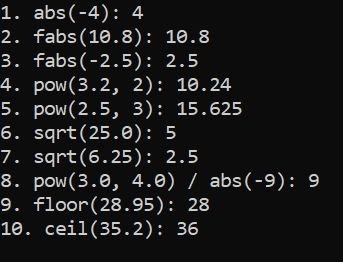
std::cout << "9. floor(28.95): " << floor(28.95) << std::endl;

std::cout << "10. ceil(35.2): " << ceil(35.2) << std::endl;

return 0;

}

# Output



# Task 8

#include <iostream>

int qualityPoints(int average);//declare function

int main()

{

int average;

std::cout << "Enter Average: "; //taking average

std::cin >> average;

int points = qualityPoints(average);

std::cout << "Quality Points: " << points << std::endl; //output

return 0;

}

int qualityPoints(int average) //defining

{

if (average >= 90 && average <= 100)

{

return 4;

}

else if (average >= 80 && average < 90)

{

return 3;

}

else if (average >= 70 && average < 80)

{

return 2;

}

else if (average >= 60 && average < 70)

{

return 1;

}

else

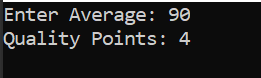
{

return 0;

}

}

# Output



Task 9

#include <iostream>

using namespace std;

void addFirst(int& first, int& second);

void doubleFirst(int one, int two);

void squareFirst(int& ref, int val);

int main()

{

int num = 5;

cout << "Line 1: Inside main: num = " << num

<< endl; //Line 1

addFirst(num, num); //Line 2

cout << "Line 3: Inside main after addFirst:"

<< " num = " << num << endl; //Line 3

doubleFirst(num, num); //Line 4

cout << "Line 5: Inside main after "

<< "doubleFirst: num = " << num << endl; //Line 5

squareFirst(num, num); //Line 6

cout << "Line 7: Inside main after "

<< "squareFirst: num = " << num << endl; //Line 7

system("pause");

return 0;

}

void addFirst(int& first, int& second)

{

cout << "Line 8: Inside addFirst: first = "

<< first << ", second = " << second << endl; //Line 8

first = first + 2; //Line 9

cout << "Line 10: Inside addFirst: first = "

<< first << ", second = " << second << endl; //Line 10

second = second \* 2; //Line 11

cout << "Line 12: Inside addFirst: first = "

<< first << ", second = " << second << endl; //Line 12

}

void doubleFirst(int one, int two)

{

cout << "Line 13: Inside doubleFirst: one = "

<< one << ", two = " << two << endl; //Line 13

one = one \* 2; //Line 14

cout << "Line 15: Inside doubleFirst: one = "

<< one << ", two = " << two << endl; //Line 15

two = two + 2; //Line 16

line cout << "Line 17: Inside doubleFirst: one = "

<< one << ", two = " << two << endl; //Line 17

}

void squareFirst(int& ref, int val)

{

cout << "Line 18: Inside squareFirst: ref = "

<< ref << ", val = " << val << endl; //Line 18

ref = ref \* ref; //Line 19

cout << "Line 20: Inside squareFirst: ref = "

<< ref << ", val = " << val << endl; //Line 20

val = val + 2; //Line 21

cout << "Line 22: Inside squareFirst: ref = "

<< ref << ", val = " << val << endl; //Line 22

}

# Explain lines :

**Line 1:** cout the num 5 .

**Line 2:** call a function add

**Line 3:** cout num on screen which is 14 .

**Line 4:** call a function doubleFirst

**Line 5:** cout the num on screen . which is 14.

**Line 6:** call a function squareFirst

**Line 7:** cout the num on screen which is 196.

**Line 8:** cout the value of first and second variables on screen which are 5 5

**Line 9:** add 2

**Line 10:** cout the value of first and second variables/parameters on screen which are (7,7)

**Line 11:** second multiply 2

**Line 12:** cout the value of first and second variables/parameters on screen which are (14,14)

**Line 13:** cout the values of one and two variables/parameters on screen which are(14,14)

**Line 14:** multiply one by 2

**Line 15 :** cout the values of one and two variables/parameters on screen which are 28 14

**Line 16:** two = two + 2

**Line 17:** cout the values of one and two variables on screen which are (28,16)

**Line 18:** cout the values of ref and val variables on screen which are (14,14)

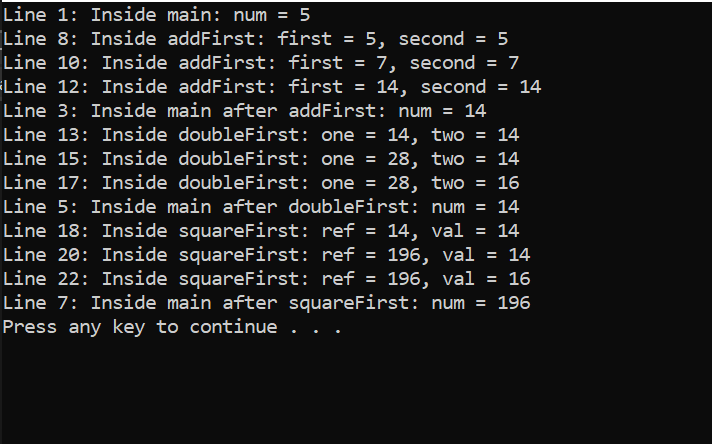
**Line 19:** taking square

**Line 20:** cout the values of ref and val variables on screen which are 196 14

**Line 21:** val = val + 2

**Line 22:** cout the values of ref and val variables on screen which are 196 16

# Output



# Task 10

#include<iostream>

int sum(int, int);

float sum(float, float);//declare

int sum(int, int, int);

int main()

{

int choice;

std::cout << "Enter 1 for int\n";

std::cout << "Enter 2 for float\n";

std::cout << "Enter 3 int \n";

std::cin >> choice;

switch (choice) {

case 1: {

int num1, num2;

std::cout << "Enter two integers\n";

std::cin >> num1 >> num2;

std::cout << "Sum is : " << sum(num1, num2) << "\n";

break;

}

case 2: {

float num1, num2;

std::cout << "Enter two floats" << "\n";

std::cin >> num1 >> num2;

std::cout << "Sum is : " << sum(num1, num2) << std::endl;

break;

}

case 3: {

int num1, num2, num3;

std::cout << "Enter three integers " << "\n";

std::cin >> num1 >> num2 >> num3;

std::cout << "Sum is : " << sum(num1, num2, num3) << "\n";

break;

}

default:

std::cout << "Invalid choice" << "\n";

}

system("pause");

}

int sum(int num1, int num2) ///defining functions

{

return (num1 + num2);

}

int sum(int num1, int num2, int num3)

{

return (num1 + num2 + num3);

}

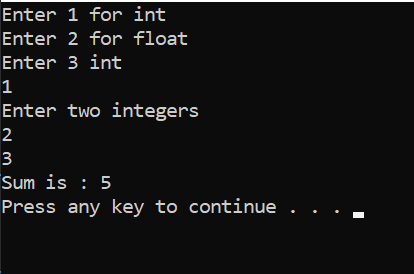
float sum(float num1, float num2)

{

return (num1 + num2);

}

# Output



# Task 11

#include<iostream>

int sum(int, int, int = 0, int = 0);//declare

int main()

{

int choice; //enter choice

std::cout << "Enter 1 for 2\n";

std::cout << "Enter 2 for 3\n";

std::cout << "Enter 3 for 4\n";

std::cin >> choice;

switch (choice) {

case 1: {

int a, b;

std::cout << "Enter two integer numbers \n";

std::cin >> a >> b;

std::cout << "Sum is : " << sum(a, b) << "\n";

break;

}

case 2: {

int num1, num2, num3;

std::cout << "Enter three integer numbers" << "\n";

std::cin >> num1 >> num2 >> num3;

std::cout << "Sum is : " << sum(num1, num2, num3) << std::endl;

break;

}

case 3: {

int num1, num2, num3, num4;

std::cout << "Enter four integer numbers" << "\n";

std::cin >> num1 >> num2 >> num3 >> num4;

std::cout << "Sum is : " << sum(num1, num2, num3, num4) << "\n";

break;

}

default:

std::cout << "Invalid choice" << "\n";

}

system("pause");

}

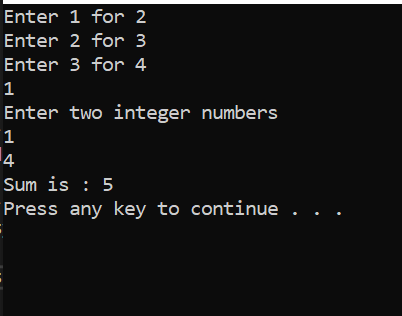
int sum(int num1, int num2, int num3, int num4)//definig

{

return (num1 + num2 + num3 + num4);

}

# Output



# Task 12

#include<iostream>

void uppercase(char a[], int);

int main()

{

int size = 0;

char arr[100];

std::cout << "Enter String \n";

gets\_s(arr);

while (arr[size] != '\0')

{

size++;

}

uppercase(arr, size); //calling

std::cout << "The string in uppercase : " << arr << "\n";//output

system("pause");

}

void uppercase(char a[], int size)//defining

{

for (int i = 0; i <= size; i++)

{

if (a[i] >= 'a' && a[i] <= 'z')

{

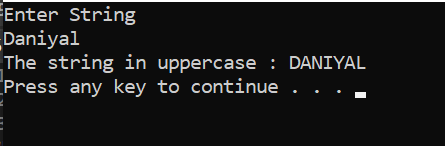
a[i] = a[i] - 32;

}

}

}

# Output



# Task 13

#include<iostream>

int countDigit(char a[], int); //declare

int main()

{

char arr[100];

std::cout << "Input the string ";

gets\_s(arr);

std::cout << "The number of digits in the array are : " << countDigit(arr, 100) << "\n"; //calling function in cout

system("pause");

}

int countDigit(char a[], int size)//defining function

{

int count = 0;

while (a[count] != '\0')//checking

{

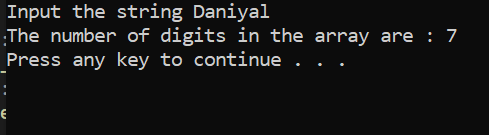
count++;

}

return(count);

}

# Output



# Task 14

#include <iostream>

#include <iomanip>

using namespace std;

int volume(int l = 1, int w = 1, int h = 1);

void funcOne(int& x, double y = 12.34, char z = 'B');

int main()

{

int a = 23;

double b = 48.78;

char ch = 'M';

cout << fixed << showpoint;

cout << setprecision(2);

cout << "Line 1: a = " << a << ", b = "

<< b << ", ch = " << ch << endl;

cout << "Line 2: Volume = " << volume()// the value return because int function is used so 1 is return from function body. so now the function return the volume.

// inverted commas were missing

<< endl; //Line 2

cout << "Line 3: Volume = " << volume(5.4, 4)// type cast....the value return because int function is used and 5.4 convert into 5 and there is a chances of loss of data because conversion of double to int so 20 value is return.

<< endl; //Line 3

cout << "Line 4: Volume = " << volume('A')//ascii will be returned.... here the ASCII of 'A' is 65 so it return 65 value from function

<< endl; //Line 4

cout << "Line 5: Volume = "<< volume(6, 4, 5) << endl; //multiply all....In function after multiplication 120 value is return in call function.

funcOne(a); //correct... call funcOnE(a) and a=23 and its address go to formal parameter so after multiplication its value is 46 y = 12.34 and z = B (y,z values come by default)

funcOne(a, 42.68, 1); //correct formal parameters ... In this void function value of a = 46 ,b=42.68,ch=1 go to the formal parameters and then x=92,y=42.68 because data type function is used and z= null because here data type is char.

funcOne(a, 34.65, 'Q'); //by refreence .... In this function value of a = 92 because we sent the address in this function and recent 2 funcOne functions so x=184,y=34.65,z=Q.

cout << "Line 9: a = " << a << ", b = "

<< b << ", ch = " << ch << endl;

system("pause");

return 0;

}

int volume(int l, int w, int h) //defining

{

return l \* w \* h;

}

void funcOne(int& x, double y, char z)

{

x = 2 \* x;

cout << "Line 12: x = " << x << ", y = "

<< y << ", z = " << z << endl;

}

# Output

