OOPS Assignment 3

Made by : Arnav Bansal

Group : CSE-G1

SID : 23103046

1-C++ Program To enter required number of book records and display them using array

declaration of class’s objects .

#include <iostream>

#include <vector>

#include <string>

using namespace std;

class Book {

public:

string title;

string author;

int year;

void getData() {

cout << "Enter title: ";

cin >> ws;

getline(cin, title);

cout << "Enter author: ";

getline(cin, author);

cout << "Enter year: ";

cin >> year;

}

void displayData() const {

cout << "Title: " << title << ", Author: " << author << ", Year: " << year << endl;

}

};

int main(){

int n;

cout << "Enter number of books: ";

cin >> n;

vector<Book> books(n);

for (int i = 0; i < n; ++i) {

cout << "Enter details for book " << i + 1 << endl;

books[i].getData();

}

cout << "\nDisplaying book details:\n";

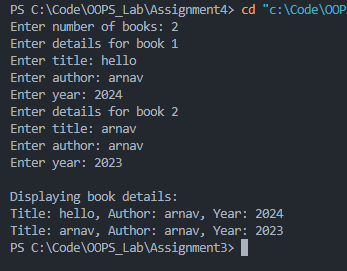
for (const auto &book : books) {

book.displayData();

}

return 0;

}

  
  
2-C++ Program To enter the data of a students and display it using indirection

operator(using pointer to class objects) and structure operator.

#include <iostream>

#include <vector>

#include <string>

using namespace std;

class Student {

public:

string name;

int age;

void getData() {

cout << "Enter name: ";

cin >> name;

cout << "Enter age: ";

cin >> age;

}

void displayData() const {

cout << "Name: " << name << ", Age: " << age << endl;

}

};

int main() {

Student \*student = new Student;

cout << "Enter student details:\n";

student->getData();

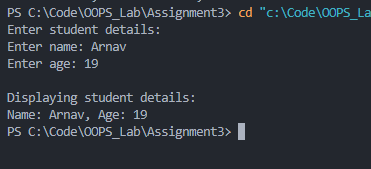
cout << "\nDisplaying student details:\n";

student->displayData();

delete student;

return 0;

}



3-C++ Program To find the highest of two numbers using the nesting of member functions.

#include <iostream>

#include <vector>

#include <string>

using namespace std;

class Number {

public:

int a, b;

void getNumbers() {

cout << "Enter two numbers: ";

cin >> a >> b;

}

int highest() {

return max(a, b);

}

void displayHighest() {

cout << "The highest number is: " << highest() << endl;

}

};

int main() {

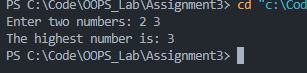
Number num;

num.getNumbers();

num.displayHighest();

return 0;

}



4-C++ Program using of inline function sum of three numbers.

#include <iostream>

#include <vector>

#include <string>

using namespace std;

int sum(int a, int b, int c) {

return a + b + c;

}

int main() {

int x, y, z;

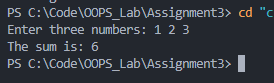
cout << "Enter three numbers: ";

cin >> x >> y >> z;

cout << "The sum is: " << sum(x, y, z) << endl;

return 0;

}



5- C++ Program to simulate an arithmetic calculator for integer. The program should be able

to produce the result calculated and the number of arithmetic operators performed so far .Any

wrong operation is to be reported.

#include <iostream>

#include <vector>

#include <string>

using namespace std;

int main() {

int a, b, result, count = 0;

char op;

bool run = true;

while (run) {

cout << "Enter operation (a + b, a - b, a \* b, a / b) or 'q' to quit: ";

cin >> a >> op >> b;

switch (op) {

case '+':

result = a + b;

break;

case '-':

result = a - b;

break;

case '\*':

result = a \* b;

break;

case '/':

if (b != 0) {

result = a / b;

} else {

cout << "Division by zero error!" << endl;

continue;

}

break;

case 'q':

run = false;

continue;

default:

cout << "Invalid operation!" << endl;

continue;

}

count++;

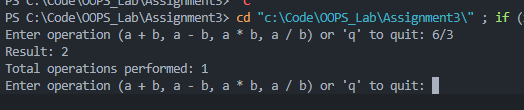
cout << "Result: " << result << endl;

cout << "Total operations performed: " << count << endl;

}

return 0;

}



6- C++ Program Printing simple and Inverted Pyramid pattern using for loop.

#include <iostream>

using namespace std;

int main() {

int n;

cout << "Enter number of rows: ";

cin >> n;

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

for (int j = i; j < n; j++) {

cout << " ";

}

for (int k = 1; k <= (2 \* i - 1); k++) {

cout << "\*";

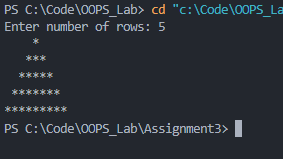
}

cout << endl;

}

return 0;

}



7- C++ Program Operator Precedence and Associativity in C++.

#include <iostream>

int main() {

int a = 5, b = 10, c = 20;

int result = a + b \* c / 10 - 2;

std::cout << "Result: " << result << std::endl;

return 0;

}

