Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 РЕФЕРАТ

По дисциплине «Основы алгоритмизации и программирования»

На тему «Тестирование кода»

Выполнила:

Студентка1 курса 7 группы

Гриценко Анна Александровна

2023, Минск

Тестирование кода из лекции

#include <iostream> // Подключение библиотеки для использования cout

using namespace std; // Использование стандартного пространства имен

// Определение структуры Funds для хранения информации о финансах

struct Funds {

char\* bank; // Название банка

float bankfund; // Финансовые средства на счете банка

float savefund; // Финансовые средства на сберегательном счете

}\*pFunds, stan = {

"Supper\_bank", // Имя банка

2000.02, // Финансовые средства на счете банка (invoice #1)

3000.03 // Финансовые средства на сберегательном счете (invoice #2)

};

// Прототипы функций

float sumByValue(struct Funds z); // Передача структуры по значению

float sumByPoint(struct Funds\* x); // Передача структуры по указателю

float sumByReference(struct Funds& w); // Передача структуры по ссылке

float sumByMemoryClass(); // Передача структуры с использованием класса памяти

Funds& InitFunds(float value); // Функция, которая возвращает ссылку на структуру

void main() //основная функция

{

float total;//объявление переменной вещественного типа

setlocale(LC\_CTYPE, "rus");//подключение кириллицы

/\* Передача структуры по значению \*/

total = sumByValue(stan);//вызов функции sumByValue с аргументом stan и присвоение резельтата ее работы переменной total

printf("sumByValue. Всего на счетах в банке: %.2f\n\n", total);//вывод текста в консоль

/\* Передача структуры по указателю \*/

total = sumByPoint(&stan);//В данной строке кода вызывается функция sumByPoint и результат выполнения функции sumByPoint будет присвоен переменной total

printf("sumByPoint. Всего на счетах в банке: %.2f\n", total);//вывод текста в консоль

pFunds = &stan;//присваиваение указателю pFunds адрес переменной stan

total = sumByPoint(pFunds);//вызов функция sumByPoint с аргументом pFunds и присвоение резельтата ее работы переменной total

printf("sumByPoint. Всего на счетах в банке: %.2f\n\n", total);//вывод текста в консоль

/\* Передача структуры по ссылке \*/

Funds& MyRefStr = stan;//создается ссылка MyRefStr типа Funds&, которая ссылается на объект stan типа Funds

total = sumByReference(MyRefStr);//В данной строке кода вызывается функция sumByReference и результат выполнения функции sumByReference будет присвоен переменной total

printf("sumByReference. Всего на счетах в банке: %2.f\n\n", total);//вывод текста в консоль

/\* Передача структуры в функцию с использованием класса памяти \*/

total = sumByMemoryClass();

printf("sumByMemoryClass. Всего на счетах в банке: %.2f\n\n", total);//вывод текста в консоль

// Функция возвращает ссылку на структуру

Funds fx = InitFunds(1000.13);//создается объект типа Funds с именем fx, который инициализируется с помощью функции InitFunds, которая принимает значение 1000.13 в качестве аргумента

cout << "bankfund=" << fx.bankfund << endl << "savefund=" << fx.savefund << endl << endl;//вывод текста в консоль

system("PAUSE");//команда,которая используется для приостановки выполнения программы до тех пор, пока пользователь не нажмет клавишу

}

// Передача самой структуры в функцию

float sumByValue(struct Funds z) {

return (z.bankfund + z.savefund);

/\* Этот вариант наименее предпочтителен, т.к. функция получает копию структуры и, соответственно, всех её полей.

А копирование требует дополнительных затрат, которые в данном случае не нужны \*/

}

// Передача структуры по указателю

float sumByPoint(struct Funds\* money) {

return (money->bankfund + money->savefund);//возвращает сумму элементов структуры bankfund и savefund

}

// Передача в функцию ссылки на структуру (работает в С++, в "сыром" С ссылок нет)

float sumByReference(struct Funds& w) {

/\* В этом случае w - это ссылка на оригинальную структуру, а не её копия, как в случае значений \*/

return (w.bankfund + w.savefund);//возвращает сумму элементов структуры bankfund и savefund

}

// Передача данных структуры в функцию с использованием класса памяти

float sumByMemoryClass() {

return (stan.bankfund + stan.savefund);//возвращает сумму элементов структуры bankfund и savefund

}

// Функция, которая возвращает ссылку на структуру

Funds& InitFunds(float value) {

Funds fx;//объект fx типа Funds

fx.bankfund = fx.savefund = value;//устанавливаются значения членов bankfund и savefund равными переданному значению value

return fx;//возвращается объект fx

}

