|  |  |
| --- | --- |
| http://www.pl130.ru/doc/index/%D0%9B%D0%BE%D0%B3%D0%BE%D1%82%D0%B8%D0%BF22.jpg | Санкт-Петербургское государственное бюджетное  профессиональное образовательное учреждение  "Колледж электроники и приборостроения" |

отчет

**по практической работе № 16**

**по дисциплине «Основы алгоритмизации и программирования»**

Тема: Создание библиотеки подпрограмм

|  |  |  |
| --- | --- | --- |
| Студент гр. 13 ИТ |  | Липинский К.С. |
| Преподаватель |  | Счастливцев А.К. |

Санкт-Петербург

2024

**Цель работы.**

Научиться создавать и подключать библиотеки подпрограмм.

**Выполнение работы.**

**Задание 1.**

Program.cs

using static StackLib.General;

namespace Prac;

public static class Program

{

private static Stack<object> \_stack = new();

private static void Main()

{

var status = true;

string menu = "1: Добавление элемента в вершину стека\n"

+ "2: Удаление элемента из вершины стека\n"

+ "3: Реверсирование стека\n"

+ "4: Печать стека\n"

+ "5: Очистка стека\n"

+ "6: Выход\n\n"

+ "Выбор: ";

while (status)

{

Console.Write(menu);

int choice = Convert.ToInt32(Console.ReadLine());

switch (choice)

{

case 1:

StackAdd(\_stack);

break;

case 2:

StackDelete(\_stack);

break;

case 3:

StackReverse(ref \_stack);

break;

case 4:

StackPrint(\_stack);

break;

case 5:

StackClear(\_stack);

break;

case 6:

status = false;

break;

default:

Console.WriteLine("Неверный выбор!");

break;

}

Console.WriteLine("Нажмите любую клавишу");

Console.ReadKey();

Console.Clear();

}

}

}

General.cs

namespace StackLib;

public class General

{

public static void StackClear(Stack<object> stack)

{

switch (stack.Count)

{

case 0:

Console.WriteLine("\nСтек уже пустой");

break;

default:

stack.Clear();

Console.WriteLine("\nИз стека были успешно удалены все элементы");

break;

}

}

public static void StackAdd(Stack<object> stack)

{

Console.WriteLine(stack.Count == 0 ? "Текущий стек пустой" : "Текущий стек содержит элементы.");

Console.WriteLine("Введите значение, которое необходимо добавить. Для выхода нажмите 'Q'.");

while (true)

{

var input = Console.ReadLine();

if (string.IsNullOrEmpty(input) || input == "Q")

break;

stack.Push(input);

Console.WriteLine($"Добавлено значение: {input}");

}

Console.WriteLine("Текущий стек: ");

foreach (var item in stack) Console.Write($"{item} ");

Console.WriteLine();

}

public static void StackDelete(Stack<object> stack)

{

Console.WriteLine("Текущий стек: ");

foreach (var item in stack) Console.Write($"{item} ");

Console.WriteLine();

stack.Pop();

Console.WriteLine("Стек после удаления: ");

foreach (var item in stack) Console.Write($"{item} ");

Console.WriteLine();

}

public static void StackPrint(Stack<object> stack)

{

if (stack.Count == 0)

{

Console.WriteLine("Текущий стек пустой");

}

else

{

Console.WriteLine("Текущий стек: ");

foreach (var item in stack) Console.Write($"{item} ");

Console.WriteLine();

}

}

public static void StackReverse(ref Stack<object> stack)

{

if (stack.Count == 0)

{

Console.WriteLine("Текущий стек пустой");

}

else

{

Console.WriteLine("Текущий стек: ");

foreach (var item in stack) Console.Write($"{item} ");

Console.WriteLine();

Stack<object> reversed = new(stack);

stack = reversed;

Console.WriteLine("Реверсированный стек: ");

foreach (var item in stack) Console.Write($"{item} ");

Console.WriteLine("\n");

}

}

}

**Вопросы.**

1. [**Библиотека подпрограмм** — это сборник подпрограмм или объектов, используемых для разработки программного обеспечения](https://bing.com/search?q=). [Это своего рода набор «строительных блоков» для кода, который помогает программистам ускорять и упрощать свою работу](https://bing.com/search?q=).
2. [**Статическая библиотека** — это собрание \*.obj-файлов библиотеки в \*.lib, который мы можем прицепить к нашей программе в момент компиляции](https://bing.com/search?q=). [Она крепко зашита в среду и является непосредственной частью среды](https://bing.com/search?q=).

[**Динамическая библиотека** — это файл, содержащий машинный код, который загружается в память процесса загрузчиком программ операционной системы либо при создании процесса, либо по запросу уже работающего процесса](https://bing.com/search?q=). [Она является полуопциональной, может быть связанной со средой и используется несколькими программами одновременно4](https://yandex.ru/q/question/v_chiom_raznitsa_mezhdu_staticheskoi_i_72f6d1e0/).