**МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КПІ»**

**Зображення, що містить будівля, ескіз, панорама, чорно-білий

Автоматично згенерований опис**

**Кафедра інформаційних систем та технологій**

Лабораторна робота №2

з дисципліни «Розробка програмного забезпечення на платформі .Net»

на тему:

«Модульне тестування. Ознайомлення з засобами та практиками

модульного тестування»

|  |  |  |
| --- | --- | --- |
| Викладач:  Бардін В. |  | Виконала:                                               Студентка групи ІС-12  Гусєва Тетяна |

Київ – 2023

**Мета лабораторної роботи** – навчитися створювати модульні тести для

вихідного коду розроблювального програмного забезпечення.

**Завдання:**

1. Додати до проекту власної узагальненої колекції (застосувати

виконану лабораторну роботу No1) проект модульних тестів,

використовуючи певний фреймворк (Nunit, Xunit, тощо).

2. Розробити модульні тести для функціоналу колекції.

3. Дослідити ступінь покриття модульними тестами вихідного коду

колекції, використовуючи, наприклад, засіб AxoCover.

Варіант:

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

**Реалізація:**

public class UnitTest1

{

[Fact]

public void AddLast\_IncreasesCount()

{

var deque = new DequeClass<int>();

deque.AddLast(1);

Assert.Equal(1, deque.Count);

}

[Fact]

public void RemoveFirst\_DecreaseCount()

{

DequeClass<int> deque = new DequeClass<int>();

deque.AddLast(1);

deque.RemoveFirst();

Assert.Equal(0, deque.Count);

}

[Fact]

public void First\_ThrowsExceptionWhenEmpty()

{

var deque = new DequeClass<int>();

Assert.Throws<InvalidOperationException>(() => deque.First);

}

[Fact]

public void Last\_ThrowsExceptionWhenEmpty()

{

var deque = new DequeClass<int>();

Assert.Throws<InvalidOperationException>(() => deque.Last);

}

[Fact]

public void Clear\_EmptiesTheDeque()

{

var deque = new DequeClass<char>();

deque.AddLast('A');

deque.AddLast('B');

deque.Clear();

Assert.True(deque.IsEmpty);

Assert.Equal(0, deque.Count);

}

[Fact]

public void Contains\_ReturnsTrueForExistingElement()

{

var deque = new DequeClass<int>();

deque.AddLast(10);

deque.AddLast(20);

deque.AddLast(30);

Assert.True(deque.Contains(20));

}

[Fact]

public void Contains\_ReturnsFalseForNonExistingElement()

{

var deque = new DequeClass<string>();

deque.AddFirst("Apple");

deque.AddFirst("Banana");

deque.AddFirst("Cherry");

Assert.False(deque.Contains("Grapes"));

}

[Theory]

[InlineData(1, 2, 3)]

[InlineData(10, 20, 30, 40, 50)]

public void AddLastAndRemoveFirst\_ReturnsCorrectElements(params int[] elements)

{

var deque = new DequeClass<int>();

foreach (var element in elements)

{

deque.AddLast(element);

}

foreach (var element in elements)

{

Assert.Equal(element, deque.RemoveFirst());

}

}

[Theory]

[InlineData("a", "b", "c")]

[InlineData("w", "q", "y", "x", "z")]

public void AddFirstAndRemoveLast\_ReturnsCorrectElements(params string[] elements)

{

var deque = new DequeClass<string>();

foreach (var element in elements)

{

deque.AddFirst(element);

}

foreach (var element in elements)

{

Assert.Equal(element, deque.RemoveLast());

}

}

[Fact]

public void GetEnumerator\_ReturnsAllElements()

{

var deque = new DequeClass<int>();

deque.AddLast(1);

deque.AddLast(2);

deque.AddLast(3);

var elements = deque.ToList();

Assert.Equal(new List<int> { 1, 2, 3 }, elements);

}

[Fact]

public void GetEnumerator\_EmptyDeque\_ReturnsEmptyEnumeration()

{

var deque = new DequeClass<string>();

var elements = deque.ToList();

Assert.Empty(elements);

}

}

**Результат:**

Зображення, що містить текст, знімок екрана, програмне забезпечення, монітор

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, програмне забезпечення, Мультимедійне програмне забезпечення

Автоматично згенерований опис