**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

Державний університет інфраструктури та технологій

Коледж морського і річкового флоту

**ЗВІТ**

**з лабораторної роботи № 3** **«Створення конструкторів ініціалізації у класах»**

з дисципліни **«Об’єктно-орієнтованого програмування»**

**Варіант № 11**

студент групи **ПР-318**

спеціальності **«ІПЗ»**

Пронь Дмитро Вікторович

**Викладач:** **Горбаченко В.А**.

**Київ – 2020**

**Мета роботи**: Навчитися створювати конструктори та деструктори класу.

**Порядок виконання роботи**

1. На основі програмного коду ЛР№2 та індивідуального варіанту завдання об’єкту реального світу (Велосипед) доповнено реалізовані класи конструкторами та деструкторами, які виконуються при створенні та знищенні класу.
2. Проведено тестування всіх частин програмного коду на коректність роботи.

3) Прокоментовано кожен рядок програмного коду.

4) Висновки: В даній лабораторній роботі було створено конструктори та деструктори класу об’єкту реального світу, а саме «Велосипед».

**Контрольні запитання**

1. **Дати визначення конструктору класу.**

**Конструктор** — спеціальний метод класу, який автоматично викликається при створенні об'єкта.

1. **Дати визначення деструктору класу**.

**Деструктор** — спеціальний метод [класу](https://uk.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B0%D1%81_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)" \o "Клас (програмування)), який автоматично викликається при знищенні [об'єкта](https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%27%D1%94%D0%BA%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)" \o "Об'єкт (програмування)) і призначений для його [деініціалізації](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%96%D1%86%D1%96%D0%B0%D0%BB%D1%96%D0%B7%D0%B0%D1%86%D1%96%D1%8F" \o "Ініціалізація) (наприклад: звільнення ресурсів — [пам'яті](https://uk.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0_%D0%BF%D0%B0%D0%BC%27%D1%8F%D1%82%D1%8C" \o "Оперативна пам'ять), закриття файлів тощо).

1. **Наведіть приклад створення конструктора класу.**

public MyBook()

{

Title = "";

Author = "";

year = 1900;

price = 0.00f;

}

1. **Наведіть приклад створення деструктора класу.**

~Person()

    {

        Console.Beep();

        Console.WriteLine("Disposed");

    }

1. **Чим відрізняється деструктор від конструктора?**

Конструктор выкликається при створенні об’єкта, а деструктор при його знищенні.

1. **Наскільки реалізований клас в C# був змінений в порівнянні з попереднім кодом? Що саме було реалізовано?**

Було додано конструктори і деструктори.

1. **Чи відбулися в класі(-ах) кардинальні зміни в структурі? Якщо так, то чому саме?**

Так, ми можемо створювати і знищувати об’єкти.

// конструктор класса \_2Data с параметрами

\_2Data::\_2Data(double initData1, double initData2)

{

data1 = initData1;

data2 = initData2;

cout << "\nСработал конструктор с параметрами.\n";

}

// конструктор класса \_2Data без параметров

\_2Data::\_2Data()

{

data1 = 0;

data2 = 0;

cout << "\nСработал конструктор без параметров.\n";

}

**Реалізація коду програми C#**

using k2020\_lr3;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace k2020\_lr3

{

class TWheel

{

// Клас «Колесо»

public int spokeCout; //Кількість спиць

public int diametr; //Діаметр колес

public void ScanParams()

{

Console.BackgroundColor = ConsoleColor.Red;

Console.ForegroundColor =

ConsoleColor.Yellow;

Console.WriteLine("\n\n[TWheel] :: SCANPARAMS");

Console.ResetColor();

Console.Write("Кiлькiсть спиць: ");

spokeCout = Convert.ToInt32(Console.ReadLine()); // Число

Console.Write("Дiаметр колес : ");

diametr = Convert.ToInt32(Console.ReadLine()); // Число

}

public void PrintParams()

{

Console.BackgroundColor = ConsoleColor.DarkGreen;

Console.ForegroundColor =

ConsoleColor.White;

Console.WriteLine("\n\n[TWheel] :: PRINTPARAMS");

Console.ResetColor();

Console.WriteLine("Кiлькiсть спиць : " + spokeCout);

Console.WriteLine("Дiаметр колес : " + diametr);

Console.WriteLine();

}

}

class Tframe

{

// Класс «Рама»

public string colour; //колір рами

public void ScanParams()

{

Console.BackgroundColor = ConsoleColor.Red;

Console.ForegroundColor =

ConsoleColor.Yellow;

Console.WriteLine("\n\n[Tframe] :: SCANPARAMS");

Console.ResetColor();

Console.Write("Колiр рами: ");

colour = Console.ReadLine(); // Текст

}

public void PrintParams()

{

Console.BackgroundColor = ConsoleColor.DarkGreen;

Console.ForegroundColor =

ConsoleColor.White;

Console.WriteLine("\n\n[Tframe] :: PRINTPARAMS");

Console.ResetColor();

Console.WriteLine("Колiр рами : " + colour);

Console.WriteLine();

}

}

class Trudder

{

// Клас «Руль»

public string colour; //колір руля

public int height; //Висота руля

public void ScanParams()

{

Console.BackgroundColor = ConsoleColor.Red;

Console.ForegroundColor =

ConsoleColor.Yellow;

Console.WriteLine("\n\n[Trudder] :: SCANPARAMS");

Console.ResetColor();

Console.Write("Колiр руля: ");

colour = Console.ReadLine(); // Текст

Console.Write("Висота руля : ");

height = Convert.ToInt32(Console.ReadLine()); // Число

}

public void PrintParams()

{

Console.BackgroundColor = ConsoleColor.DarkGreen;

Console.ForegroundColor =

ConsoleColor.White;

Console.WriteLine("\n\n[Trudder] :: PRINTPARAMS");

Console.ResetColor();

Console.WriteLine("Колiр руля : " + colour);

Console.WriteLine("Висота руля : " + height);

Console.WriteLine();

}

}

class TBike

{

// Класс «Велосипед»

public string model; //Модель

public string firm; //Назва фірми яка його виготовила

public string dob; //дата виробництва

public bool isworked; //Працюючому стані

public TWheel wheel; // Клас «Колесо»

public Tframe frame; // Класс «Рама»

public Trudder rudder; // Клас «Руль»

public TBike()

{

// Constructor

wheel = new TWheel();

frame = new Tframe();

rudder = new Trudder();

}

~TBike()

{

// Destructor

}

public void ScanParams()

{

Console.BackgroundColor = ConsoleColor.Red;

Console.ForegroundColor =

ConsoleColor.Yellow;

Console.WriteLine("[TBike] :: SCANPARAMS");

Console.ResetColor();

Console.Write("Модель : ");

model = Console.ReadLine(); // Текст

Console.Write("Фiрма : ");

firm = Console.ReadLine(); // Текст

Console.Write("Дата : ");

dob = Console.ReadLine(); // Текст

Console.Write("Працюючому станi : ");

string tmp\_work = Console.ReadLine().ToUpper();

isworked = (tmp\_work == "YES" || tmp\_work == "Y" ? true : false); // bool

}

public void PrintParams()

{

Console.BackgroundColor = ConsoleColor.DarkGreen;

Console.ForegroundColor = ConsoleColor.White;

Console.WriteLine("[TBike] :: PRINTPARAMS");

Console.ResetColor();

Console.WriteLine("Модель : " + model);

Console.WriteLine("Фiрма : " + firm);

Console.WriteLine("Дата виготовлення : " + dob);

Console.WriteLine("В працюючому станi : " + isworked);

}

public void ScanAllParams()

{

ScanParams();

Console.WriteLine("========================================");

wheel.ScanParams();

Console.WriteLine("========================================");

frame.ScanParams();

Console.WriteLine("========================================");

rudder.ScanParams();

Console.WriteLine("========================================");

}

public void PrintAllParams()

{

PrintParams();

Console.WriteLine("========================================");

wheel.PrintParams();

Console.WriteLine("========================================");

frame.PrintParams();

Console.WriteLine("========================================");

rudder.PrintParams();

Console.WriteLine("========================================");

}

}

}

class Program

{

static void Main(string[] args)

{

TBike bike = new TBike();

bike.ScanAllParams(); // зчитуэмо всі необхідні дані для заповнення полів класів

Console.WriteLine("========================================");

bike.PrintAllParams(); // виводимо всі дані полів класів

Console.WriteLine("========================================");

Console.WriteLine("PROGRAM STOP!");

Console.Beep();

Console.ReadLine();

}

}