

Практична робота №1
Знайомство з платформою .NET
Система оцінювання

№	Тема	К-ть балів
1.	<i>Захист принаймні одного завдання з роботи</i>	1
2.	Завдання 1	0,2*
3.	Завдання 2	0,3*
4.	Завдання 3	0,4*
5.	Завдання 4	0,3*
6.	Завдання 5	0,3*
7.	Завдання 6	0,4*
8.	Завдання 7	0,4*
9.	Завдання 8	0,6*
10.	Завдання 9	0,7*
11.	<i>Здача звіту</i>	0,4
	Всього	5

* – діє бонусна система

Завдання на практичне заняття

1. **(Обмін валют)** Напишіть програму, яка буде приймати на вхід суму грошей у гривнях та обчислюватиме відповідну суму купівлі та продажу в інших валютах на момент запуску програми. Курси валют візьміть за [посиланням](#).

2. Закон ідеального газу – математичне наближення поведінки газів при зміні тиску, об'єму та температури. Зазвичай він описується рівнянням

$$PV = nRT,$$

де P – тиск у паскалях (Па), V – об'єм, n – кількість молів речовини,

$R = 8.314 \frac{\text{Дж}}{\text{моль} \cdot \text{К}}$ – універсальна газова стала, T – температура в градусах Кельвіна.



Напишіть програму, яка обчислює молярну масу газу, коли користувач задає тиск, об'єм і температуру. Протестуйте свою програму на визначення кількості молей газу в балоні акваланга. Типовий балон містить 12 літрів газу під тиском 20,000,000 Па. Беремо кімнатну температуру (20°C).

3. Створіть консольний проект .NET Framework та напишіть клас, який знаходить площу чотирикутника, утвореного чотирма містами України: Києвом, Львовом, Одесою, Харковом. Програма має отримувати GPS координати (довготу та широту) міст, знаходити відстані між ними за формулою

$$d = R \cdot \arccos(\sin x_1 \sin x_2 + \cos x_1 \cos x_2 \cos(y_1 - y_2)),$$

де R – радіус Землі (в середньому 6371,01км), а потім обчислювати площу чотирикутника як суму площ двох трикутників (кожна за формулою Герона), отриманих з'єднанням двох несусідніх міст.

Рекомендується окремо виділити метод для знаходження відстані між двома населеними пунктами, а також виокремити метод для обчислення площі прямокутника.

Портуйте даний проект на .NET Core, як описано, наприклад, за посиланнями <https://docs.microsoft.com/en-us/dotnet/core/porting/>, <https://www.cafe-encounter.net/p2380/migrating-net-framework-to-netcore> та порівняйте структуру проектів, розміри виконуваних файлів тощо.

4. Виконайте завдання відповідно до свого варіанту (НомерУСпискуПідгрупи % 4 + 1):

Варіант 1	Варіант 2	Варіант 3	Варіант 4
Уявімо, що Ви тільки-но відкрили новий накопичувальний рахунок зі ставкою 14% річних. Отриманий прибуток виплачується в кінці року та додається до балансу рахунку. Напишіть програму, що починає роботу зі зчитування кількості покладених на рахунок грошей. Далі вона має обчислити та вивести на екран суму на рахунку через 1, 2 і 3 роки. Показ сум обмежується 2 знаками після крапки.	Програма в цій вправі повинна починати роботу зі зчитування вартості їжі, замовленої з ресторану (назва страви, кількість порцій, ціна за порцію). Потім програма обчислить податок та чайові за їжу. Розраховуйте на чайові в розмірі 14% від замовлення та податок у розмірі 18% від замовлення. Вивід вашої програми має включати розмір податку, чайових та загальну суму для оплати. Формат виводу передбачає представлення всіх значень з використанням 2 знаків після коми.	Компанія з виробництва енергетичних напоїв провела опитування, згідно якого близько 14% споживачів її продукції купують принаймні один напій на тиждень. 64% з таких покупців віддають перевагу цитрусовому напою. Програма повинна отримувати на вхід кількість опитаних, а на виході виводити кількість споживачів, що купують принаймні 1 напій на тиждень, а також кількість покупців, які віддають перевагу цитрусовим напоям.	Пекарня продає буханки хлібу за 8.50 грн. На вчорашній хліб встановлено знижку на 60%. Напишіть програму, яка починає зі зчитування кількості буханок вчорашнього хлібу, які хоче придбати користувач. Потім програма має вивести на екран звичайну вартість товару, скидку за те, що він вчорашній, а також загальну суму покупки. Всі значення мають виводитись двома десятковими знаками після крапки та вирівнюватись по правому краю.

- Напишіть програму, яка буде виводити, скільки днів пройшло від Вашого дня народження до поточного часу. Інформацію щодо роботи з часом можна знайти [тут](#).
- Для заданого RGB-кольору, представленого у вигляді шістнадцяткового числа, яке вводить користувач, визначте окремі значення для червоного, зеленого та синього каналів. Для цього слід використати побітові оператори.
- Напишіть програму, яка перевірятиме, чи дві клітинки шахової дошки однакового кольору, не застосовуючи оператори галуження чи циклу. Відповідь повинна бути

булевою змінною, значення якої true означає однаковість кольору, а false – відмінність. Клітинка представляється координатою з двох цілочисельних значень від 1 до 8.

8. **(Системні типи)** Виведіть таблицю з інформацією щодо значимих системних типів мови програмування C#, використовуючи програмні можливості мови (sizeof(), typeof() та ін.). Приклад подібного виводу:

Консоль отладки Microsoft Visual Studio				
Тип	Розмір у байтах	Тип .NET	Мінімальне значення	Максимальне значення
byte	1	System.Byte	0	255
bool	1	System.Boolean	False	True
sbyte	1	System.SByte	-128	127
char	2	System.Char		☐
short	2	System.Int16	-32768	32767

9. Дослідіть одну зі збірок, завантажених разом з Вашим базовим проектом відповідно до свого варіанту. Наприклад, у збірці AssemblyAnalysis знайдено тільки 1 тип – Program.

Отримати перелік збірок можна за допомогою наступного коду:

```
// Отримуємо масив збірок
Assembly[] assems = AppDomain.CurrentDomain.GetAssemblies();

Console.WriteLine("Перелік завантажених у поточний application domain збірок:");

foreach (Assembly assem in assems)
    Console.WriteLine($"Збірка {assem.ToString()}");
```

Вибір збірки для дослідження здійснюється за такою формулою:

НомерПершоїЛітериІменіВалфавіті % КількістьЗбірок + 1

Наприклад, для Володимира варіант буде $3\%6+1 = 4$, а Олексія – $19\%6+1 = 2$.

Відобразіть у звіті обчислення варіанту.

Наступний крок – визначення типів, присутніх в обраній збірці. Про інспектування збірок можна прочитати [тут](#). Метод GetTypes() формує масив типів, представлених у збірці. Варіант для вибору типу для подальшого інспектування обчислюється за формулою:

НомерПершоїЛітериПрізвищаВалфавіті % КількістьТипів + 1

Відобразіть у звіті обчислення варіанту.

Виведіть перелік типів на екран та виведіть члени цього типу. Дані щодо типу можна отримати за допомогою методу GetMembers. Приклад виводу інформації в утвореній структурі MemberInfo вже показано в посиланні вище. Загальний вигляд виводу наближено має такий вигляд:

Консоль отладки Microsoft Visual Studio

Перелік завантажених у поточний application domain збірок:

Збірка System.Private.CoreLib, Version=4.0.0.0, Culture=neutral, PublicKeyToken=7cec85d7bea7798e
Збірка AssemblyAnalysis, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
Збірка System.Runtime, Version=4.2.2.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a
Збірка System.Console, Version=4.1.2.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a
Збірка System.Runtime.Extensions, Version=4.2.2.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a
Збірка System.Threading, Version=4.1.2.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a

Досліджуємо збірку AssemblyAnalysis, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
Назва типу: Program

Досліджуємо тип Program

Member type: Method, member name: GetType.
Member type: Method, member name: ToString.
Member type: Method, member name: Equals.
Member type: Method, member name: GetHashCode.
Member type: Constructor, member name: .ctor.