

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«УЖГОРОДСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ»
ФАКУЛЬТЕТ МАТЕМАТИКИ ТА ЦИФРОВИХ ТЕХНОЛОГІЙ
Кафедра системного аналізу та теорії оптимізації

А.Ю.Брила, М.М. Ломага, П.П. Антосяк

ТИПИ ДАНИХ. ЗМІННІ, КОНСТАНТИ У C#

Методичні матеріали з організації самостійної роботи студентів з дисципліни
«Програмування»

Ужгород 2023

Типи даних, змінні, константи у С#. (Методичні матеріали з організації самостійної роботи студентів з дисципліни «Програмування»). /Укладачі: А.Ю. Брила, М.М. Ломага, П.П. Антосяк. – Ужгород, 2023.– 32 с.

Навчальний матеріал методичних вказівок призначений для аудиторної і самостійної підготовки студентів при вивченні дисципліни «Програмування».

Основна мета самостійної роботи студента – закріплення теоретичних відомостей, які викладаються на лекціях, та вміння їх застосувати, розв’язуючи задачі, що виникають на практиці. Завдання розроблених методичних матеріалів полягає в чіткій, цілеспрямованій допомозі студентам в організації самостійної підготовки до практичних занять з дисципліни «Програмування». Розроблена система тестових питань по темі «Вступ. Типи даних. Змінні, константи» дає змогу поглибити знання та вміння їх застосовувати при розробці алгоритмів.

Методичні вказівки призначені для студентів різних напрямків підготовки.

Рецензенти:

к.ф.-м.н., доц. Погоріляк О.О.,

к.т.н., доц. Андрашко Ю.В.

Рекомендовано до друку:

Кафедрою системного аналізу та теорії оптимізації (Протокол №10 від 18 травня 2023 року);

Науково-методичною комісією факультету математики та цифрових технологій, (Протокол № 9 від 23 травня 2023 року);

Вченою радою факультету математики та цифрових технологій ДВНЗ “Ужгородський національний університет”, (Протокол №9 від 25 травня 2023 року).

ВСТУП. ТИПИ ДАНИХ. ЗМІННІ, КОНСТАНТИ

1. ТЕХНОЛОГІЯ .NET FRAMEWORK

Технологія .Net Framework представляє собою середовище, яке дає можливість розробляти і виконувати програми незалежно від платформи (операційної системи). Вона дозволяє розробляти проекти, частини яких можуть бути виконані у різних мовах програмування. Основною мовою програмування цієї технології є мова C#. У цій технології єдиний спосіб опису програмного коду (метадані), одне середовище виконання (Common Language Runtime) і одна базова бібліотека (BCL).

Міжмовна інтеграція досягається за рахунок того, що програма, написана на мові високого рівня (C# чи Basic, чи C++, чи ін.) не одразу компілюється у машинний код, а спочатку відбувається компіляція у команди (псевдокод) єдиної проміжкової мови Intermediate Language (IL).

Незалежність від платформи досягається за рахунок того, що під час виконання програми команди проміжкової мови перетворюються у машинні команди у середовищі Common Language Runtime (CLR) за допомогою відповідних компіляторів (JIT-компіляторів).

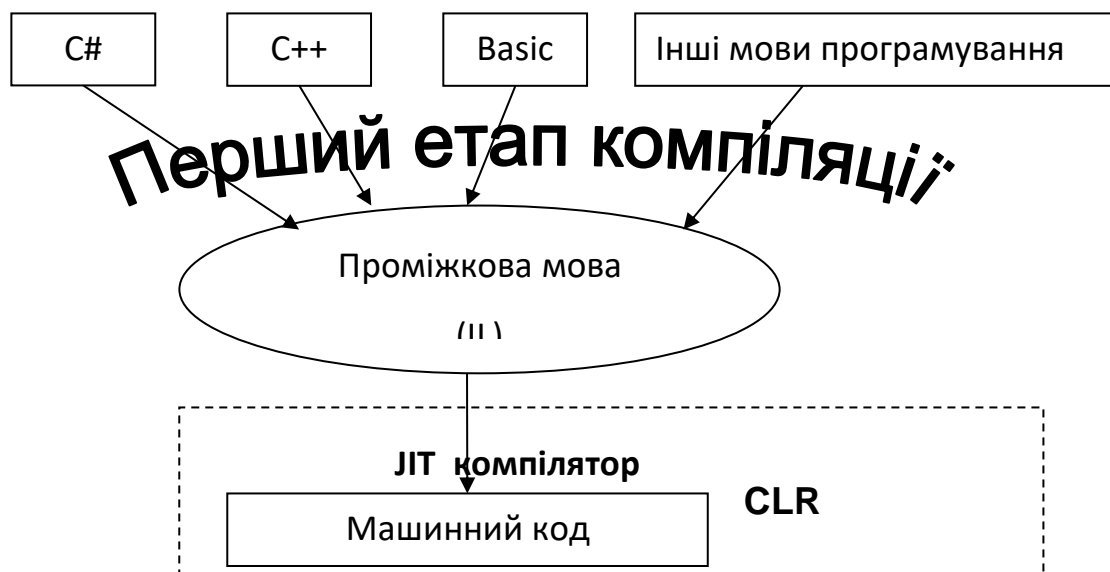


Рисунок 1. Схема виконання програми у .NET

Отже, технологія .Net Framework реалізована в середовищі Common Language Runtime (CLR). CLR можна інтерпретувати, як деякий препроцесор, який перетворює команди проміжкової мови Intermediate Language (IL) в команди процесора. При цьому використання CLR для розробки програм забезпечує:

- міжмовну інтеграцію;
- автоматичне керування пам'яттю (збирання сміття);
- контроль типів;
- контроль версій програм.

Більшість із цих можливостей можливі завдяки тому, що CLR реалізовано з використанням метаданих. Метадані представляють собою об'єднання програм і даних в одному файлі формату Portable Executable (PE). Програма на проміжковій мові містить не тільки сам програмний код, але і метадані, які надають середовищу виконання інформацію про всі типи, що використовуються, всі сигнатури, всіх елементів цих типів та іншу інформацію, яка в програмах з використанням COM поміщалась в бібліотеки типів і системний реєстр. Таким чином, програма і вся необхідна їй інформація розміщена в одному файлі, що суттєво полегшує установку програми, оскільки виключає реєстрацію в системному реєстрі і пошук відповідних бібліотек.

Як було зазначено, C# є основною мовою програмування технології .Net Framework. Програма може розроблятися і на інших мовах програмування, але при цьому необхідно дотримуватися певних правил, що визначено у середовищі CLR. Так правила описання і використання типів у середовищі CLR визначаються системою віртуальних об'єктів (NGWS Virtual Object System або ж VOS). Використання системи VOS спрощує міжмовну інтеграцію і забезпечує безпечність типів без суттєво зниження роботи програми. В мові використовуються чотири основні складові VOS:

1. Система типів – призначена для уніфікації типів даних, які використовуються різними мовами програмування.

2. Метадані – містять описання і посилання на типи, які визначені в системі.
3. Common Language Specification – специфікація єдиної мови.
4. Virtual Execution System – віртуальне середовище виконання. Це середовище представляє собою реалізацію VOS і відповідає за завантаження і виконання програм, які використовують CLR.

Питання для самоконтролю

1. Що собою являє технологія *.Net Framework*?
2. За рахунок чого досягається міжмовна інтеграція у *.Net Framework*?
3. За рахунок чого досягається незалежність від платформи у *.Net Framework*?
4. Які основні можливості надає середовище *Common Language Runtime*?
5. Яка мова програмування є основною мовою програмування технології *.Net Framework*?

2. МОВА ПРОГРАМУВАННЯ. ПОНЯТТЯ ПРОГРАМИ

З формальної точки зору **мова програмування (МП)** – це набір вихідних символів (*алфавіт*) разом із системою правил утворення з цих символів формальних конструкцій (*синтаксис*) та системою правил їх тлумачення (інтерпретації) (*семантика*), за допомогою яких описуються алгоритми.

Алфавіт, синтаксис та семантика – це три основні складові частини будь-якої МП. До **алфавіту** МП, як правило, входять літери латинського алфавіту, арабські цифри, знаки арифметичних операцій, розділові знаки, спеціальні символи. Із символів алфавіту будують послідовності, які називають **словами (лексемами)**. Кожне слово у МП має своє змістове призначення. **Правила синтаксису** пояснюють, як потрібно будувати ті чи інші мовні повідомлення для опису всіх понять мови, здійснення описів та запису вказівок. **Правила семантики** пояснюють, яке призначення має кожен опис та які дії повинна виконати обчислювальна машина під час виконання кожної із вказівок. Вказівки на виконання конкретних дій називають ще **командами** або **операторами** мови.

Текст алгоритму, записаний засобами конкретної мови програмування, називають *програмою*.

Усі слова або ж лексеми, поділяють на

- *службові* (зарезервовані або ж ключові) *слова*;
- *стандартні ідентифікатори*;
- *ідентифікатори користувача*;
- *знаки операцій*;
- *літерали*;
- *розділові знаки*.

Службові слова мають наперед визначене призначення і використовуються для формування структури програми, здійснення описів, позначення операцій, формування керуючих конструкцій (вказівок). Наприклад, службовими словами для мови C# є: `abstract, do, in, protected, as, public, try, base, else, interface, readonly, typeof, enum, internal, ref, break, event, is, return, byte, explicit, lock, unchecked, case, extern, sealed, unsafe, catch, namespace, finally, new, sizeof, using, checked, fixed, alloc, virtual, class, const, for, operator, volatile, continue, foreach, out, struct, while, goto, override, switch, default, if, params, this, delegate, implicit, private, throw`.

Імена (позначення) для програмних об'єктів (типів даних, констант, змінних, функцій і т. п.) формують у вигляді ідентифікаторів. *Ідентифікатор* – це послідовність латинських літер, цифр і знаку підкреслення, яка розпочинається з латинської літери. У мові C# максимальна довжина ідентифікатора є необмеженою.

Стандартні ідентифікатори використовуються як імена для стандартних (передбачених авторами мови програмування) типів даних, констант, підпрограм (зокрема, стандартних математичних функцій). Приклад: `int, long, double, char, bool` та ін.

Ідентифікатори користувача є іменами тих програмних об'єктів (констант, змінних, функцій тощо), які створює сам користувач. Службові слова та ідентифікатори користувача не повинні збігатися.

Коментар – невиконувана частина тексту програми, що ігнорується компілятором і служить для вставки деяких поміток у програмі тільки для програміста. Коментарі бувають однорядковими та багаторядковими. Однорядковий коментар починається з символів «//» і закінчується у кінці рядка. Тобто всі символи до кінця рядка вважаються коментарем.

Приклад:

// Це однорядковий коментар

Багаторядковий коментар починається з символів «/*» і закінчується символами «*/».

Приклад:

/* Це
багаторядковий
коментар */

Літерал – це явно вказане значення деякого типу. Розрізняють такі типи літералів:

Тип літерала	Опис	Приклади
цілочисловий	у десятковій системі числення – звичне нам ціле число	125, -89, 108;
	у вісімковій системі числення – починається з 0	023, 075, 0416;
	у шістнадцятковій системі числення – починається з 0x або 0X	0x4A, 0x6FE2, 0xABC
дійсного типу	у форматі з фіксованою крапкою – у записі числа є крапка, що розділяє цілу і дробову частини	25.69, 2.0, 145.058
	у форматі з плаваючою крапкою – у записі використано символ «e» або «E», що розділяє мантису від порядку	3.5e5, 3e12, 678e2
символьний	заданий у явному вигляді – символ записується у одинарних лапках	'Z', 'a', 'E'

Тип літерала	Опис	Приклади
	прості ескейп-послідовності – службові символи починаються з символу «\»	\n - перехід на новий рядок, \t - горизонтальна табуляція, \' - апостроф, і т.д.
	ескейп-послідовності Unicode – символи «\u», за якими вказують код символу з чотирьох цифр у шістнадцятковій системі числення	'\u0123', '\u3A58'
рядковий	дослівний – починається із символу «@», за яким у подвійних лапках вказується рядок символів (при цьому ескейп-послідовності сприймаються як символи)	@”Я люблю \t C#”
	регулярний – у подвійних лапках вказується рядок символів (ескейп-послідовності обробляються)	”Я люблю \t C#”
логічний	може приймати два значення	true, false
порожньої адреси (нульовий літерал)	використовується у випадку, коли покажчик не містить жодної адреси	null

Розділові знаки служать для відокремлення однієї лексеми від іншої. Ними є пробіл, табуляція, символ нового рядка, символ «;» та коментарі.

Питання для самоконтролю

1. Що таке мова програмування?
2. Які три основні складові частини будь-якої мови програмування?
3. Що таке програма?
4. Назвіть основні різновиди лексем.
5. Для чого призначені службові слова?
6. Що таке ідентифікатор?
7. Для чого використовують у програмі коментарі?
8. Що таке літерал?
9. Назвіть основні типи літералів.

3. ВЕЛИЧИНИ. ТИП ВЕЛИЧИН

Поняття величини та типу

У своїй роботі програміст має справу з таким поняттям, як величина. З точки зору програмування *величини* – це дані, що обробляються програмами. *Дані* – це інформація, введена у пам'ять комп'ютера або підготовлена до введення.

Носіями даних у програмах є *константи*, *змінні* (значення яких зберігається в оперативній пам'яті) та *файли* (на зовнішніх носіях інформації).

Константи – це величини, значення яких у процесі виконання програми не змінюється. *Змінні* — це величини, значення яких у процесі виконання програми можуть змінюватися. Імена констант і змінних, як і інших програмних об'єктів, записують у формі ідентифікаторів. Кожна змінна і константа належать до визначеного типу.

Тип даних – це сукупність властивостей певного набору даних, від яких залежать: діапазон значень, якого можуть набувати ці дані, а також сукупність операцій, які можна виконувати над цими даними.

З іншого боку **тип даних** – це описання того, яку структуру, розмір мають комірки оперативної пам'яті при зберіганні відповідного елемента даних.

Елемент даних певного типу – це комірка або комірки оперативної пам'яті, що мають фіксовану адресу, розряди яких розшифровуються згідно описання даного типу даних.

З кожним типом даних зв'язано своє унікальне ім'я (ідентифікатор), яке є синонімом певного описання елемента даних відповідного типу. Наприклад, ідентифікатор `byte` є синонімом опису: 8 послідовних розрядів містить ціле значення без знаку в діапазоні від 0 до 255 (у двійковому вигляді займає 1 байт).

Тип констант визначається компілятором автоматично, але позначивши змінну ідентифікатором, ще треба вказати її тип.

Слід зазначити, що тип даних – це абстрактне описання і тому прямо використовуватися він не може. Його використання здійснюється через елементи даних відповідного типу. Наприклад, елемент даних типу `byte` – це комірка оперативної пам'яті з певною адресою, розряди якої декодуються або

розуміються згідно описання типу `byte`. Оскільки безпосередньо, через вказівку адрес, працювати з елементами даних незручно, то кожному елементу даних ставиться у відповідність ідентифікатор (ім'я змінної чи константи), користуючись яким можна здійснювати доступ до елемента даних.

У зв'язку з цим можна дати інше означення константи та змінної.

Якщо елемент даних не може змінювати свого значення, тобто завжди містить одне і те ж саме значення, то відповідний ідентифікатор називається *константою даного типу*. Якщо елемент даних певного типу може змінювати своє значення під час виконання програми, то ідентифікатор, що зв'язаний з цим елементом даних називається змінною відповідного типу. *Значення змінної* – це елемент даних, з якими ця змінна пов'язана.

Отже, у програмах змінна характеризується такими ознаками: *іменем, типом і значенням*.

Стандартні типи у C#

В мові підтримується стандартний набір типів даних. Однак кожен тип даних є об'єктом, що не передбачається в стандартному наборі. Для вирішення цього протиріччя в мові розрізняють: *типи значень* і *типи посилань*. Типи значень включають в себе прості типи змінних, а також переліковний тип даних і структури, тобто ті типи, які зазвичай не є об'єктами. До простих типів відносяться, наприклад, такі типи як `char`, `int`, `float`. Щоб перетворити їх в об'єкти і одночасно забезпечити їх використання в тих мовах, де вони не є об'єктами, кожному простому типу ставиться у відповідність об'єкт `CLS`.

До типів посилань відносяться типи класів, інтерфейсів, масивів та делегатів.

До стандартних «простих» типів у мові C# відносяться наступні:

Тип	Опис	Область значень
<code>object</code>	Базовий клас для всіх інших типів	
<code>string</code>	Рядковий тип, послідовність символів Unicode	
<code>sbyte</code>	8-розрядне ціле число з знаком	-128 до 127
<code>short</code>	16-розрядне ціле число з знаком	-32768 до 32767
<code>int</code>	32-розрядне ціле число з знаком	-2147483648 до 2147483647

Тип	Опис	Область значень
long	64-розрядне ціле число з знаком	-9223372036854775808 до 9223372036854775807
byte	8-розрядне ціле число без знака	0 до 255
ushort	16-розрядне ціле число без знака	0 до 65535
uint	32-розрядне ціле число без знака	0 до 4294967295
ulong	64-розрядне ціле число без знака	0 до 18446744073709551615
float	Число з плаваючою крапкою 4 байти, точність — 7 розрядів	$\pm 1,5 \cdot 10^{-45}$ до $\pm 3,4 \cdot 10^{33}$
double	Число з плаваючою крапкою 8 байт, точність — 16 розрядів	$\pm 5 \cdot 10^{-324}$ до $\pm 1,7 \cdot 10^{306}$
bool	Логічний тип	true або false
char	Тип символу Unicode	U+0000 до U+ffff
decimal	Тип десяткового числа 12 байт, точність — 28 розрядів	

Зазначимо, що кожен із наведених типів є скороченням деякого системного типу. Наприклад, тип `Int` є альтернативною назвою структури `System.Int32`. В більшості випадків ці імена повністю рівноправні.

Опис змінних та констант

Опис змінних

Загальне правило опису:

- *без початкової ініціалізації*

`<тип змінної> <ідентифікатор змінної>;`

- *з початковою ініціалізацією*

`<тип змінної> <ідентифікатор змінної> = <значення>;`

Приклади:

```
int a;           // Без початкової ініціалізації.
int b=40;       // Змінна b ініціалізовується значенням 40.
double s=3.67;  // Змінна s ініціалізовується значенням 3.67.
```

Опис констант

Загальне правило опису:

`const <тип константи> <ідентифікатор константи> = <значення>;`

Приклади:

```
const int a = 10;    // константа цілого типу
const float b = 2.5; // константа дійсного типу
```

Перетворення типів

В мові розрізняють два види перетворень (приведень) змінних – *явне* та *неявне*.

Неявне перетворення типів використовується в тому випадку, коли дане перетворення є «природним». Тобто, наприклад, перетворюється величина типу `float` в величину типу `double`.

Приклад:

```
float f = 1.23;
double d = f; // Неявне перетворення
```

Таке перетворення є природним, оскільки обидва типи використовуються для представлення дійсних типів, причому цільовий тип має більший діапазон представлення і більшу точність. При такому перетворення не відбувається втрата інформації. При проведенні неявного перетворення немає необхідності вказувати цільовий тип (тип до якого здійснюється перетворення).

Явне перетворення типів вимагає явного задання цільового типу, до якого здійснюється перетворення. Цільовий тип вказується в дужках перед значенням тип якого перетворюється.

Загальне правило явного перетворення:

```
<змінна> = (<цільовий тип>) <вираз>;
```

Приклад:

```
double d = 2.9;
float f = (float)d; // Явне перетворення
```

Для цільового типу повинно існувати неявне перетворення до типу змінної, в якій буде збережено результат.

Переліковний тип

Переліковний тип це тип, який задається повним переліком свої значень. Значеннями перелікового типу є іменовані константи типів `byte`, `sbyte`, `short`, `ushort`, `int`, `uint`, `long`, `ulong`. Якщо при описанні перелікового типу в явному вигляді не вказано тип його констант, то вважається, що константи типу `int`. Всі константи повинні бути одного типу, який називається *базовим типом*. Тип `char`

не може використовуватися в якості базового типу. В переліковному типі можна задавати довільну кількість іменованих констант.

Загальне правило опису:

```
<атрибути> <модифікатори> enum <ім'я типу> [: <базовий тип>]
{
    <атрибути> <іменована конст.1> [= <знач.1>],
    <атрибути> <іменована конст.2> [= <знач.2>],
    .....
    <атрибути> <іменована конст.N> [= <знач.N>]
}
```

Приклад:

```
enum Week : Byte
{
    Monday,        // =0
    Tuesday,       // =1
    Wednesday,     // =2
    Thursday,      // =3
    Friday,        // =4
    Saturday,      // =5
    Sunday         // =6
}
```

Всі іменовані константи перелікового типу ініціалізуються значеннями базового типу. Ця ініціалізація може бути *явною* або *неявною*. За замовчуванням застосовується неявна ініціалізація (дивись приклад вище), при цьому, перша іменована константа у переліку ініціалізується нулем, друга – одиницею, і т.д.

Також іменовані константи можуть бути проініціалізовані явно довільними значеннями із базового типу або ж значенням іншої іменованої константи із цього ж переліку. При цьому ініціалізація іменованих констант не обов'язково повинна здійснюватися послідовними значеннями, не обов'язково ініціалізувати явно всі іменовані константи та не обов'язково ініціалізувати іменовані константи унікальними значеннями.

Приклад:

```
enum Week : Byte
{
    Monday    =2,
    Tuesday   =35,
    Wednesday =2,
    Thursday,           // =3
}
```

```

    Friday,           // =4
    Saturday,        // =5
    Sunday =1,
    Sabbath =Saturday // =5
}

```

Ініціалізація змінної перелікового типу значенням здійснюється через ім'я типу та оператор "." (крапка).

Приклад:

```
Week d = Week.Friday;
```

Також між змінними перелікового типу та змінними відповідного базового типу існує явне перетворення типів.

Приклади:

```

Week d = Week.Monday;
byte b = (byte)d;           // b = 2
byte i = 1;
Week s = (Week)i;          // s = Week.Sunday

```

Питання для самоконтролю

1. Що розуміють під терміном "величина" у програмуванні?
2. Дайте означення константи та змінної.
3. Дайте означення типу даних.
4. На які два різновиди поділяють усі типи у мові C#?
5. Які типи належать до типів посилань?
6. Назвіть та опишіть прості типи мови C#.
7. Яке загальне правило опису змінних?
8. Яке загальне правило опису констант?
9. Яке загальне правило опису перелікового типу?
10. У яких випадках використовується неявне перетворення типів?
11. Який загальний синтаксис явного перетворення типів?

НАБІР ТЕСТІВ ДЛЯ САМОКОНТРОЛЮ

Типи даних

1.	Які із вказаних типів даних можуть бути використані для збереження року народження студента? 1) char; 2) string; 3) byte; 4) int; 5) integer; 6) Немає правильної відповіді.
2.	Які із вказаних типів даних можуть бути використані для збереження віку студента? 1) char; 2) string; 3) byte; 4) int; 5) integer; 6) Немає правильної відповіді.
3.	Які із вказаних типів даних можуть бути використані для збереження ваги студента (у кілограмах без заокруглення)? 1) char; 2) string; 3) byte; 4) int; 5) double; 6) Немає правильної відповіді.
4.	Які із вказаних типів даних можуть бути використані для збереження імені студента? 1) char; 2) string; 3) byte; 4) int; 5) double; 6) Немає правильної відповіді.
5.	Які із вказаних типів даних можуть бути використані для збереження ініціалів студента (ім'я, по батькові)? 1) symbols; 2) string; 3) byte; 4) int; 5) double; 6) Немає правильної відповіді.
6.	Літерал якого типу вказують у одинарних лапках (наприклад, 'F')? 1) char; 2) string; 3) byte; 4) int; 5) double; 6) Немає правильної відповіді.
7.	Змінній яких типів можна присвоїти значення 'B' ? 1) char; 2) string; 3) byte; 4) int; 5) double; 6) Немає правильної відповіді.
8.	Змінній яких типів можна присвоїти значення "B" ? 1) char; 2) string; 3) byte; 4) int; 5) double; 6) Немає правильної відповіді.
9.	Змінній яких типів можна присвоїти значення "B" ? 1) char; 2) string; 3) byte; 4) int; 5) double; 6) Немає правильної відповіді.
10.	Змінній яких типів можна присвоїти значення '\n' ? 1) char; 2) string; 3) byte; 4) int; 5) double; 6) Немає правильної відповіді.
11.	Якого типу є літерал цілого типу 27 ? 1) sbyte; 2) short; 3) byte; 4) uint; 5) long; 6) Немає правильної відповіді.
12.	Якого типу є літерал цілого типу -27 ?

	1) sbyte; 2) short; 3) byte; 4) uint; 5) long; 6) Немає правильної відповіді.
13.	Якого типу є літерал 2.7? 1) float; 2) decimal; 3) double; 4) Немає правильної відповіді.
14.	Які із вказаних типів даних є знаковими? 1) sbyte; 2) short; 3) byte; 4) uint; 5) long; 6) Жоден із перерахованих.
15.	Які із вказаних типів даних є беззнаковими? 1) sbyte; 2) short; 3) byte; 4) uint; 5) long; 6) Жоден із перерахованих.
16.	Яке найбільше значення може містити змінна цілого знакового типу, якщо для її збереження виділено 4 байти. 1) 2^{32} ; 2) $2^{32} - 1$; 3) $2^{31} - 1$; 4) Немає правильної відповіді.
17.	Яке найбільше значення може містити змінна цілого беззнакового типу, якщо для її збереження виділено 4 байти. 1) 2^{32} ; 2) $2^{32} - 1$; 3) $2^{31} - 1$; 4) Немає правильної відповіді.
18.	Продовжіть твердження. Для дійсних типів даних кількість значущих цифр – це ... 1) наперед визначений список цифр, які можуть бути збережені; 2) наперед визначений список цифр, які не можуть бути збережені; 3) кількість розрядів у числі; 4) Немає правильної відповіді.
19.	При збереженні значень якого типу даних одним із елементів, який треба зберегти, є мантиса? 1) Цілого; 2) Дійсного; 3) Символьного; 4) Цілого та дійсного; 5) Немає правильної відповіді.
20.	Які із елементів потрібно зберегти при збереженні значення типу double? 1) знак; 2) кількість цифр; 3) окремо цілу і дробову частину; 4) пункти 1) і 3); 5) порядок; 6) мантису; 7) Немає правильної відповіді.
21.	Які із вказаних значень можна присвоїти змінній типу bool? 1) true; 2) TRUE; 3) True; 4) Жодного
22.	Які із вказаних значень можна присвоїти змінній типу bool? 1) FALSE; 2) notTrue; 3) False; 4) Жодного.
23.	Які із перерахованих типів даних можна використати при розв'язанні вказаної задачі? «Знайти загальну вартість n холодильників, якщо ціна кожного дорівнює R (ціну вказано у тисячах гривень без заокруглення)». 1) char; 2) string; 3) bool; 4) int; 5) double; 6) Немає правильної відповіді.
24.	Які із перерахованих типів даних можна використати для розв'язання вказаної задачі? «Знайти загальну кількість студентів, які знаходяться у 3-х аудиторіях».

	1) char;2) string;3) bool;4) int; 5) integer; 6) Немає правильної відповіді.
25.	Які із перерахованих типів даних можна використати для розв'язання вказаної задачі (без виконання операції приведення типів)? «З клавіатури вводиться ціле число. Знайти остачу від ділення на 5». 1) char;2) string;3) bool;4) int; 5) double; 6) Немає правильної відповіді.
26.	Які із перерахованих типів даних можна використати для розв'язання вказаної задачі? «З клавіатури вводиться цифра у шістнадцятковій системі числення. Знайти відповідне значення цієї цифри у десятковій системі числення». 1) char;2) float;3) bool;4) int; 5) double; 6) Немає правильної відповіді.
27.	Зміна якого із вказаних типів даних може зберегти результат обчислення значення вказаного виразу? <code>a++ > b</code> 1) char;2) string;3) bool;4) int; 5) double; 6) Немає правильної відповіді.
28.	Зміна якого із вказаних типів даних може зберегти результат обчислення значення вказаного виразу? <code>(a++) / (double)(b++)</code> 1) char;2) string;3) bool;4) int; 5) double; 6) Немає правильної відповіді.
29.	Зміна якого із вказаних типів даних може зберегти результат обчислення значення вказаного виразу? "Peace "+"Freedom" 1) char;2) string;3) bool;4) int; 5) double; 6) Немає правильної відповіді.
30.	Зміна якого із вказаних типів даних може зберегти результат обчислення значення вказаного виразу? <code>'W'+ " letter"</code> 1) char;2) string;3) bool;4) int; 5) double; 6) Немає правильної відповіді.
31.	Зміна якого із вказаних типів даних може зберегти результат обчислення значення вказаного виразу? "not"+ " true" 1) char;2) string;3) bool;4) int; 5) double; 6) Немає правильної відповіді.
32.	Зміна якого із вказаних типів даних може зберегти результат обчислення значення вказаного виразу? "false"+ " true" 1) char;2) string;3) bool;4) int; 5) double; 6) Немає правильної відповіді.

Змінні, константи

1.	<p>Що дозволяє описати вказана загальна форма? <code>[тип] [ідентифікатор] ;</code> 1) Змінну; 2) Константу; 3) Літерал; 4) Немає правильної відповіді.</p>
2.	<p>Що дозволяє описати вказана загальна форма? <code>[тип] [ідентифікатор] = [значення] ;</code> 1) Змінну; 2) Константу; 3) Літерал; 4) Немає правильної відповіді.</p>
3.	<p>Яке значення буде містити змінна a? <code>int a=23;</code> <code>a=27;</code> 1) 23; 2) 27; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
4.	<p>Яке значення буде містити змінна a? <code>int a;</code> <code>a=23;</code> 1) 23; 2) 27; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
5.	<p>Що дозволяє описати вказана загальна форма? <code>const [тип] [ідентифікатор] = [значення] ;</code> 1) Змінну; 2) Константу; 3) Літерал; 4) Немає правильної відповіді.</p>
6.	<p>Що дозволяє описати вказаний фрагмент програми? <code>const [тип] [ідентифікатор];</code> <code>[ідентифікатор] = [значення];</code> 1) Змінну; 2) Константу; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
7.	<p>Яке значення буде містити A? <code>const int A=23;</code> <code>A=27;</code> 1) 23; 2) 27; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
8.	<p>Яке значення буде містити A? <code>const int A=4+8;</code> 1) 4; 2) 8; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
9.	<p>У якому з варіантів вказано опис змінної цілого типу? 1) <code>a=12;</code> 2) <code>a=0;</code> 3) <code>int[]a;</code> 4) Немає правильної відповіді</p>
10.	<p>У якому з варіантів вказано опис константи дійсного типу? 1) <code>a=1;</code> 2) <code>a=0;</code> 3) <code>const double a=1;</code> 4) Немає правильної відповіді</p>
11.	<p>У якому з варіантів вказано повний опис константи цілого типу? 1) <code>a=1;</code> 2) <code>a=0;</code> 3) <code>const int a;</code> 4) Немає правильної відповіді.</p>

12.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>int a=7;</pre> <p>1) Описано константу цілого типу; 2) Описано змінну цілого типу; 3) Описано літерал цілого типу; 4) Немає правильної відповіді.</p>
13.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>int b=9, a=3, c=4;</pre> <p>1) Описано константи цілого типу; 2) Описано змінні цілого типу; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
14.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>int b, a=3, c;</pre> <p>1) Описано константи цілого типу; 2) Описано змінні цілого типу; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
15.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>const int b=9, a, c=4;</pre> <p>1) Описано константи цілого типу; 2) Описано змінні цілого типу; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
16.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>const int b=9, a=3, c=7;</pre> <p>1) Описано константи цілого типу; 2) Описано змінні цілого типу; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
17.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>const int B=9; int d=B;</pre> <p>1) Описано константу цілого типу; 2) Описано змінну цілого типу; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
18.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>int d=9; const int B=d;</pre> <p>1) Описано константу цілого типу; 2) Описано константу логічного типу; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
19.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>const int B=9; double d=B;</pre> <p>1) Описано константу; 2) Описано змінну; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
20.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>const int B = 45; const int C = B;</pre> <p>1) Описано константи; 2) Описано змінні; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
21.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>const int B = 45; const int C = B;</pre>

	<pre>const int D= B+C;</pre> <p>1) Описано константи; 2) Описано змінні; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
22.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>const int B = 45; const int C = B; const int D= B+8;</pre> <p>1) Описано константи; 2) Описано змінні; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
23.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>const char R= "A";</pre> <p>1) Описано константу символьного типу; 2) Описано змінну символьного типу; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
24.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>const char R= 'A';</pre> <p>1) Описано константу символьного типу; 2) Описано змінну символьного типу; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
25.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>char R= "A";</pre> <p>1) Описано константу символьного типу; 2) Описано змінну символьного типу; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
26.	<p>Охарактеризуйте вказаний фрагмент програми.</p> <pre>char R= 'A';</pre> <p>1) Описано константу символьного типу; 2) Описано змінну символьного типу; 3) Фрагмент містить помилку; 4) Немає правильної відповіді.</p>
27.	<p>Який стиль було використано при формуванні імені totalSum ?</p> <p>1) Camel-case; 2) Snake-case; 3) Kebab-case; 4) Немає правильної відповіді.</p>
28.	<p>Який стиль було використано при формуванні імені TOTAL_SUM ?</p> <p>1) Camel-case; 2) Snake-case; 3) Kebab-case; 4) Немає правильної відповіді.</p>
29.	<p>Який стиль було використано при формуванні імені константи TOTAL-SUM?</p> <p>1) Camel-case; 2) Snake-case; 3) Недопустиме ім'я; 4) Немає правильної відповіді.</p>
30.	<p>Який стиль було використано при формуванні імені змінної total-Sum ?</p> <p>1) Camel-case; 2) Snake-case; 3) Недопустиме ім'я; 4) Немає правильної відповіді.</p>
31.	<p>При іменуванні чого прийнято використовувати стиль Snake-case?</p> <p>1) Змінних; 2) Констант; 3) Змінних і констант; 4) Немає правильної відповіді.</p>
32.	<p>При іменуванні чого прийнято використовувати стиль Camel-case?</p>

	1) Змінних; 2) Констант; 3)Змінних і констант; 4) Немає правильної відповіді.
33.	Продовжіть твердження, доповнивши його двома пунктами із вказаного списку. "Областю видимості змінної є пункт 1 , пункт 2 ." Вибір пункту 1: 1) рядок; 2) частина оперативної пам'яті; 3) частина блоку. Вибір пункту 2: 4) що знаходиться після опису змінної; 5) що знаходиться перед описом змінної.
34.	У яких рядках вказаного фрагмента програми допущено помилки? 1) { 2) int a=9; 3) a=45; 4) a=23; 5) } 6) a=89; 7) Вказаний фрагмент не містить помилки.
35.	У яких рядках вказаного фрагмента програми допущено помилки? 1) { 2) int a=9; 3) { 4) a=45; 5) } 6) a=23; 7) } 8) Вказаний фрагмент не містить помилки.
36.	У яких рядках вказаного фрагмента програми допущено помилки? 1) { 2) int a=9; 3) { 4) a=45; 5) } 6) int a=23; 7) } 8) Вказаний фрагмент не містить помилки.
37.	У яких рядках вказаного фрагмента програми допущено помилки? 1) { 2) int a=9; 3) { 4) a=45; 5) } 6) double a=23; 7) } 8) вказаний фрагмент не містить помилки.
38.	У яких рядках вказаного фрагмента програми допущено помилки?

	<pre> 1) { 2) int a=9; 3) a=45; 4) } 5) { 6) double a=23; 7) a=3.7; 8) } 9) Вказаний фрагмент не містить помилки.</pre>
39.	<p>У якому із вказаних (пронумерованих) рядків необхідно додати один опис змінної <code>a</code> для того, щоб вказаний фрагмент програми був коректним?</p> <pre> { 1) 2) a=45; } { 3) 4) a=3.7; } 5) Це неможливо зробити використовуючи вказані пронумеровані рядки програми.</pre>
40.	<p>У якому із вказаних (пронумерованих) рядків необхідно додати один опис змінної <code>a</code> для того, щоб вказаний фрагмент програми був коректним?</p> <pre> 1) { 2) 3) a=45; } { 4) 5) a=3.7; } 6) це неможливо зробити використовуючи вказані пронумеровані рядки програми</pre>

Приклади опису

1.	<p>Описати змінну цілого типу з використанням стилю <code>Camel-case</code> для зберігання кількості студентів.</p> <pre> 1) int studentsCount; 2) int STUDENTS_COUNT; 3) int STUDENTS_count; 4) int students_COUNT; 5) Немає правильної відповіді.</pre>
2.	<p>Описати змінну цілого типу з використанням стилю <code>Camel-case</code> для зберігання кількості студентів.</p> <pre> 1) BYTE studentsCount; 2) bYte STUDENTS_COUNT; 3) byte STUDENTS_count;</pre>

	<p>4) <code>byte students_COUNT;</code> 5) Немає правильної відповіді.</p>
3.	<p>Описати константу цілого типу з використанням стилю Snake-case для зберігання кількості студентів .</p> <p>1) <code>const int studentsCount=10;</code> 2) <code>const int STUDENTS_COUNT=30;</code> 3) <code>const int STUDENTS_COUNT; STUDENTS_COUNT=9;</code> 4) <code>const int students_COUNT;</code> 5) Немає правильної відповіді.</p>
4.	<p>Описати константу цілого типу .</p> <p>1) <code>constInt studentsCount=10;</code> 2) <code>const int STUDENTS-COUNT=30;</code> 3) <code>const int STUDENTS_COUNT; STUDENTS_COUNT=9;</code> 4) <code>const int students_COUNT;</code> 5) Немає правильної відповіді.</p>
5.	<p>Описати змінну цілого типу .</p> <p>1) <code>int students Count;</code> 2) <code>int STUDENTS-COUNT;</code> 3) <code>int STUDENTS-count;</code> 4) <code>int students~COUNT;</code> 5) Немає правильної відповіді.</p>
6.	<p>Описати змінну цілого типу з використанням стилю Camel-case для зберігання кількості студентів .</p> <p>1) <code>studentsCount int;</code> 2) <code>STUDENTS_COUNT int;</code> 3) <code>STUDENTS_count int;</code> 4) <code>students_COUNT int;</code> 5) Немає правильної відповіді.</p>
7.	<p>Описати константу цілого типу з використанням стилю Snake-case для зберігання кількості студентів .</p> <p>1) <code>int const studentsCount=10;</code> 2) <code>int const STUDENTS_COUNT=30;</code> 3) <code>int const STUDENTS_COUNT; STUDENTS_COUNT=9;</code> 4) <code>int const students_COUNT;</code> 5) Немає правильної відповіді.</p>
8.	<p>Описати константу цілого типу .</p> <p>1) <code>const studentsCount=10;</code> 2) <code>const STUDENTS-COUNT=30;</code> 3) <code>const STUDENTS_COUNT; STUDENTS_COUNT=9;</code> 4) <code>const students_COUNT;</code> 5) Немає правильної відповіді.</p>
9.	<p>Описати змінну дійсного типу .</p> <p>1) <code>int totalPrice;</code> 2) <code>double totalPrice;</code> 3) <code>bool totalPrice;</code></p>

	<p>4) <code>real totalPrice;</code> 5) Немає правильної відповіді.</p>
10.	<p>Описати змінну дійсного типу . 1) <code>int totalPrice;</code> 2) <code>float totalPrice;</code> 3) <code>bool totalPrice;</code> 4) <code>real totalPrice;</code> 5) Немає правильної відповіді.</p>
11.	<p>Описати змінну символьного типу . 1) <code>int symbol;</code> 2) <code>double symbol;</code> 3) <code>bool symbol;</code> 4) <code>real symbol;</code> 5) Немає правильної відповіді.</p>
12.	<p>Описати змінну логічного типу . 1) <code>int isEmpty;</code> 2) <code>double isEmpty;</code> 3) <code>bool isEmpty;</code> 4) <code>logic isEmpty;</code> 5) Немає правильної відповіді.</p>
13.	<p>Описати змінну логічного типу . 1) <code>int isEmpty;</code> 2) <code>double isEmpty;</code> 3) <code>boolean isEmpty;</code> 4) <code>logic isEmpty;</code> 5) Немає правильної відповіді.</p>
14.	<p>Описати змінну символьного типу . 1) <code>int symbol;</code> 2) <code>char symbol;</code> 3) <code>bool symbol;</code> 4) <code>real symbol;</code> 5) Немає правильної відповіді.</p>
15.	<p>Описати змінну рядкового типу . 1) <code>string string;</code> 2) <code>char string;</code> 3) <code>bool string;</code> 4) <code>real string;</code> 5) Немає правильної відповіді.</p>
16.	<p>Описати змінну рядкового типу . 1) <code>string name;</code> 2) <code>char name;</code> 3) <code>bool name;</code> 4) <code>name string;</code> 5) Немає правильної відповіді.</p>

17.	<p>Вкажіть необхідні пункти для опису змінної рядкового типу .</p> <ol style="list-style-type: none"> 1) const; 2) char; 3) string; 4) int; 5) double; 6) float; 7) bool; 8) <u>ідентифікатор</u>; 9) вказаних пунктів недостатньо.
18.	<p>Вкажіть необхідні пункти для опису константи рядкового типу .</p> <ol style="list-style-type: none"> 1) const; 2) char; 3) string; 4) int; 5) double; 6) float; 7) bool; 8) <u>ідентифікатор</u>; 9) <u>літерал рядкового типу</u>; 10) <u>оператор присвоєння</u>; 11) Вказаних пунктів недостатньо.
19.	<p>Вкажіть необхідні пункти для опису константи цілого типу .</p> <ol style="list-style-type: none"> 1) const; 2) char; 3) string; 4) int; 5) double; 6) float; 7) bool; 8) <u>ідентифікатор</u>; 9) <u>літерал цілого типу</u>; 10) <u>оператор присвоєння</u>; 11) Вказаних пунктів недостатньо.
20.	<p>Вкажіть необхідні пункти для опису константи логічного типу .</p> <ol style="list-style-type: none"> 1) const; 2) char; 3) string; 4) int; 5) double; 6) float; 7) bool; 8) <u>ідентифікатор</u>; 9) <u>літерал логічного типу</u>; 10) <u>оператор присвоєння</u>; 11) Вказаних пунктів недостатньо.
21.	<p>У яких із фрагментів описано змінні цілого типу?</p>

	<p>1) <code>int a=b=c=9;</code> 2) <code>char a=b=c=9;</code> 3) <code>string a=b=c=9;</code> 4) <code>bool a=b=c=9;</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
22.	<p>У яких із фрагментів описано змінні цілого типу? 1) <code>int a=b=c=9;</code> 2) <code>int a,b,c=9;</code> 3) <code>int a,b,c;</code> 4) <code>bool a=b=c=9;</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
23.	<p>У яких із фрагментів описано змінні дійсного типу? 1) <code>double a=b=c=9;</code> 2) <code>double a=b=c=9;</code> 3) <code>float a=b=c=9;</code> 4) <code>real a=b=c=9;</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
24.	<p>У яких із фрагментів описано змінні дійсного типу? 1) <code>double a=b=c=9;</code> 2) <code>double a,b,c=9;</code> 3) <code>double a,b,c;</code> 4) <code>real a=b=c=9;</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
25.	<p>У яких із фрагментів описано змінні дійсного типу? 1) <code>double a=b=c=9;</code> 2) <code>double a,b,c=9;</code> 3) <code>float a,b,c;</code> 4) <code>real a=b=c=9;</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
26.	<p>У яких із фрагментів описано змінні символного типу? 1) <code>char a=b=c= 'A' ;</code> 2) <code>char a,b,c= 'A' ;</code> 3) <code>symbol a,b,c;</code> 4) <code>symbol a=b=c=9;</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
27.	<p>У яких із фрагментів описано змінні символного типу? 1) <code>char a=b=c= 'char' ;</code> 2) <code>char a= 'c' ,b= 'h' , c= 'a' ,d= 'r' ;</code> 3) <code>char a= 'c' ,b= 'a' , c= 'h' ,d= 'r' ;</code> 4) <code>char a= 'r' ,b= 'a' , c= 'h' ,d= 'c' ;</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
28.	<p>У яких із фрагментів описано змінні символного типу? 1) <code>char a=b=c= "char" ;</code> 2) <code>char a= "c" ,b= "h" , c= "a" ,d= "r" ;</code> 3) <code>char a= "c" ,b= "a" , c= "h" ,d= "r" ;</code></p>

	<p>4) <code>char a="r",b="a",c="h",d="c";</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
29.	<p>У яких із фрагментів описано змінні рядкового типу? 1) <code>string a=b=c="string";</code> 2) <code>char a="c",b="h",c="a",d="r";</code> 3) <code>string a='r',b='a',c='h',d='c';</code> 4) <code>char a="r",b="a",c="h",d="c";</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
30.	<p>У яких із фрагментів описано змінні логічного типу? 1) <code>bool a=b=c="true";</code> 2) <code>bool a=true,b=false;</code> 3) <code>bool a=false,b=true;</code> 4) <code>boolean a=true,b=true;</code> 5) Жоден із вказаних фрагментів не є правильним.</p>
31.	<p>Значення змінної цілого типу <code>a</code> вводитьься з клавіатури. Які із вказаних типів даних можна використати для опису змінної, у якій необхідно зберегти значення останнього біта змінної <code>a</code>? 1) <code>int</code>; 2) <code>string</code>; 3) <code>bit</code>; 4) Немає правильної відповіді</p>
32.	<p>Значення змінної цілого типу <code>a</code> вводитьься з клавіатури. Які із вказаних типів даних можна використати для опису змінної, у якій необхідно зберегти значення першого біта змінної <code>a</code>? 1) <code>int</code>; 2) <code>byte</code>; 3) <code>bit</code>; 4) Немає правильної відповіді</p>

Оператор присвоєння. Явне і неявне перетворення типів даних

1.	<p>Загальну форму чого приведено нижче? <code>змінна = вираз ;</code> 1) Оператора присвоєння; 2) Арифметичного виразу; 3) Логічного виразу; 4) Немає правильної відповіді .</p>
2.	<p>Загальну форму чого приведено нижче? <code>вираз = змінна ;</code> 1) Оператора присвоєння; 2) Арифметичного виразу; 3) Логічного виразу; 4) Немає правильної відповіді .</p>
3.	<p>Чим є вказана загальна форма? <code>змінна == вираз</code> 1) Оператором присвоєння; 2) Арифметичним виразом; 3) Логічним виразом; 4) Немає правильної відповіді .</p>
4.	<p>Чи обов'язково змінна і вираз у вказаній загальній формі повинні бути одного типу? <code>змінна = вираз ;</code> 1) Так; 2) Ні; 3) Залежить від назви змінної; 4) Немає правильної відповіді .</p>
5.	<p>Яким буде значення змінної <code>a</code>? <code>int c=7;</code></p>

	<pre>int d=5; int a=d=c; 1) 5; 2) 7; 3) 12; 4) 'd'.</pre>
6.	<p>Яким буде значення змінної a?</p> <pre>int c=7; int d=5; int a=c=d; 1) 5; 2) 7; 3) 12; 4) 'd'.</pre>
7.	<p>Яким буде значення змінної a?</p> <pre>int c, d=5; int a=d=c; 1) 5; 2) 7; 3) 12; 4) Немає правильної відповіді.</pre>
8.	<p>Яким буде значення змінної a?</p> <pre>char a="A"; 1) 65; 2) 64; 3) A; 4) Немає правильної відповіді.</pre>
9.	<p>При виконанні якого із перетворень типів даних може відбуватися втрата інформації?</p> <p>1) Явному перетворенні типів даних; 2) Неявному перетворенні типів даних; 3) При виконанні явного і неявного перетворення типів даних; 4)Немає правильної відповіді.</p>
10.	<p>При виконанні якого із перетворень типів даних не відбувається втрата інформації?</p> <p>1) Явному перетворенні типів даних; 2) Неявному перетворенні типів даних; 3) При виконанні явного і неявного перетворення типів даних; 4)Немає правильної відповіді.</p>
11.	<p>Що дозволяє виконати вказана загальна форма?</p> <p>(<u>ЦІЛЬОВИЙ ТИП</u>) <u>ВИРАЗ</u></p> <p>1) Описати змінну; 2) Видалити змінну; 3) Виконати неявне приведення типу; 4) Немає правильної відповіді.</p>
12.	<p>Яким буде значення змінної d?</p> <pre>int c=5; double d=c/2; 1) 5; 2) 2; 3) 2.5; 4) Немає правильної відповіді.</pre>
13.	<p>Яким буде значення змінної d?</p> <pre>int c=4; double d=Math.Pow(c, 1/2); 1) 2; 2) 4; 3) 1; 4) Немає правильної відповіді.</pre>
14.	<p>Яким буде значення змінної d?</p> <pre>int c=4; double d=Math.Pow(c, 1.0/2); 1) 2; 2) 4; 3) 1; 4) Немає правильної відповіді.</pre>
15.	<p>Яким буде значення змінної d?</p> <pre>int c=2;</pre>

	<pre>double d=Math.Pow(c,1/c);</pre> <p>1) 2; 2) 4; 3) 1; 4) Немає правильної відповіді.</p>
16.	<p>Яким буде значення змінної d?</p> <pre>int c=2;</pre> <pre>double d=Math.Pow(4,1/(double)c);</pre> <p>1) 2.0; 2) 4.0; 3) 1.0; 4) Немає правильної відповіді.</p>
17.	<p>Яким буде значення змінної d?</p> <pre>int c=2;</pre> <pre>double r=4;</pre> <pre>double d=Math.Pow(c,1/c*r);</pre> <p>1) 2; 2) 4; 3) 1; 4) Немає правильної відповіді.</p>
18.	<p>Що містить вказаний фрагмент програми?</p> <pre>double d=3.5;</pre> <pre>int c=d;</pre> <p>1) Неявне приведення типів даних; 2) Явне приведення типів даних; 3) Явне і неявне приведення типів даних; 4) Немає правильної відповіді.</p>
19.	<p>Що містить вказаний фрагмент програми?</p> <pre>int c=10;</pre> <pre>double d=c;</pre> <p>1) Неявне приведення типів даних; 2) Явне приведення типів даних; 3) Явне і неявне приведення типів даних; 4) Немає правильної відповіді.</p>
20.	<p>Що містить вказаний фрагмент програми?</p> <pre>int c=10;</pre> <pre>double d=(double)c;</pre> <p>1) Неявне приведення типів даних; 2) Явне приведення типів даних; 3) Явне і неявне приведення типів даних; 4) Немає правильної відповіді.</p>
21.	<p>Що містить вказаний фрагмент програми?</p> <pre>byte b=7;</pre> <pre>int c=b;</pre> <pre>double d=b;</pre> <p>1) Неявне приведення типів даних; 2) Явне приведення типів даних; 3) Явне і неявне приведення типів даних; 4) Немає правильної відповіді.</p>
22.	<p>Що містить вказаний фрагмент програми?</p> <pre>int c=8;</pre> <pre>byte b=c;</pre> <p>1) Неявне приведення типів даних; 2) Явне приведення типів даних; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
23.	<p>Що містить вказаний фрагмент програми?</p> <pre>byte b=8;</pre> <pre>int c=b;</pre> <p>1) Неявне приведення типів даних; 2) Явне приведення типів даних; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
24.	<p>Що містить вказаний фрагмент програми?</p> <pre>float b=8;</pre>

	<pre>double c=b;</pre> <p>1) Неявне приведення типів даних; 2) Явне приведення типів даних; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
25.	<p>Що містить вказаний фрагмент програми?</p> <pre>double b=8; float c=b;</pre> <p>1) Неявне приведення типів даних; 2) Явне приведення типів даних; 3) Вказаний фрагмент містить помилку; 4) Немає правильної відповіді.</p>
26.	<p>Яким буде значення змінної c?</p> <pre>double d=3.5; int c=d;</pre> <p>1) 5; 2) 3; 3) 3.5; 4) Вказаний фрагмент містить помилку; 5) Немає правильної відповіді.</p>
27.	<p>Яким буде значення змінної c?</p> <pre>double d=3.5; int c=(int)d;</pre> <p>1) 5; 2) 3; 3) 3.5; 4) Вказаний фрагмент містить помилку; 5) Немає правильної відповіді.</p>
28.	<p>Яким буде значення змінної c?</p> <pre>double d=3.5; d=5; int c=d;</pre> <p>1) 5; 2) 3; 3) 3.5; 4) Вказаний фрагмент містить помилку; 5) Немає правильної відповіді.</p>
29.	<p>У якому з варіантів наведено приклад явного приведення типів?</p> <p>1) a=(int)d; 2) double a= c; 3) a=int; 4) У жодному.</p>
30.	<p>У якому з варіантів наведено приклад неявного приведення типів?</p> <p>1) c=(int)d; 2) int= c; 3) d=int; 4) У жодному.</p>

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. C# Tutorial. 2023. URL: <https://www.javatpoint.com/c-sharp-tutorial>
2. Learn C# Programming. 2023. URL: <https://www.tutorialsteacher.com/csharp>
3. C#.NET Tutorials For Beginners and Professionals. 2023. URL: <https://dotnettutorials.net/course/csharp-dot-net-tutorials/>

ЗМІСТ

ВСТУП. ТИПИ ДАНИХ. ЗМІННІ, КОНСТАНТИ	3
1. ТЕХНОЛОГІЯ .NET FRAMEWORK	3
2. МОВА ПРОГРАМУВАННЯ. ПОНЯТТЯ ПРОГРАМИ	5
3. ВЕЛИЧИНИ. ТИП ВЕЛИЧИН	9
Поняття величини та типу	9
Стандартні типи у C#.....	10
Опис змінних та констант.....	11
Перетворення типів	12
Переліковий тип	12
НАБІР ТЕСТІВ ДЛЯ САМОКОНТРОЛЮ	15
Типи даних.....	15
Змінні, константи	18
Приклади опису	22
Оператор присвоєння. Явне і неявне перетворення типів даних	27
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	30

Укладачі: к. ф.-м. н., доц. Брила А.Ю.,
ст.викл., Ломага М.М.
к. ф.-м. н. Антосяк П.П.

Рецензенти: к.ф.-м.н., доц. Погоріляк О.О.,
к.т.н., доц. Андрашко Ю.В.

ТИПИ ДАНИХ. ЗМІННІ, КОНСТАНТИ У C#

Методичні матеріали з організації самостійної роботи студентів з дисципліни
«Програмування»