

## *Лабораторна робота* *з дисципліни «Алгоритмізація та програмування» № 2*

### **Тема: Розробка лінійних програм**

**Мета роботи:** вивчити правила оголошення змінних базових типів даних; навчитись записувати та інтерпретувати вирази на основі форматів машинного зображення даних, визначеної множини операцій мови C++ та їх пріоритетів.

### **Теоретичні відомості**

#### *Типи даних*

В C++ визначено наступні основні (базові) типи даних:

- `char` – символний
- `wchar_t` – символний двох-байтовий
- `int` – цілочисельний
- `float` – з плаваючою крапкою
- `double` – з плаваючою крапкою подвійної точності
- `bool` – логічний (або булевий)
- `void` – без значення

В C++ перед типами `char`, `int` і `double` можна записувати модифікатори. Модифікатор використовується для уточнення базового типу та зміни його значення. Можливі модифікатори `signed`, `unsigned`, `long` та `short` можна застосовувати до цілочисельних типів. Крім того, модифікатори `signed` та `unsigned` можна використовувати з типом `char`, а модифікатор `long` – з типом `double`. Модифікатори в парах `signed / unsigned` та `long / short` – взаємовиключні (з кожної такої пари можна використовувати лише один).

За замовчуванням призначається знаковий тип, тому специфікатор `signed` вказувати необов'язково. Для знакових типів старший біт відповідає за знак, при цьому 0 – відповідає додатному числу, а 1 – від'ємному.

Змінні типу `float` займають у пам'яті 4 байти: 1 старший біт – знаковий, наступні 8 бітів – порядок, а інші 23 біти – мантиса.

Тип `char` використовується для подання символу, а для визначення його десяткового коду – тип `unsigned char` чи `signed char`. Значенням змінної типу `char` є відповідний код символу (розміром 1 байт). Для визначення десяткових кодів символів кирилиці специфікатор типу ідентифікатора даних має бути типом `unsigned char`. Крім того, тип `char` використовується для оголошення рядків типу `char*`.

Змінна логічного типу `bool` займає лише 1 байт і може отримувати лише одне з двох значень: `true` (істина) чи `false` (хибність). Переважно змінні типу `bool` використовуються для індикації стану чи для побудови логічних виразів в умовах. Слід зауважити, що в мові C змінним логічного типу можна присвоювати цілочисельні значення; при цьому значення 0 сприймається як `false`, а будь-яке ненульове значення – як `true`.

Тип `void` (порожній, без значення) використовується для визначення типу функцій, які не повертають жодного результату, а всі дії та обчислення виконуються всередині цих функцій. Цей тип використовують для подання порожнього списку

аргументів функції. Тип *void* використовують як базовий тип для вказівників. Крім того, його використовують в операціях перетворення типів.

Тип	Назва	Розмір, байт	Діапазон значень
<u>char</u>	знаковий символний	1	−128...127
<u>unsigned char</u>	беззнаковий символний	1	0...255
<u>short</u>	знаковий короткий цілий	2	−32 768...32767
<u>unsigned short</u>	беззнаковий короткий цілий	2	0...65 535
<u>int</u>	знаковий цілий	2	−32 768...32767
		4	−2 147 483 648... 2 147 483 647
<u>unsigned int</u>	беззнаковий цілий	2	0...65 535
		4	0... 4 294 967 295
<u>long</u>	знаковий довгий цілий	4	−2 147 483 648... 2 147 483 647
<u>unsigned long</u>	беззнаковий довгий цілий	4	0... 4 294 967 295
<u>float</u>	дійсний одинарної точності	4	$1.1 \cdot 10^{-38} \dots$ $3.4 \cdot 10^{38}$
<u>double</u>	дійсний подвійної точності	8	$2.2 \cdot 10^{-308} \dots$ $1.7 \cdot 10^{308}$
<u>long double</u>	довгий дійсний	10	$3.3 \cdot 10^{-4932} \dots$ $1.1 \cdot 10^{4932}$
<u>bool</u>	логічний	1	<u>false</u> , <u>true</u>
<u>enum</u>	перелік	2 чи 4	
<u>void</u>	порожній, без значення		

**Структуровані** (похідні) типи утворюються з основних, до них належать масиви будь-яких типів, вказівники, функції, файли, структури, об'єднання, переліки тощо.

### Оголошення змінних

Ім'я (ідентифікатор) змінної може складатися з літер латиниці (a...z, A...Z), цифр (0...9) та символа підкреслення ("\_"), але обов'язково має починатися з літери чи символа підкреслення. Мова C++ є чутливою до регістру, тобто відповідні великі та малі літери вважаються різними символами.

При оголошенні змінних вказується їхній тип, назва та за потреби початкове значення. Початковими значеннями можуть бути як певні значення, так і вирази. Оголошення змінних має такий формат:

```
<тип> <ім'я_змінної_1> [= <початкове_значення_змінної_1>]
    [, <ім'я_змінної_2> [= <початкове_значення_змінної_2>]
    ...
    , <ім'я_змінної_N> [= <початкове_значення_змінної_N>]];
```

Числові змінні потрібно ініціалізовувати вручну, інакше їхні значення будуть невизначеними (випадковими). Наприклад

```
int b;           // Інтерпретується як signed int c
short c;         // Інтерпретується як signed short int a
```

```

unsigned d;    // Інтерпретується як unsigned int d
signed e;      // Інтерпретується як signed int f
int f, g=4;    // f – неініціалізована, g=4
int h(7), i=2*g, j=h+g; // h=7, i=8, j=11
int q = 057;   // q – ініціалізована вісімковим значенням 057

```

Окремий символ записується в одинарних лапках (·), а рядок символів – у подвійних лапках (" "). Наприклад

```

char cx='a';    // cx ініціалізована символом 'a'
char x, c='\n'; // x неініціалізована, c ініціалізована символом <Enter>
char st[5]="Lviv"; // масив із 5 символів ініціалізований значенням "Lviv"
char s[]="student"; // рядок s ініціалізований значенням "student"

```

### Оголошення констант

Константа – це стала величина, значення якої не змінюється під час виконання програми. При оголошенні константам обов'язково потрібно задавати певні значення. Значення констант не можна змінювати. Існує два способи визначення констант:

- за допомогою макроса **#define**;
- за допомогою ключового слова **const**.

Директива **#define** має такий формат: **#define <ідентифікатор> <значення>**

Наприклад

```

#define A 5
#define MESSAGE "Go home!"
#define square(x) ((x) * (x))

```

Для оголошення констант також використовується специфікатор **const**, після якого вказується її тип. Формат оголошення констант такий:

```

const <тип> <ім'я_константи_1>[= <значення_константи_1>] [, <ім'я_константи_2>[=
<значення_константи_2>]
...
, <ім'я_константи_N>[= <значення_константи_N>]];

```

У мові C++ розрізняють чотири типи констант: цілі, дійсні, символьні константи та рядки. Для цілих констант тип можна не вказувати, оскільки за замовчуванням константам призначається тип **int**. Наприклад

```

const int A = 3;
const B = 286; // інтерпретується як const int B=286
const C = 2*A;

```

При оголошенні дійсних констант слід вказувати відповідний тип (**float**, **double** чи **long double**) і записувати їхні значення з десятковою крапкою.

Крім звичайної форми, дійсні константи можна записувати в експоненціальній формі з плаваючою крапкою, в якій десяткове дійсне число  $N$  записують у вигляді мантиси  $m$  і порядку  $p$ :  $N = m \cdot 10^p$ . Число у формі з плаваючою крапкою може бути подане безліччю способів. Для однозначного подання на мантису накладається обмеження  $0 \leq |m| < 1$ , при дотриманні якого число знаходиться у нормалізованій формі. Наприклад

```

const float Pi = 3.14159, fi = -1.84;
const double a = 0.45e+2; // еквівалентно a=0.45·102=45

```

```
const double b = 0.2e-3; // еквівалентно b=0.2·10-3=0.0002
```

Символьна константа містить лише один символ в одинарних лапках ( ' ') і оголошується так:

```
const char d='+', a='ц', c='9';
```

Значенням символьної константи є числовий код символу. Рядкова константа (літерал) – це послідовність символів, включаючи великі та малі літери кирилиці, латиниці, а також цифри, пробіли, розташовані у подвійних лапках ( " "), наприклад:

```
const char S[]="Київ – столиця України!"; // рядок типу char
```

```
const char *SS="Допоможи. Не будь байдужим!"; // рядок типу char*
```

Якщо ключове слово `const` стоїть перед оголошенням структурованих типів (масив, структура, переліки), це призводить до того, що кожен елемент відповідної структури також є константою.

### Арифметичні операції

До арифметичних операцій належать додавання (+), віднімання (–), множення (\*), ділення (/), остача від ділення цілих чисел (%), інкремент (++) та декремент (--).

Арифметичний вираз складається з операндів, арифметичних операцій і оператора присвоювання. Вираз, який завершується крапкою з комою, називається оператором. Оператор задає закінчений опис певної дії. Виконання оператора виразу полягає в обчисленні виразу. Окремим випадком виразу є порожній оператор (;), який використовується, коли за синтаксисом оператор потрібен, а за змістом – ні.

У мові C++, крім простого присвоювання, ще є складені операції присвоювання

Оператор	Призначення	Приклад	Еквівалент
<code>+=</code>	присвоювання з додаванням	<code>x += y</code>	<code>x = x + y</code>
<code>-=</code>	присвоювання з відніманням	<code>x -= y</code>	<code>x = x - y</code>
<code>*=</code>	присвоювання з множенням	<code>x *= y</code>	<code>x = x * y</code>
<code>/=</code>	присвоювання з діленням	<code>x /= y</code>	<code>x = x / y</code>
<code>%=</code>	присвоювання остачі від ділення цілих чисел	<code>x %= y</code>	<code>x = x % y</code>
<code>&lt;&lt;=</code>	присвоювання зі зсувом ліворуч	<code>x &lt;&lt;= y</code>	<code>x = x &lt;&lt; y</code>
<code>&gt;&gt;=</code>	присвоювання зі зсувом праворуч	<code>x &gt;&gt;= y</code>	<code>x = x &gt;&gt; y</code>
<code>&amp;=</code>	присвоювання з порозрядною операцією (І)	<code>x &amp;= y</code>	<code>x = x &amp; y</code>
<code>^=</code>	присвоювання з порозрядною операцією виключне (АБО-НЕ)	<code>x ^= y</code>	<code>x = x ^ y</code>
<code> =</code>	присвоювання з порозрядною операцією (АБО)	<code>x  = y</code>	<code>x = x   y</code>

Операція *інкремент* (++) додає одиницю до операнда, а операція *декремент* (--) віднімає одиницю від операнда. Залежно від місця розміщення операції відносно операнда розрізняють дві форми цих операцій: префіксну та постфіксну. У префіксній формі операцію записують перед операндом, наприклад: ++i, --j, а в постфіксній – після операнда, наприклад: i++, j--.

Значення змінної у префіксній формі спочатку збільшується (зменшується) на одиницю і лише після цього ця змінна з її новим значенням бере участь в

арифметичному виразі. А в постфіксній формі у виразі спочатку використовується поточне значення змінної і лише після цього її значення збільшується чи зменшується на одиницю.

Для запису виразів використовують відповідні операції. У мові С існують *унарні*, *бінарні* та *тернарні* операції, залежно від кількості операндів. Більшість операцій є бінарними, тобто мають два операнди, один з яких розміщується перед знаком операції, а другий – після неї. Унарні операції мають лише один операнд, який записується після символу операції.

Обчислення у виразах виконуються згідно пріоритету операцій (табл. 11).

№ з/п	Знак операції	Назва операції	Порядок виконання
1	:: ( ) [ ] -> .	Оператор розширення області дії Вираз, виклик функцій Виділення елемента масива Виділення елемента структури або об'єднання Виділення елемента структури за допомогою вказівника	зліва направо
2	–	Унарний мінус (зміна знаку)	справа наліво
№ з/п	Знак операції	Назва операції	Порядок виконання
	~ ! *  & ++ -- (<тип> sizeof	Порозрядна інверсія (побітове НЕ) Логічне заперечення (НЕ) Звертання за визначеною адресою (розіменування вказівника) Отримання адреси (адресація) Збільшення на одиницю (інкремент) Зменшення на одиницю (декремент) Явне перетворення типу Визначення розміру в байтах	
3	* / %	Множення Ділення Остача від ділення	зліва направо
4	+ –	Додавання Віднімання	зліва направо
5	<< >>	Арифметичний зсув вліво Арифметичний зсув вправо	зліва направо
6	< > <= >=	Менше Більше Менше або дорівнює Більше або дорівнює	зліва направо
7	== !=	Дорівнює Не дорівнює	зліва направо
8	&	Порозрядне логічне множення (І)	зліва направо
9	^	Порозрядна сума за модулем два (XOR)	зліва направо
10		Порозрядне логічне додавання (OR)	зліва направо

11	&&	Логічне множення (AND)	зліва направо
12		Логічне додавання (OR)	зліва направо
13	?:	Умовна тернарна операція	справа наліво
14	= *= /= %= += - = &=  = ^=  >>= <<=	Присвоєння Присвоєння з множенням Присвоєння з діленням Присвоєння остачі від ділення цілих чисел Присвоєння з додаванням Присвоєння з відніманням Присвоєння з порозрядною операцією І (AND) Присвоєння з порозрядною операцією АБО (OR) Присвоєння з порозрядною операцією сума за модулем два (XOR) Присвоєння зі зсувом вправо Присвоєння зі зсувом вліво	справа наліво
15	,	Кома	зліва направо

**Логічна** операція – це дія, яка виконується над логічними змінними, результатом якої є 1 (true) або 0 (false). Логічні операції обчислюють кожний операнд з огляду на його еквівалентність нулю.

З логічних змінних за допомогою логічних операцій та дужок будуються логічні вирази. Результати застосування цих операцій до булевих значень подано в табл. 12. Операнди логічних операцій можуть бути цілого типу, дійсного типу чи типу вказівник, при цьому в кожній операції можуть брати участь операнди різних типів. Операнди логічних виразів обчислюються зліва направо. Якщо значення першого операнда вистачає, щоб визначити результат операції, то другий операнд не обчислюється.

A	B	!A	A    B	A&&B
0	0	1	0	0
0	1	1	1	0
1	0	0	1	0
1	1	0	1	1

Логічні побітові операції виконуються над цілими числами та опрацьовують число побітово. До цих операцій належать: побітове логічне заперечення “НЕ” (~), побітове логічне множення “І” (&), побітове логічне додавання “АБО” (|), побітове логічне виключне АБО “АБО-НЕ” (^) (табл. 13). Операнди логічних операцій мають бути будь-якого цілого типу.

A	B	~A	A   B	A&B	A^B
0	0	1	0	0	0
0	1	1	1	0	1
1	0	0	1	0	1
1	1	0	1	1	0

### Математичні функції

Для того, щоб їх використовувати математичні функції, які широко використовуються для запису різних математичних залежностей та виразів, потрібно підключити бібліотеку, яка містить ці функції:

```
#include <math.h>
```

Для використання математичних констант, крім бібліотеки math.h, потрібно зробити такі оголошення:

```
#define _USE_MATH_DEFINES
#include <cmath>
```

При цьому константу `_USE_MATH_DEFINES` обов'язково потрібно оголошувати перед підключенням бібліотеки математичних функцій.

### Перетворення типів

Перетворення типів виконуються, якщо операнди, які входять до виразу, мають різні типи.

Загальні правила перетворення типів:

- 1) операнди типу float зводяться до типу double;
- 2) якщо один операнд long double, то другий зводиться до цього ж типу;
- 3) якщо один операнд double, а другий – float чи ціле число, то другий операнд перетворюється до типу double;
- 4) операнди цілих типів char та short зводяться до типу int;
- 5) цілі операнди типу unsigned char та unsigned short зводяться до типу unsigned int;
- 6) якщо один операнд типу unsigned long, то другий цілий операнд перетворюється до типу unsigned long;
- 7) якщо один операнд типу long, то другий зводиться до типу long;
- 8) якщо один операнд типу unsigned int, то другий цілий операнд зводиться до цього ж типу.

Таким чином при обчисленні виразів операнди зводяться до типу того операнда, який має більший розмір.

Розглянемо кілька правил перетворення типів. Результат операції ділення буде цілим числом, якщо ділене і дільник є цілими, і дійсним числом, якщо один з операндів є дійсного типу. Наприклад

```
int m = 2, n = 3;
float a = m/n;    // a=0
double k=2.0;    // або k=2.;
double rez=k/3;  // rez=0.666(6)
```

При присвоюванні остаточний результат зводиться до типу змінної, яка стоїть ліворуч від знаку присвоювання (“=”). Якщо тип лівого операнда є менш точним, ніж тип результату виразу, що стоїть праворуч оператора присвоювання, то можлива втрата точності чи взагалі неправильний результат. Наприклад

```
float a = 2.8, b = 1.7;
int c = a * b;           // c=4, а не 4.76, бо c – ціле число
```

Перетворення типів може бути неявним, при виконанні операцій та виклику функцій, чи явним, при виконанні операцій перетворення типів.



Операцію явного перетворення типів використовують, щоб уникнути помилок, подібних до вищенаведених прикладів. Ця операція має такий вигляд:

(<тип>) <вираз>

Перед ім'ям змінної у дужках задається тип, до якого її слід перетворити. Наприклад

```
int m = 2, n = 3;
```

```
double a = (double) m/n;
```

```
// a=0.66(6)
```

Явне перетворення типу є джерелом можливих помилок, оскільки вся відповідальність за його результат покладається на програміста. Операцію явного перетворення типів слід використовувати обережно, лише якщо іншого способу отримати певний результат немає.

### Завдання 1

- 1) проаналізувати скільки змінних і якого типу необхідно використати, щоб розв'язати задачу;
- 2) вхідні дані ввести з клавіатури, супроводжуючи відповідними текстовими повідомленнями;
- 3) провести розрахунки і вивести результат; виведення даних також необхідно супроводжувати відповідними текстовими повідомленнями;
- 4) вивести отримані результати. Виведення даних необхідно супроводжувати відповідними текстовими повідомленнями.

#### Варіант 1

В М пакетів вміщається N їжачків. Скільки пакетів потрібно, щоб принести на пару K їжачків?

#### Варіант 2

Хор, до складу якого входить N хлопчиків та M дівчаток, вирішив заспівати пісню. При цьому L% хлопчиків лише відкривають ротові порожнини, H% дівчат кричать на всю міць. Скільки відсотків усього 24 хору співає нормально?

#### Варіант 3

Задана довжина ребра куба. Знайти площу бокової поверхні куба та його об'єм.

#### Варіант 4

Пішохід рухався протягом t1 хвилин з деякою швидкістю v1 і t2 хвилин з іншою швидкістю v2, яка у півтори рази більша за v1. Знайти швидкості руху пішохода v1 і v2, якщо пройдений ним шлях склав S метрів.

#### Варіант 5

З тераріуму втекли X гадюк, Y кобр та Z гюрз. Довжина кожної гадюки 1 м, кожної кобри 1.234 м, а гюрзи 1 м 15 см. Скільки метрів ядовитих змій втекло з тераріуму? Яку довжину вони складають у сантиметрах?

#### Варіант 6

Задане ціле двохзначне число x. За допомогою операцій виділення цілої і/або дробової частини визначити суму квадратів цифр, що його складають.

#### Варіант 7



Трикутник ABC заданий довжинами своїх сторін. Знайти довжину висоти, опущеної з вершини A.

**Варіант 8**

Визначити час падіння металевої кулі з  $k$ -го поверху будівлі, якщо висота одного поверху складає  $z$  метрів.

**Варіант 9**

Довжина сторони дзеркала квадратної форми становить  $N$  м<sup>2</sup>. Скільки людей зможе одночасно подивитися на себе в дзеркало, якщо площа обличчя кожної людини дорівнює  $M$  м<sup>2</sup> (відображення обличчя у дзеркалі не повинні перекриватись)?

**Варіант 10**

У батьковому годиннику є  $A$  колес та  $B$  інших деталей. Після того, як Володя розібрав, а потім зібрав батьків годинник  $C\%$  колес та  $D\%$  інших деталей не вмістилося? Скільки всього деталей вмістилось у годинник?

**Варіант 11**

Автомобіль і зовнішнім діаметром колеса  $d$  см проїхав  $S$  км. Скільки обертів зробило кожне колесо автомобіля?

**Варіант 12**

Селекціонер вивів новий сорт зернової культури і зняв з дослідної ділянки  $k$  кг насіння. Посіявши  $1$  кг насіння, можна за сезон зібрати  $p$  кг насіння. Через скільки років селекціонер зможе засіяти новою культурою поле площею  $S$  гектарів, якщо норма висіву  $n$  кг/гектар?

**Варіант 13**

Фермер на початку кожної зими підвищує відпускну ціну на молоко на  $p$  процентів, а кожного літа – знижує на стільки ж процентів. Як зміниться ціна на молоко через  $n$  років?

**Варіант 14**

Задані моменти початку і кінця деякого проміжку часу в годинах, хвилинах і секундах (в межах доби). Знайти тривалість цього проміжку в кожній з перелічених одиниць.

**Варіант 15**

Трикутник заданий координатами своїх вершин на площині. Знайти площу трикутника.

**Варіант 16**

Роман задумав число і нікому його не назвав. Друзі спіймали його і примусили подвоїти задумане число, а потім додати до нього число  $5$ . І тільки після того, як вони йому пообіцяли допомогти на модулі з програмування, Роман зізнався, що вийшло число  $N$ . Визначте, яке число загадав Роман?

**Варіант 17**

Студент Андрій сів виконувати домашнє завдання і просидів за столом  $2$  години. З них  $X$  хвилин він дивився у вікно,  $Y$  хвилин шукав гумку, щоб стерти малюнок, на малювання якого він витратив перед тим  $Z$  хвилин. Решту часу Андрій

програмував. Скільки програм він встиг написати, якщо на написання однієї програми він витрачав 5 хвилин?

### **Варіант 18**

Комерсант, який мав стартовий капітал  $k$  гривень, зайнявся торгівлею, яка щомісяця збільшує капітал на  $p$  %. Через скільки років він накопить суму  $s$ , достатню для купівлі власного магазину?

### **Варіант 19**

З кола радіусом  $R$  вирізаний прямокутник, велика сторона якого дорівнює  $a$ . Знайти максимальний радіус кола, яке можна вирізати з отриманого прямокутника.

### **Варіант 20**

Хор, до складу якого входить  $N$  хлопчиків та  $M$  дівчаток, вирішив заспівати пісню. При цьому  $L\%$  хлопчиків лише відкривають ротові порожнини,  $H\%$  дівчат кричать на всю міць. Скільки відсотків усього 24 хору співає нормально

## **Завдання 2**

1. Написати програму для обчислення виразів згідно свого варіанта (результати обчислень за двома формулами мають збігатися). Значення змінних потрібно ввести з клавіатури а результат розрахунків вивести з відповідним текстовим повідомленням.

### **Варіант 1**

$$z_1 = \cos \alpha + \sin \alpha + \cos 3\alpha + \sin 3\alpha;$$

$$z_2 = 2\sqrt{2} \cos \alpha \times \sin \left( \frac{\pi}{4} + 2\alpha \right).$$

### **Варіант 2**

$$z_1 = \frac{\sin 2\alpha + \sin 5\alpha - \sin 3\alpha}{\cos \alpha + 1 - 2 \sin^2 2\alpha};$$

$$z_2 = 2 \sin \alpha.$$

### **Варіант 3**

$$z_1 = 1 - \frac{1}{4} \sin^2 2\alpha + \cos 2\alpha;$$

$$z_2 = \cos^2 \alpha + \cos^4 \alpha.$$

### **Варіант 4**

$$z_1 = \cos \alpha + \cos 2\alpha + \cos 6\alpha + \cos 7\alpha;$$

$$z_2 = 4 \cos \frac{\alpha}{2} \times \cos \frac{5}{2} \alpha \times \cos 4\alpha$$

### **Варіант 5**

$$z_1 = \cos^2\left(\frac{3}{8}\pi - \frac{\alpha}{4}\right) - \cos^2\left(\frac{11}{8}\pi + \frac{\alpha}{4}\right);$$

$$z_2 = \frac{\sqrt{2}}{2} \cdot \sin \frac{\alpha}{2}.$$

**Варіант 6**

$$z_1 = (\cos \alpha - \cos \beta)^2 - (\sin \alpha - \sin \beta)^2;$$

$$z_2 = -4 \sin^2 \frac{\alpha - \beta}{2} \times \cos(\alpha + \beta).$$

**Варіант 7**

$$z_1 = \frac{\sin\left(\frac{\pi}{2} + 3\alpha\right)}{1 - \sin(3\alpha - \pi)};$$

$$z_2 = \operatorname{ctg}\left(\frac{5}{4}\pi + \frac{3}{2}\alpha\right).$$

**Варіант 8**

$$z_1 = \frac{1 - 2 \sin^2 \alpha}{1 + \sin 2\alpha};$$

$$z_2 = \frac{1 - \operatorname{tg} \alpha}{1 + \operatorname{tg} \alpha}.$$

**Варіант 9**

$$z_1 = \frac{\sin 4\alpha}{1 + \cos 4\alpha} \times \frac{\cos 2\alpha}{1 + \cos 2\alpha};$$

$$z_2 = \operatorname{ctg}\left(\frac{3}{2}\pi - \alpha\right).$$

**Варіант 10**

$$z_1 = \frac{\sin \alpha + \cos(2\beta - \alpha)}{\cos \alpha - \sin(2\beta - \alpha)};$$

$$z_2 = \frac{1 + \sin 2\beta}{\cos 2\beta}.$$

**Варіант 11**

$$z_1 = \frac{\cos \alpha + \sin \alpha}{\cos \alpha - \sin \alpha};$$

$$z_2 = \operatorname{tg} 2\alpha + \sec 2\alpha.$$

**Варіант 12**

$$z_1 = \frac{x^2 + 2x - 3 + (x+1)\sqrt{x^2 - 9}}{x^2 - 2x - 3 + (x-1)\sqrt{x^2 - 9}};$$

$$z_2 = \sqrt{\frac{x+3}{x-3}}.$$

**Варіант 13**

$$z_1 = \frac{\sqrt{2b + 2\sqrt{b^2 - 4}}}{\sqrt{b^2 - 4} + b + 2};$$

$$z_2 = \frac{1}{\sqrt{b+2}}.$$

**Варіант 14**

$$z_1 = \frac{\sqrt{(3m+2)^2 - 24m}}{3\sqrt{m} - \frac{2}{\sqrt{m}}};$$

$$z_2 = \sqrt{m}.$$

**Варіант 15**

$$z_1 = \left( \frac{a+2}{\sqrt{2a}} - \frac{a}{\sqrt{2a}+2} + \frac{2}{a-\sqrt{2a}} \right) \times \frac{\sqrt{a}-\sqrt{2}}{a+2};$$

$$z_2 = \frac{1}{\sqrt{a} + \sqrt{2}}.$$

**Варіант 16**

$$z_1 = \frac{(m-1)\sqrt{m} - (n-1)\sqrt{n}}{\sqrt{m^3n} + nm + m^2 - m};$$

$$z_2 = \frac{\sqrt{m} - \sqrt{n}}{m}.$$

**Варіант 17**

$$z_1 = 2 \sin^2(3\pi - 2\alpha) \times \cos^2(5\pi + 2\alpha)$$

$$z_2 = \frac{1}{4} - \frac{1}{4} \sin\left(\frac{5}{2}\pi - 8\alpha\right)$$

**Варіант 18**

$$z_1 = \cos \alpha + \sin \alpha + \cos 3\alpha + \sin 3\alpha;$$

$$z_2 = 2\sqrt{2} \cos \alpha \times \sin\left(\frac{\pi}{4} + 2\alpha\right).$$

**Варіант 19**

$$z_1 = 1 - \frac{1}{4} \sin^2 2\alpha + \cos 2\alpha;$$

$$z_2 = \cos^2 \alpha + \cos^4 \alpha.$$

***Вариант 20***

$$z_1 = \cos \alpha + \cos 2\alpha + \cos 6\alpha + \cos 7\alpha;$$

$$z_2 = 4 \cos \frac{\alpha}{2} \times \cos \frac{5}{2} \alpha \times \cos 4\alpha$$