# Міністерство освіти і науки України Вінницький національний технічний університет Факультет інформаційних технологій та комп'ютерної інженерії Кафедра захисту інформації

# ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №5 '' ФУНКЦІЇ ''

Вико	онав с	тудент г	р. 1БС -2	20б		
-				_ Дмитј	оишин (	E.T.
Лабо	ратор	ну робот	гу захищ	ено		
з оцін	нкою					_
Перез	вірив аф. ЗІ					
	1			ŀ	Сарась С	<u></u> Э.В.
					202	2 p.

#### Завдання №1

ли гооотп

 Створити функцію обчислення п-го члена геометричної прогресії, коли задано її перший член та знаменник.

### Лістинг програми

```
def geom_progression():
  b1 = float(input('Введіть перший член геометричної прогресії: '))
  q = float(input('Введіть знаменник геометричної прогресії: '))
  n = float(input('Введіть номер члена, який потрібно обчислити: '))
  print(b1 * pow(q, n - 1))
geom progression()
```

## Результати роботи програми

```
Введіть перший член геометричної прогресії: 2
Введіть знаменник геометричної прогресії: 4
Введіть номер члена, який потрібно обчислити: 5
512.0
```

#### Завдання №2

2. Напишіть функцію is\_password\_good(password), яка приймає як аргумент рядкове значення пароля password і повертає значення True якщо пароль  $\epsilon$  надійним і False інакше.

Пароль  $\epsilon$  надійним якщо:

- його довжина щонайменше 8 символів;
- він містить як мінімум одну заголовну літеру (верхній регістр);
- він містить як мінімум одну малу літеру (нижній регістр);
- він містить хоч би одну цифру.

### Лістинг програми

```
def is_password_good(password):
    result = True
    if len(password) < 8:
        result = False
    if password.lower() == password:
        result = False
    if password.upper() == password:
        result = False
    if password.isalpha():
        result = False
    return result
txt = input('BBeдiть пароль: ')
print(is password good(txt))</pre>
```

## Результати роботи програми

```
Введіть пароль: 15
False
```

#### Завдання №3

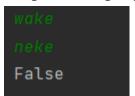
 Напишіть функцію is\_one\_away(word1, word2), яка приймає як аргументи два слова word1 і word2 і повертає значення True, якщо слова мають однакову довжину і відрізняються рівно в 1 символі і False в іншому випадку.

### Лістинг програми

```
def is_one_away(word1, word2):
    a = 0
    for i in range(len(word1)):
        if word1[i] != word2[i]:
            a += 1
    return len(word1) == len(word2) and a == 1
txt1 = input()
txt2 = input()
print(is one away(txt1, txt2))
```

### Результати роботи програми





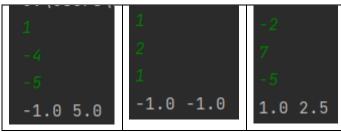
#### Завдання №4

4. Напишіть функцію solve(a, b, c), яка приймає як аргументи три цілих числа a, b, c — коефіцієнти квадратного рівняння  $ax^2+bx+c=0$ ., і повертає його корені порядку зростання.

## Лістинг програми

```
def solve(a, b, c):
    d = (b ** 2) - 4 * a * c
    x1 = ((-1 * b) - d ** 0.5) / (2 * a)
    x2 = ((-1 * b) + d ** 0.5) / (2 * a)
    return min(x1, x2), max(x1, x2)
a, b, c = int(input()), int(input()), int(input())
x1, x2 = solve(a, b, c)
print(x1, x2)
```

## Результати роботи програми



Завдання №5

Опис проекту: програма генерує випадкове число в діапазоні від 1 до 100 і просить користувача вгадати це число. Якщо здогад користувача більше випадкового числа, то програма повинна вивести повідомлення 'Занадто багато, спробуйте ще раз'. Якщо здогад менше випадкового числа, то програма повинна вивести повідомлення 'Занадто мало, спробуйте ще раз'. Якщо користувач вгадує число, програма повинна привітати його та вивести повідомлення 'Ви вгадали, вітаємо!'.

Користувач потенційно може ввести невірні дані, наприклад, не число, або число, що перевищує 100. Важливо передбачити таку можливість, щоб програма продовжувала працювати правильно. Обробка таких ситуацій називається захистом від дурня.

Напишіть функцію is\_valid(), в яку передається один рядковий аргумент. Функція повертає значення True якщо переданий аргумент є цілим числом від 1 до 100 і False інакше.

Додайте підрахунок спроб, зроблених користувачем. Коли число відгадано, програма має показати кількість спроб;

Додайте можливість генерації нового числа (повторна гра) після того, як користувач вгадав число;

Додайте можливість вказівки правого кордону для випадкового вибору числа (від 1 до n).

## Лістинг програми

```
import random
def is valid(n):
   return True if isinstance(n, (int, float)) or 0 < n < 100 else False
def validate range(n):
   return n in range(1, 100)
def input num():
       n = int(input("Введіть число у проміжку(1,100): "))
    except ValueError as e:
       print(f"Ви ввели неправильний тип даних, ваше число буде рандомно
вибрано")
       n = random.randint(1, 100)
    while not validate range(n):
       print("Ви вийшли за рамки проміжку, проміжок від 1 до 100!!!")
       n = int(input("Введіть число у проміжку(1,100): "))
   return n
def play():
   mistake count = 0
   global answer
   n = input num()
   guess number = random.randint(1, n)
    try:
```

```
answer = int(input("Загадане число: "))
    except not is valid(answer) or TypeError as e:
       raise e
    while answer != guess number:
        if answer > guess_number:
            print('Занадто багато, спробуйте ще раз ')
        elif answer < guess number:</pre>
            print('Занадто мало, спробуйте ще раз ')
        mistake\_count += 1
        answer = int(input(f"Загадане число: ви зробили {mistake count} помилку
"))
    print(f'Ви вгадали, вітаємо! число {guess number}, кількість помилок =
{mistake count}')
    play again = input("Зіграємо ще раз? yes/no ")
    return True if play again == 'yes' else False
def run():
    while True:
        if not play():
            print("Приходьте ще! ")
            break
run()
```

### Результати роботи програми

```
Введіть число у проміжку(1,100):
                                                     Введіть число у проміжку(1,100):
Загадане число:
                                                     Ви вийшли за рамки проміжку, проміжок від 1 до 100!!
Занадто мало, спробуйте ще раз
                                                     Введіть число у проміжку(1,100):
Загадане число: ви зробили 1 помилку
                                                     Загадане число:
Занадто мало, спробуйте ще раз
                                                     Занадто мало, спробуйте ще раз
Загадане число: ви зробили 2 помилку
                                                     Загадане число: ви зробили 1 помилку 🛭
Занадто мало, спробуйте ще раз
                                                     Занадто мало, спробуйте ще раз
Загадане число: ви зробили 3 помилку
                                                     Загадане число: ви зробили 2 помилку
Ви вгадали, вітаємо! число 4, кількість помилок = 3
Зіграємо ще раз? yes/no
                                                     Зіграємо ще раз? yes/no
Приходьте ще!
```

#### Висновок

Застосовано на практиці використання функцій у мові програмування Python.