**Міністерство освіти і науки України**

**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Факультет інформатики та обчислювальної техніки**

**Кафедра обчислювальної техніки**

**Лабораторна робота № 4**

з дисципліни «Системне програмування»

на тему «Макровизначення і макроси у MASM32»

Виконав: Перевірив:

студент групи ІМ-12 доцент Павлов В.Г.

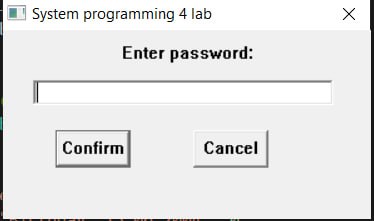
Креславський Михайло Олегович

Номер у списку групи: 11

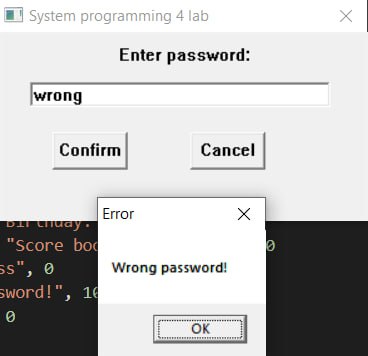
**Київ 2023**

**Скріншоти виконання програми**

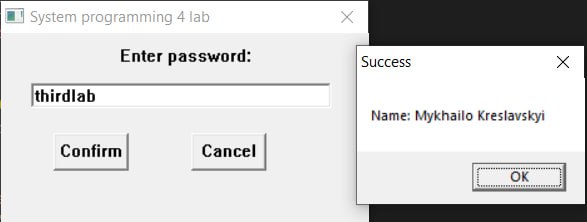
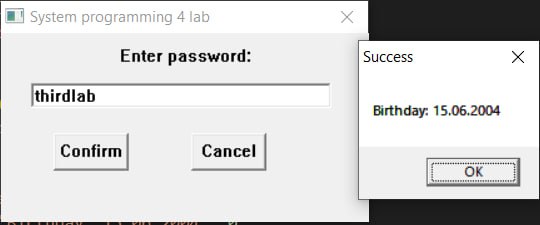
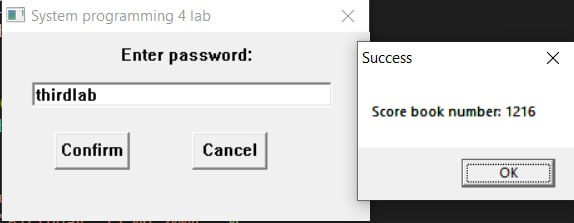
Вікно введення паролю:



При введенні неправильного паролю:



При введенні правильного паролю:

1. 
2. 
3. 

В програмі було створено 3 макроси:

1. displayMessage – приймає аргументи title та content і виводить на екран окреме вікно з отриманою інформацією.
2. encryptInput – шифрує пароль, який ввів користувач.
3. compare – порівнює зашифрований пароль із зашифрованим паролем, який ввів користувач.

**Шифрування паролю за допомогою ключа**

Пароль, який ввів користувач, шифрується за допомогою метода XOR, а потім отриманий рядок порівнюється із зашифрованим правильним паролем.

Процес шифрування:

Правильний пароль – thirdlab

Ключ – E88FQ9V4

t = 11610 = 11101002 XOR E = 6910 = 10001012 🡺 1100012 = 4910 = 1

h = 10410 = 11010002  XOR 8 = 5610 = 1110002 🡺 10100002 = 8010 = P

i = 10510 = 11010012 XOR 8 = 5610 = 1110002 🡺 10100012 = 8110 = Q

r = 11410 = 11100102 XOR F = 7010 = 10001102 🡺 1101002 = 5210 = 4

d = 10010 = 11001002 XOR Q = 8110 = 10100012 🡺 1101012 = 5310 = 5

l = 10810 = 11011002 XOR 9 = 5710 = 1110012 🡺 10101012 = 8510 = U

a = 9710 = 11000012 XOR V = 8610 = 10101102 🡺 1101112 = 5510 = 7

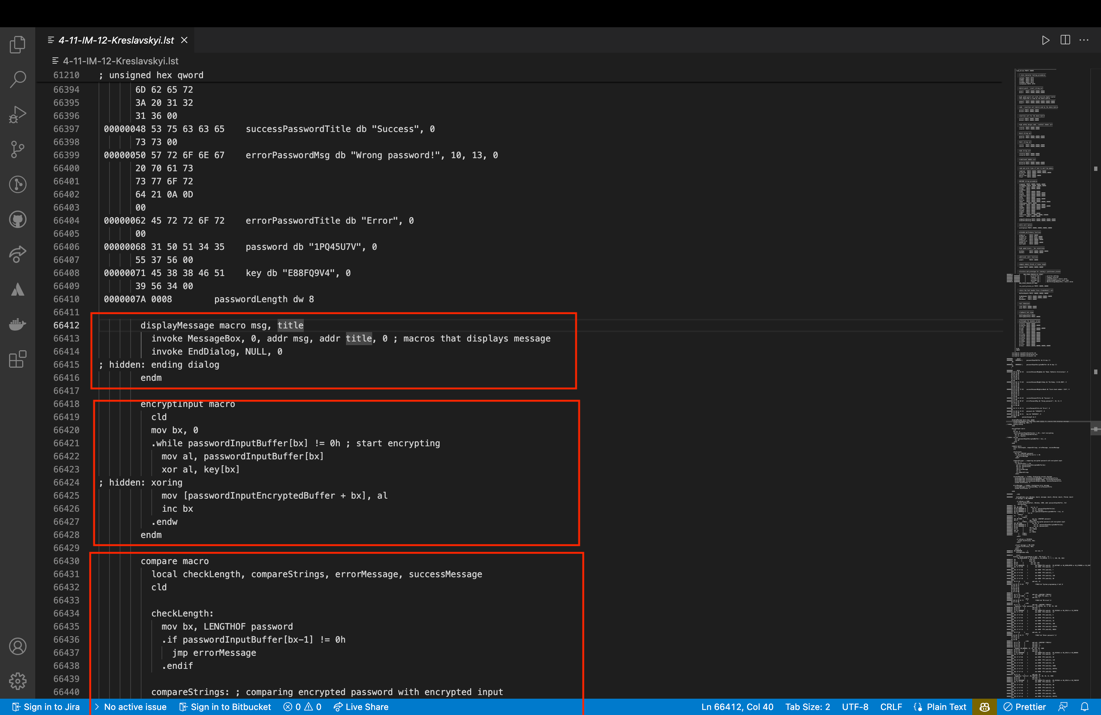
b = 9810 = 11000102 XOR 4 = 5210 = 1101002 🡺 10101102 = 8610 = V

Отже, зашифрований пароль – 1PQ45U7V

**Дослідження файлів лістингу**

1. Файл 4-11-ІМ-12-Kreslavskyi.asm, в якому макроси написані прямо у файлі з кодом

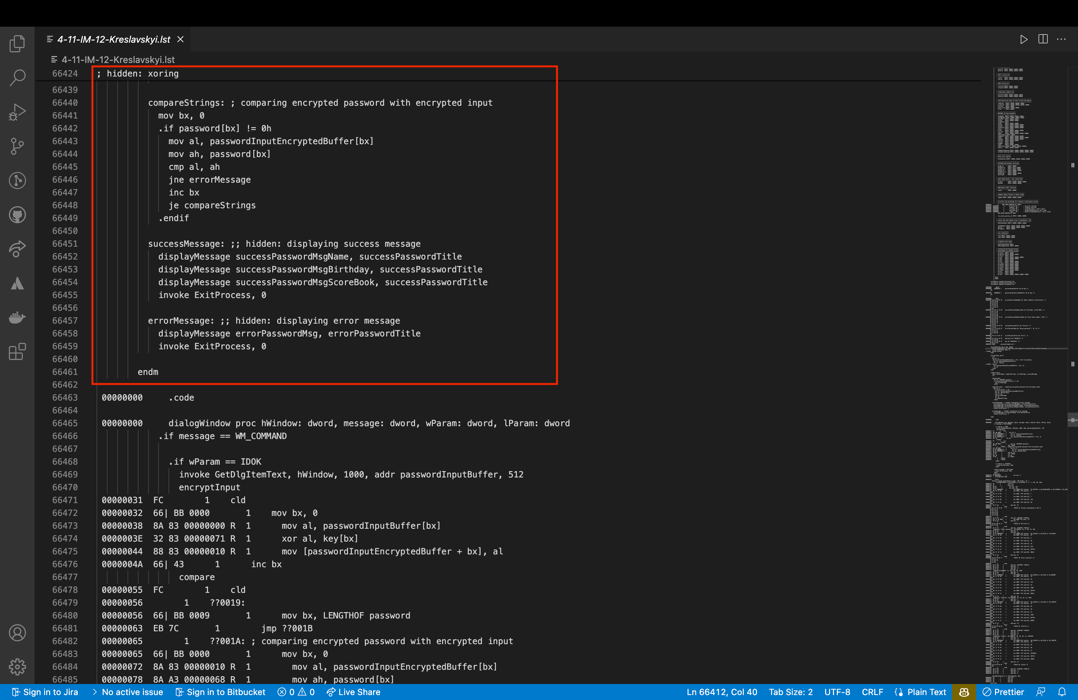
Досліджуючи файл 4-11-ІМ-12-Kreslavskyi.lst ми бачимо, що макровизначення йдуть після визначень всіх константних даних, так само, як і у файлі 4-11-ІМ-12-Kreslavskyi.asm. Також видно і звичайні, і приховані коментарі.



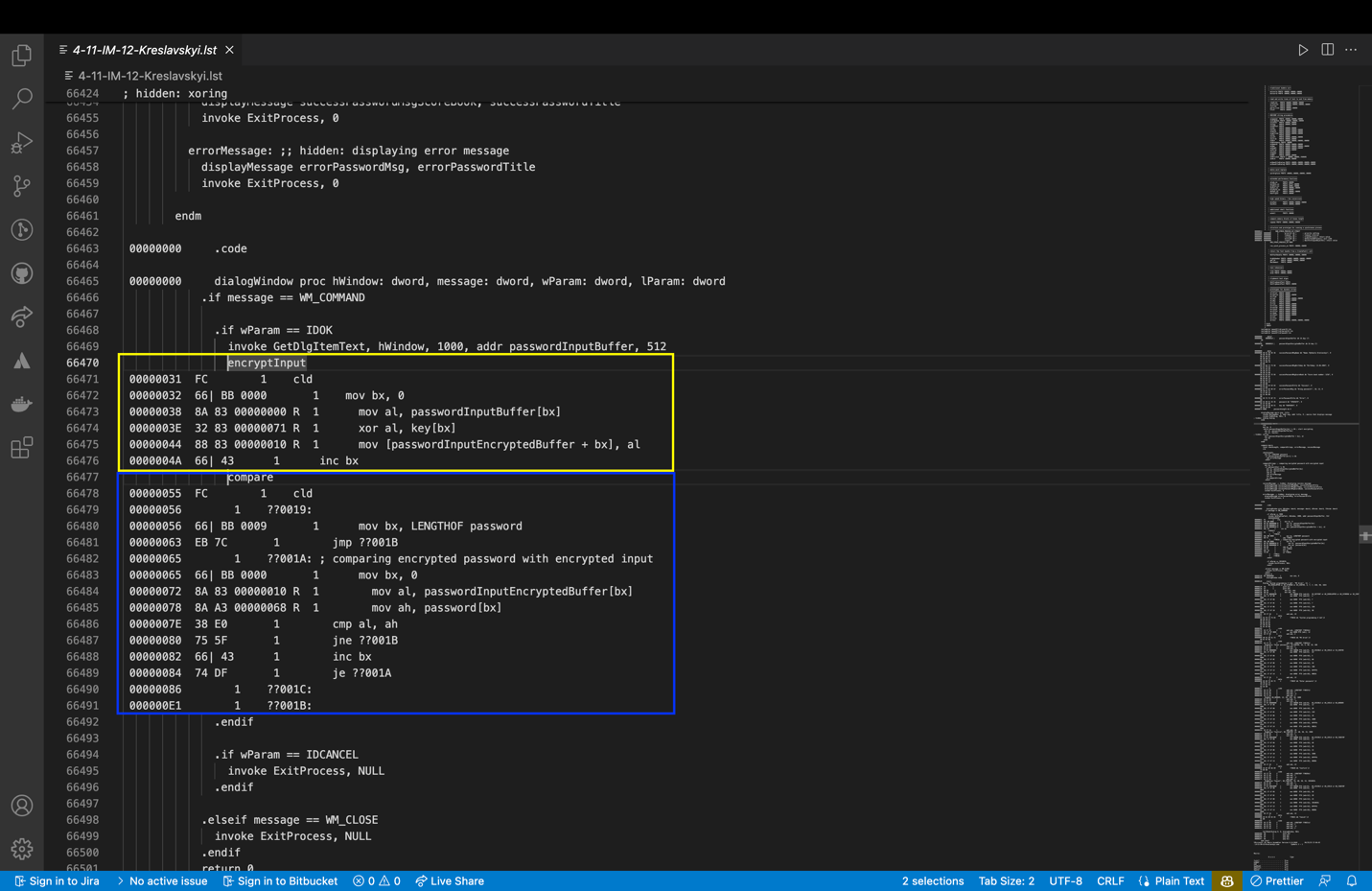
3 макрос

2 макрос

1 макрос

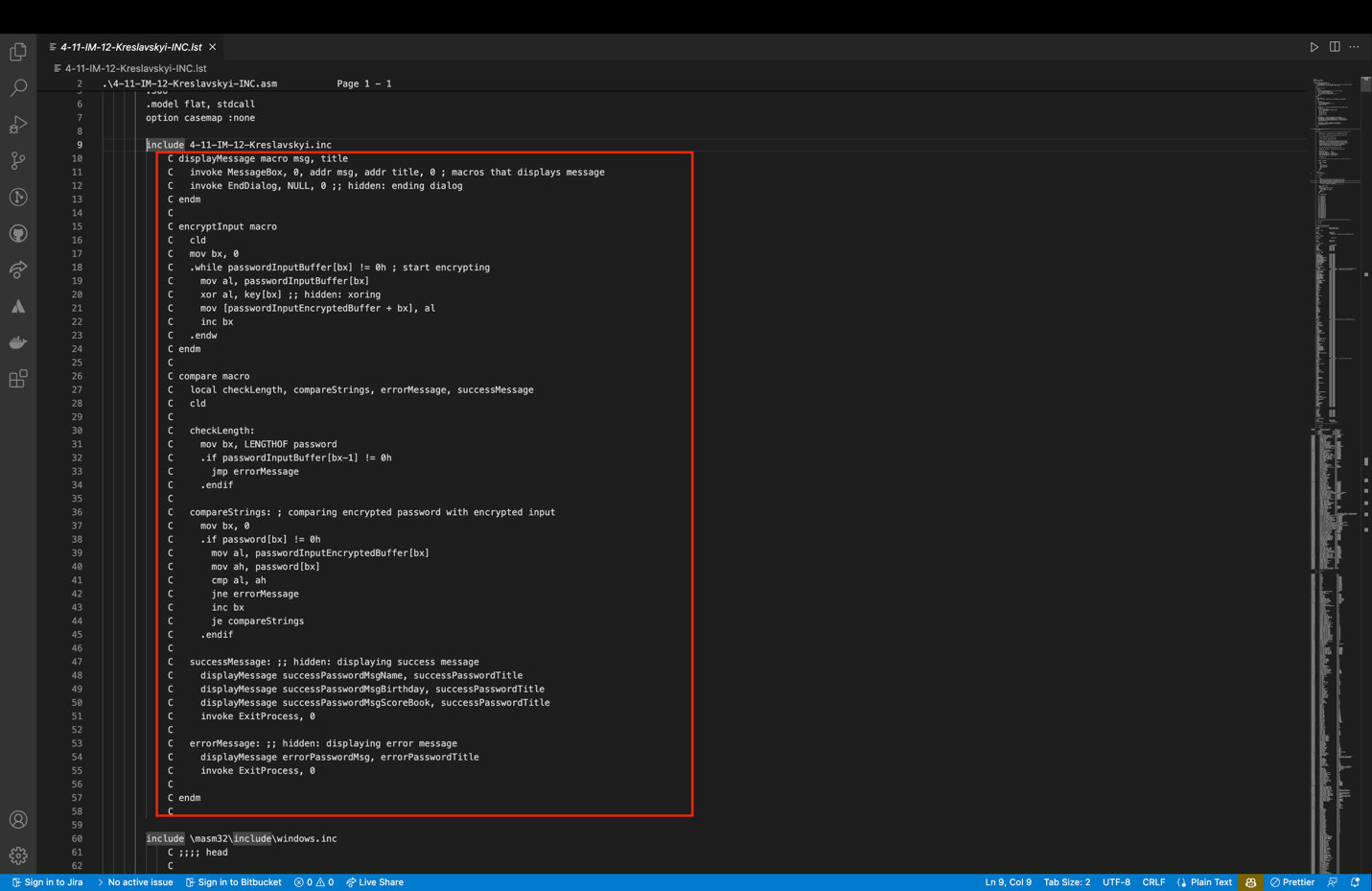


продовження 3 макросу

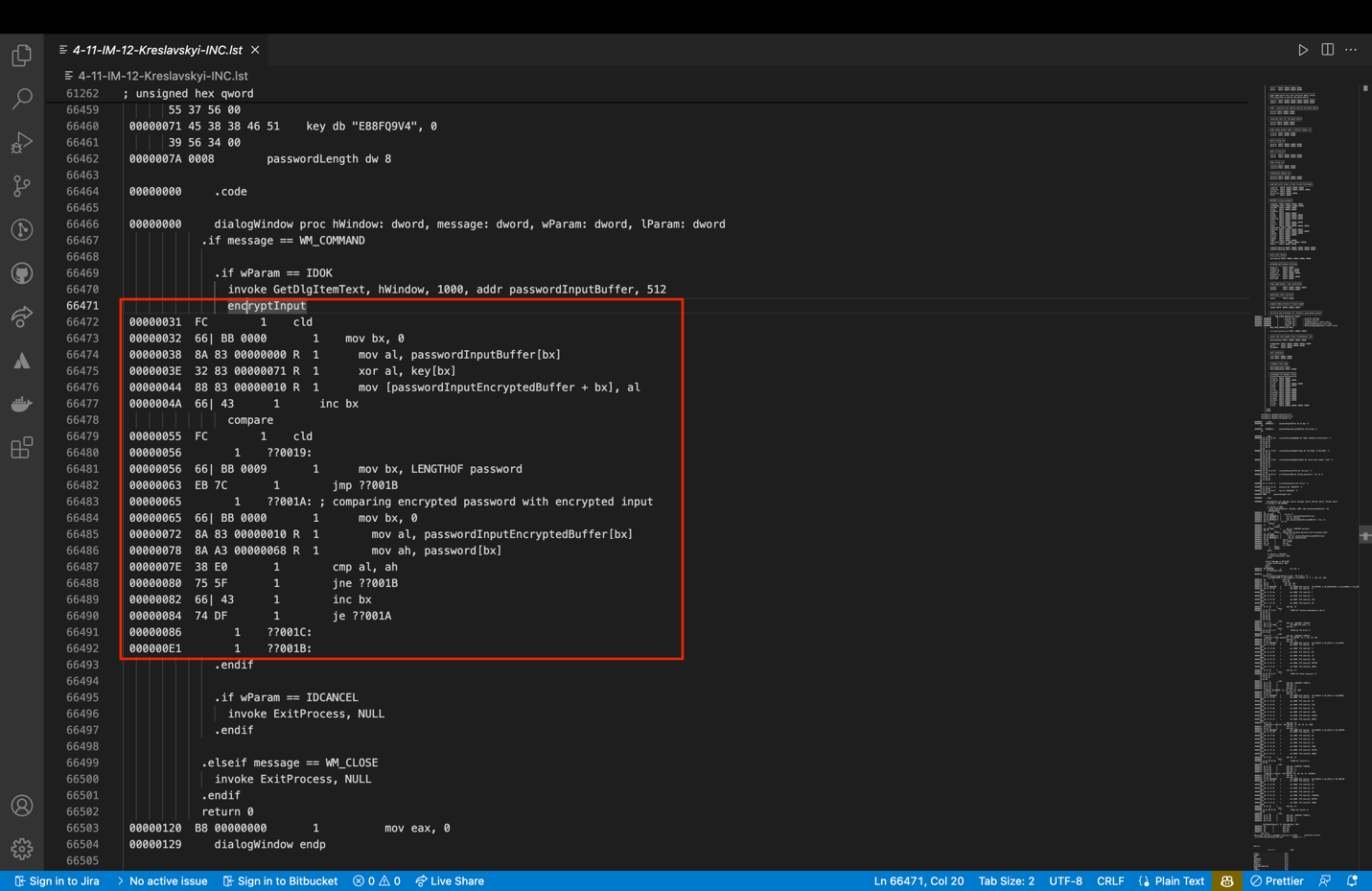
На цьому скріншоті показані трансляції 2 та 3 макросів (жовтий та синій відповідно)

Тут вже не видно прихованих коментарів, а також викликів функцій за допомогою слова invoke.

1. Файл 4-11-ІМ-12-Kreslavskyi.asm, в якому макроси були підключені з файлу 4-11-ІМ-12-Kreslavskyi.inc за допомогою include.



Досліджуючи файл 4-11-ІМ-12-Kreslavskyi-INC.lst, ми бачимо, що макроси підключені до .asm файлу за допомогою include та знаходяться одразу після інструкції їх підключення. Також бачимо, що видно як приховані, так і звичайні коментарі.



Далі, на цьому скріншоті із демонстрацією трансляції макросів, бачимо таку ж картину, як і у минулому випадку із файлом, де ці макроси не були підключені з окремого .inc файлу.

Порівнявши 2 файли лістингу, можна сказати, що вони мало чим відрізняються один від одного – тільки знаходженням макровизначень. В першому випадку там, де вони були написані (після визначень константних даних), в другому – там, де вони буди підключені, на початку файлу з основним кодом. В обох випадках видно як приховані, так і звичайні коментарі і так само в обох випадках в трансляції макросів не видно прихованих коментарів та викликів функцій за допомогою invoke.

**Висновок**

В цій лабораторній роботі я ознайомився із макросами в мові Асемблера, зрозумів принцип їх роботи та навчився писати макроси власноруч. Також я спробував 2 способи їх використання: оголошенням макросів в коді та імпортуванням їх із файлу. Досліджуючи файли лістингу, я знайшов багато спільного у цих двох способах використання макросів: в обох випадках видно як приховані, так і звичайні коментарі і так само в обох випадках в трансляції макросів не видно прихованих коментарів та викликів функцій за допомогою invoke. Але також були і відмінності, наприклад, у розташуванні макровизначень у файлах лістингу.

Маю сказати, що макроси дуже зручні у використанні та нескладні у написанні. Вони значно підвищують зрозумілість коду та зменшують кількість його повторення, при цьому не змінюючи логіку – як ми бачили у файлах лістингу, макроси транслюються так, як вони є, якщо не приймають аргументів, або, якщо приймають аргументи, то транслюються вже з аргументами.