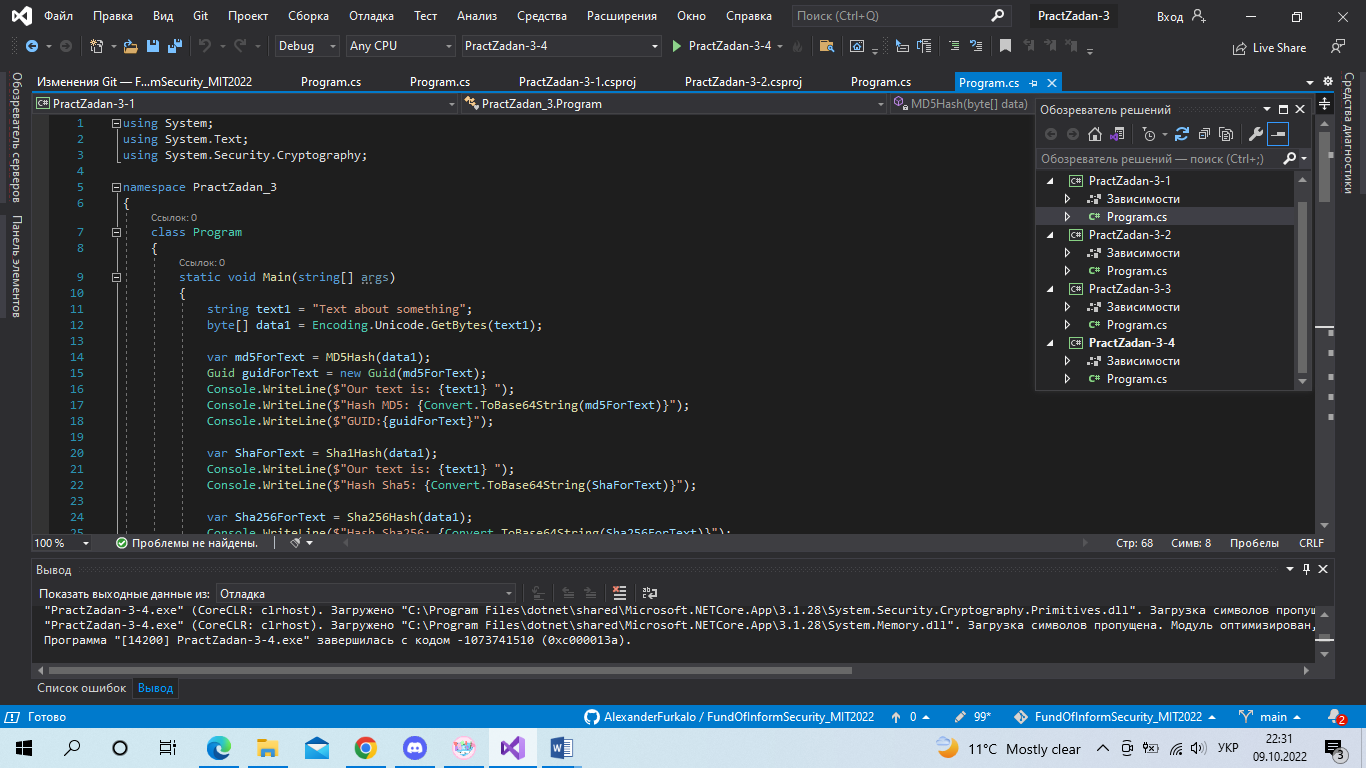
ПРАКТИЧНА РОБОТА №2  
З ОСНОВ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ  
СТУДЕНТА КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ ІМЕНІ ТАРАСА ШЕВЧЕНКО  
ФАКУЛЬТЕТУ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРИ МЕРЕЖЕВИХ ТА ІНТЕРНЕТ ТЕХНОЛОГІЙ  
ДРУГОГО КУРСУ, ДРУГОЇ ПІДГРУПИ  
ОЛЕКСАНДРА ОЛЕКСАНДРОВИЧА ФУРКАЛА  
ЗА ТЕМОЮ “ХЕШ-ФУНКЦІЇ ТА ПЕРЕВІРКА ЦІЛІСНОСТІ ІНФОРМАЦІЇ”  
ЗВІТ  
09.10.2022

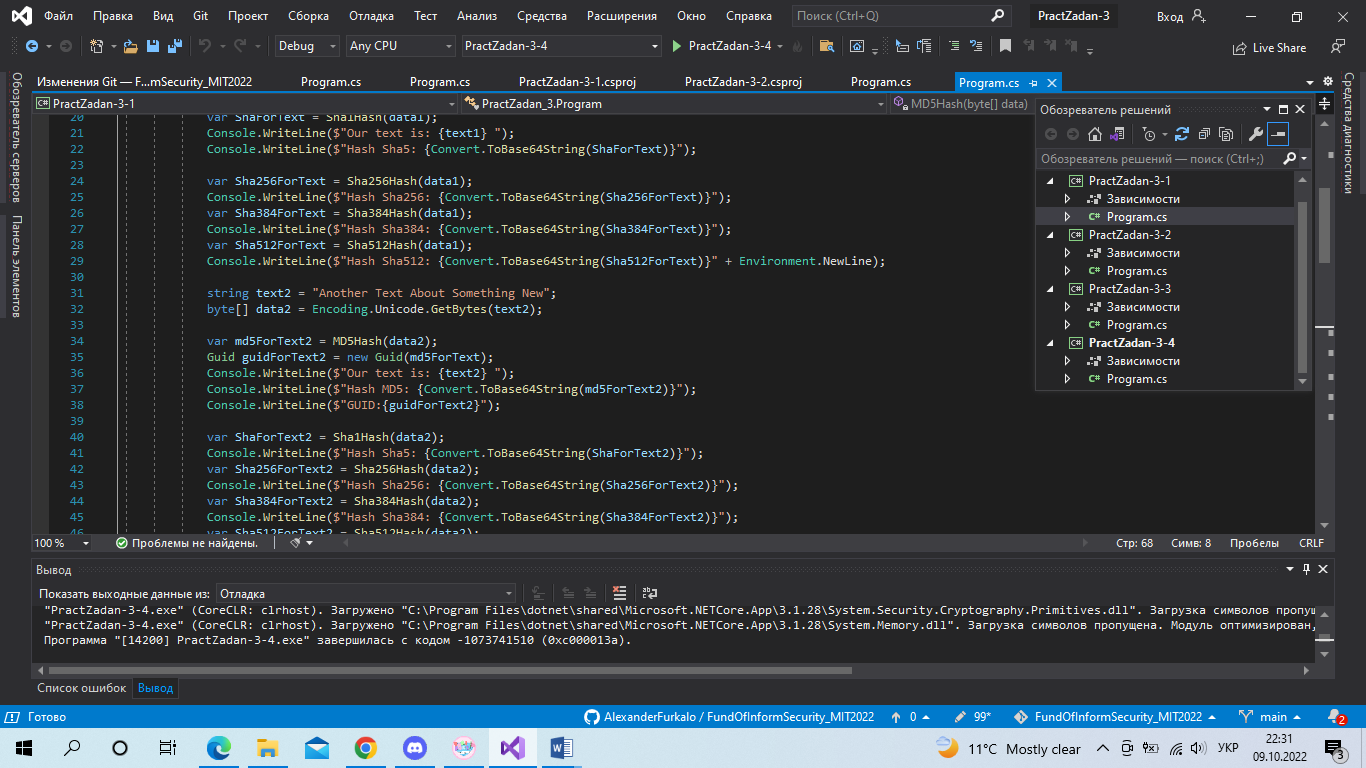
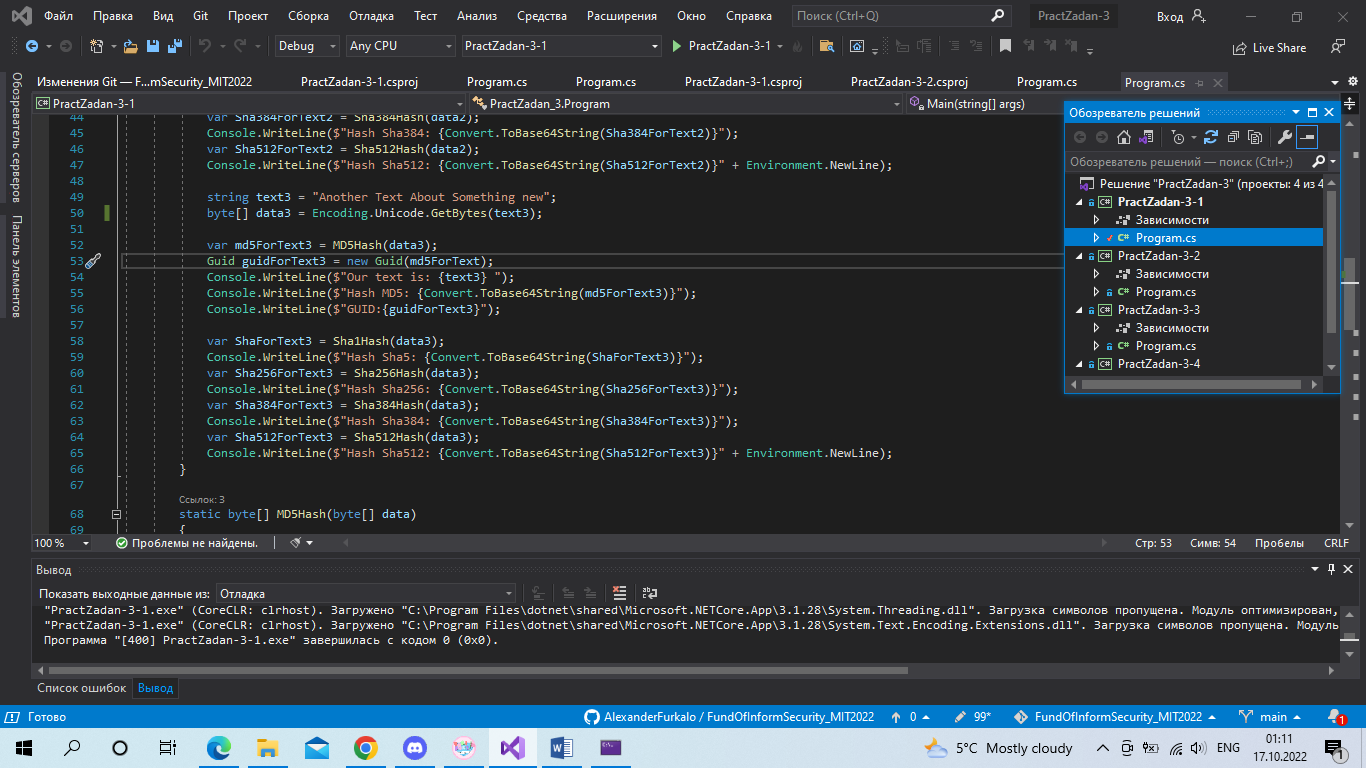
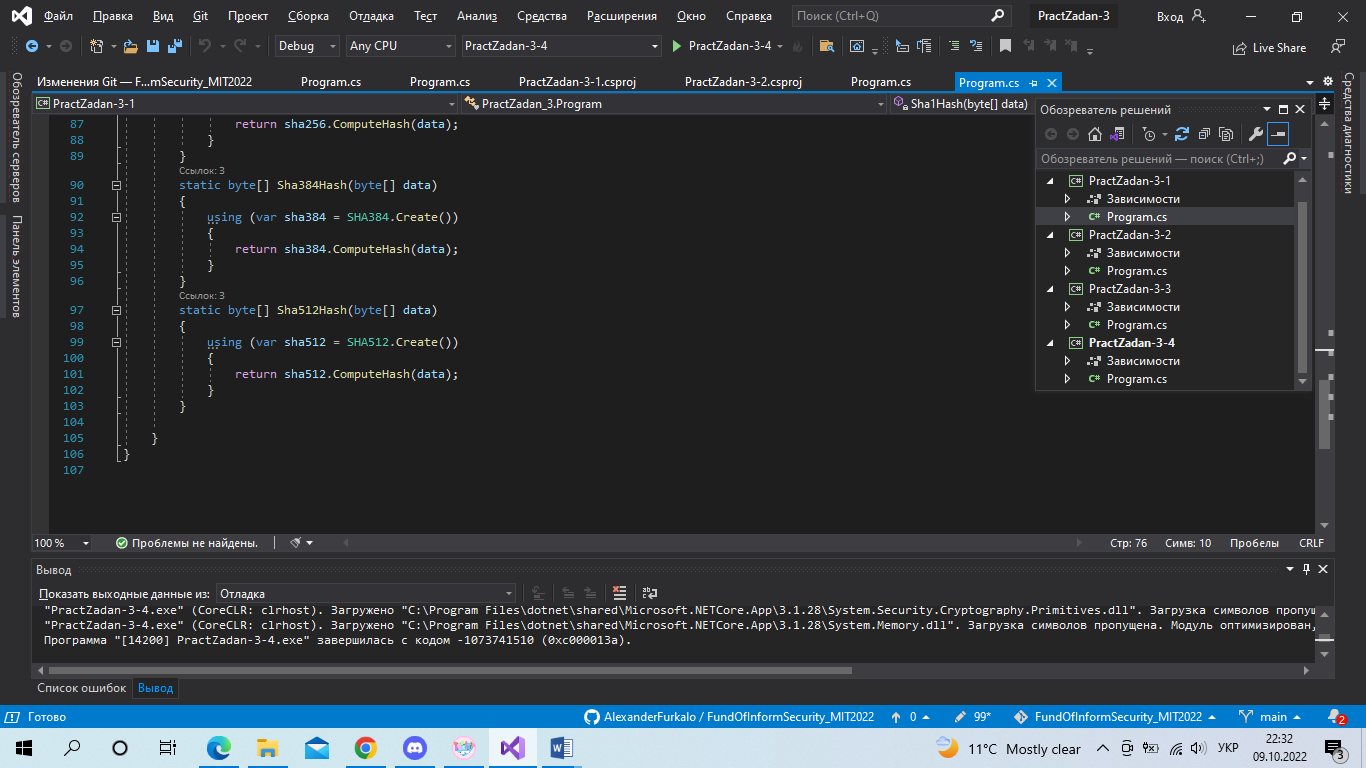
Завдання цієї практичної роботи були такі:  
1. Написати програму, яка обчислює хеш-коди за всіма відомими алгоритмами для заданих даних. Порівняти розміри хеш-кодів та значення для однакових та різних даних. Зробити висновки.  
2. Відомо, що користувач використав для пароля довжиною 8 символів лише цифри. Шляхом пасивного прослуховування мережі отримано MD5 хеш-код цього пароля:  
{564c8da6-0440-88ec-d453-0bbad57c6036}

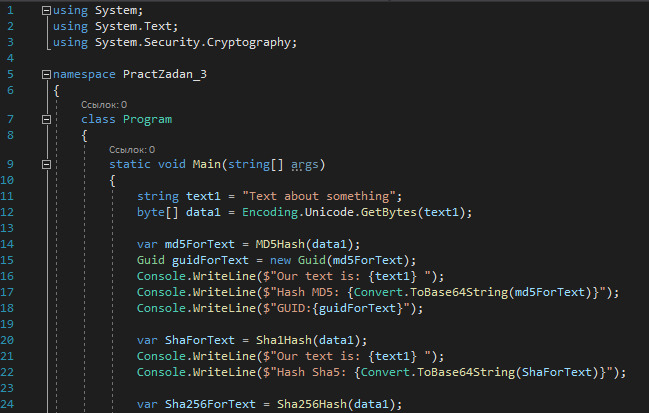
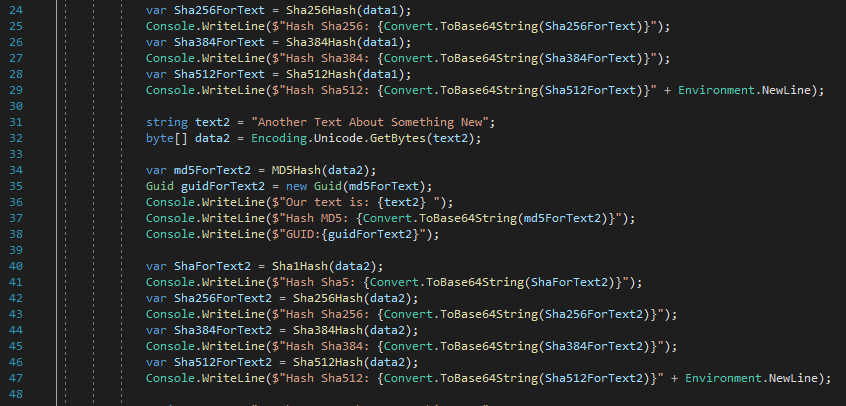
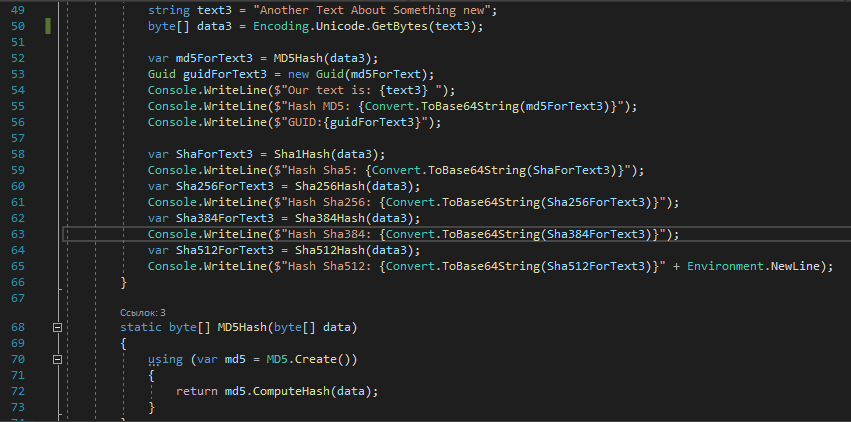
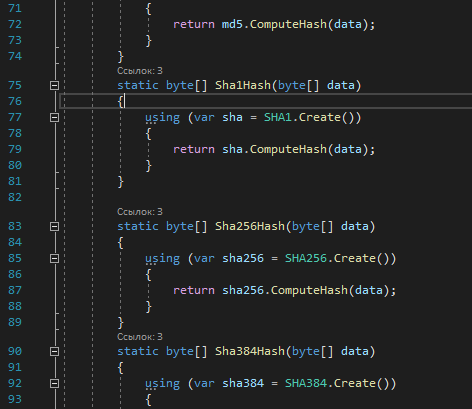
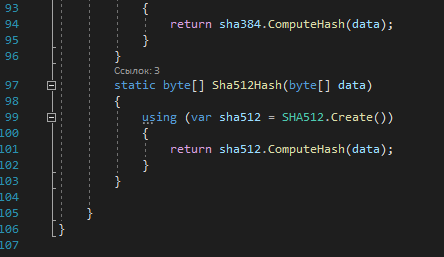
та

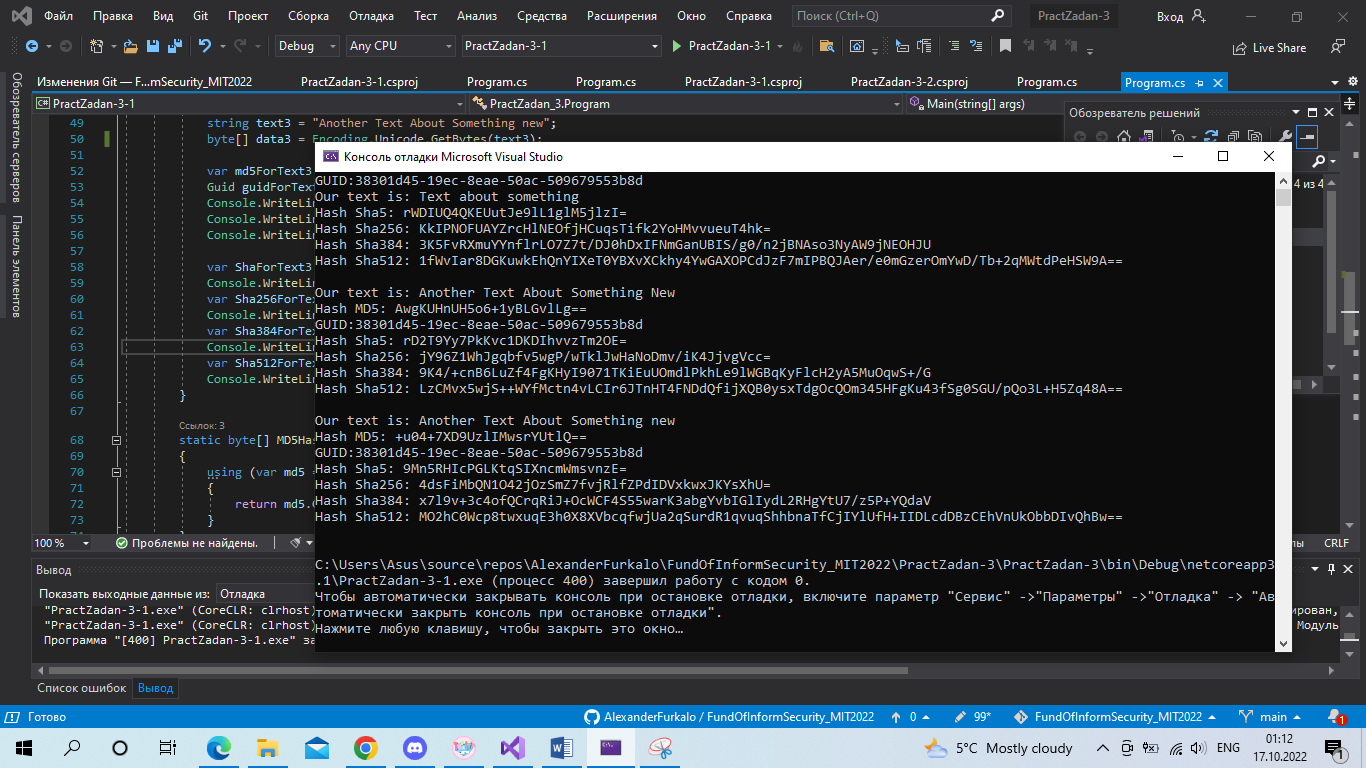
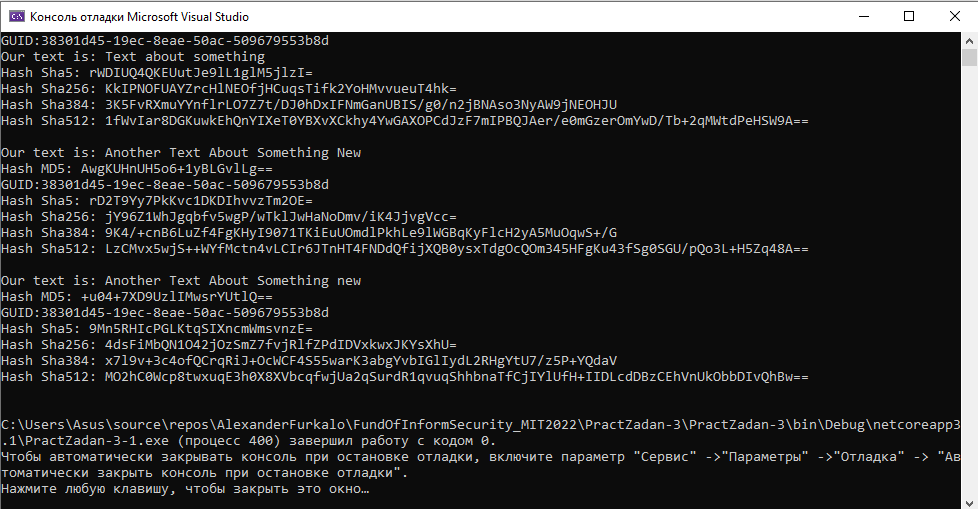
po1MVkAE7IjUUwu61XxgNg==  
Відновити пароль користувача та зробити висновки про надійність такого пароля.  
3. Написати програму для обчислення хеш-коду автентифікації повідомлення а також реалізувати можливість перевірки автентичності повідомлення.

4. Написати програму для реєстрації користувача за логіном/паролем та авторизації шляхом співставлення відповідних логінів і паролів. Зберігання пароля у відкритому вигляді неприпустиме  
5. Оформити Звіт

Перейдемо до першого з чотирьох кодів:  


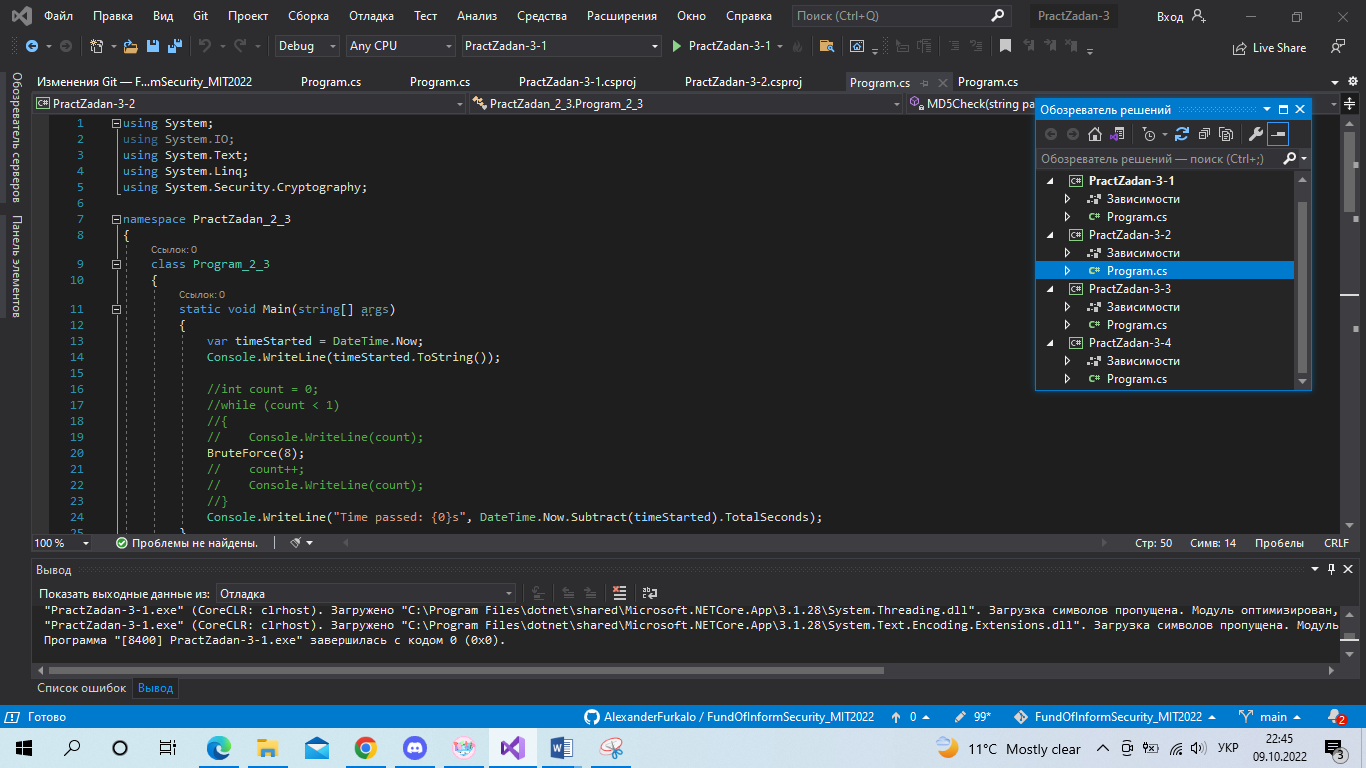
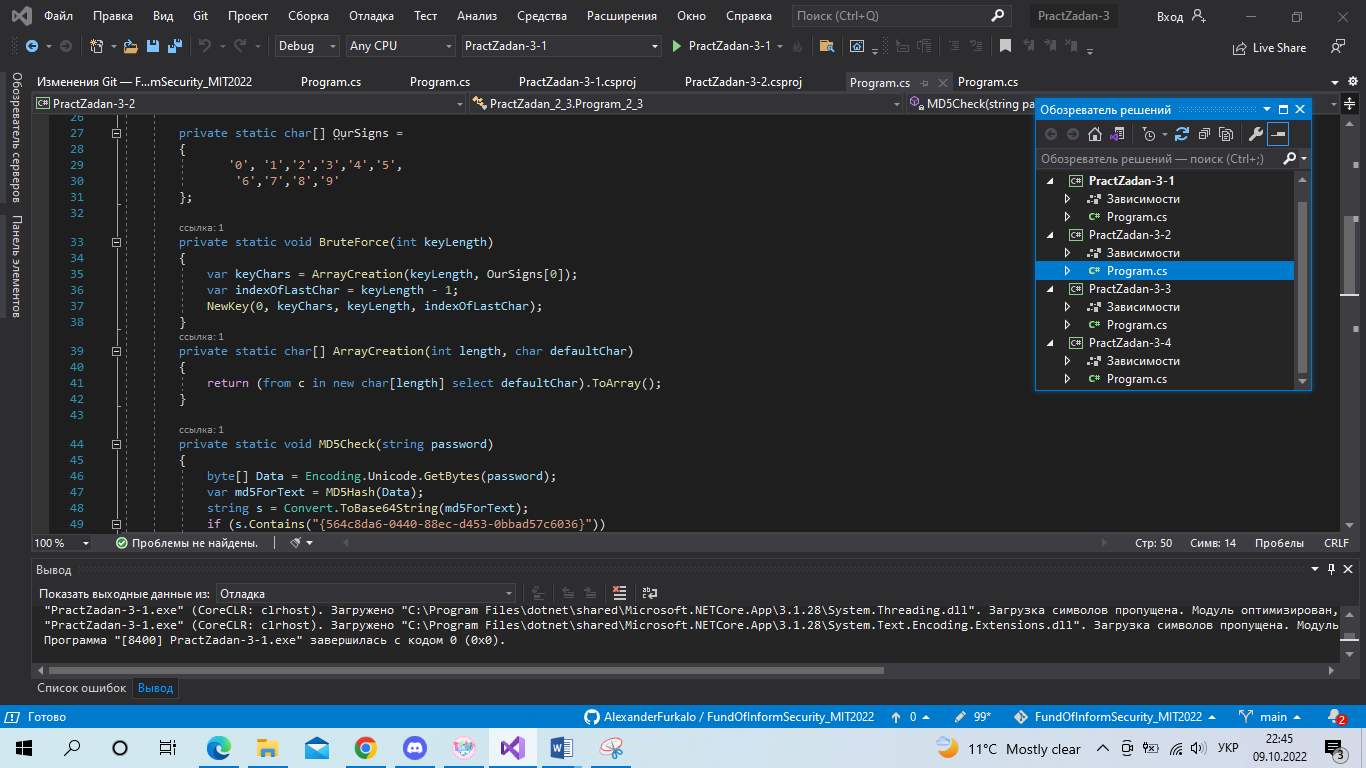
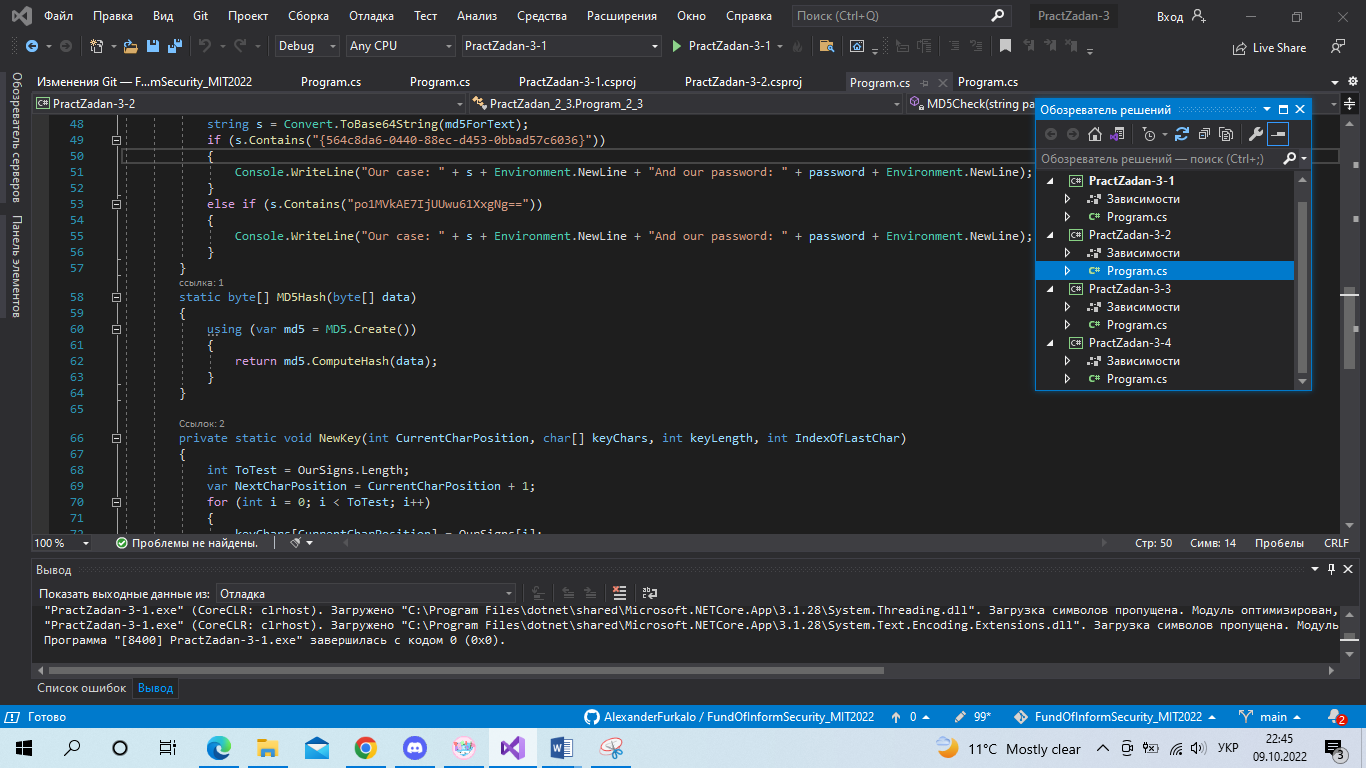
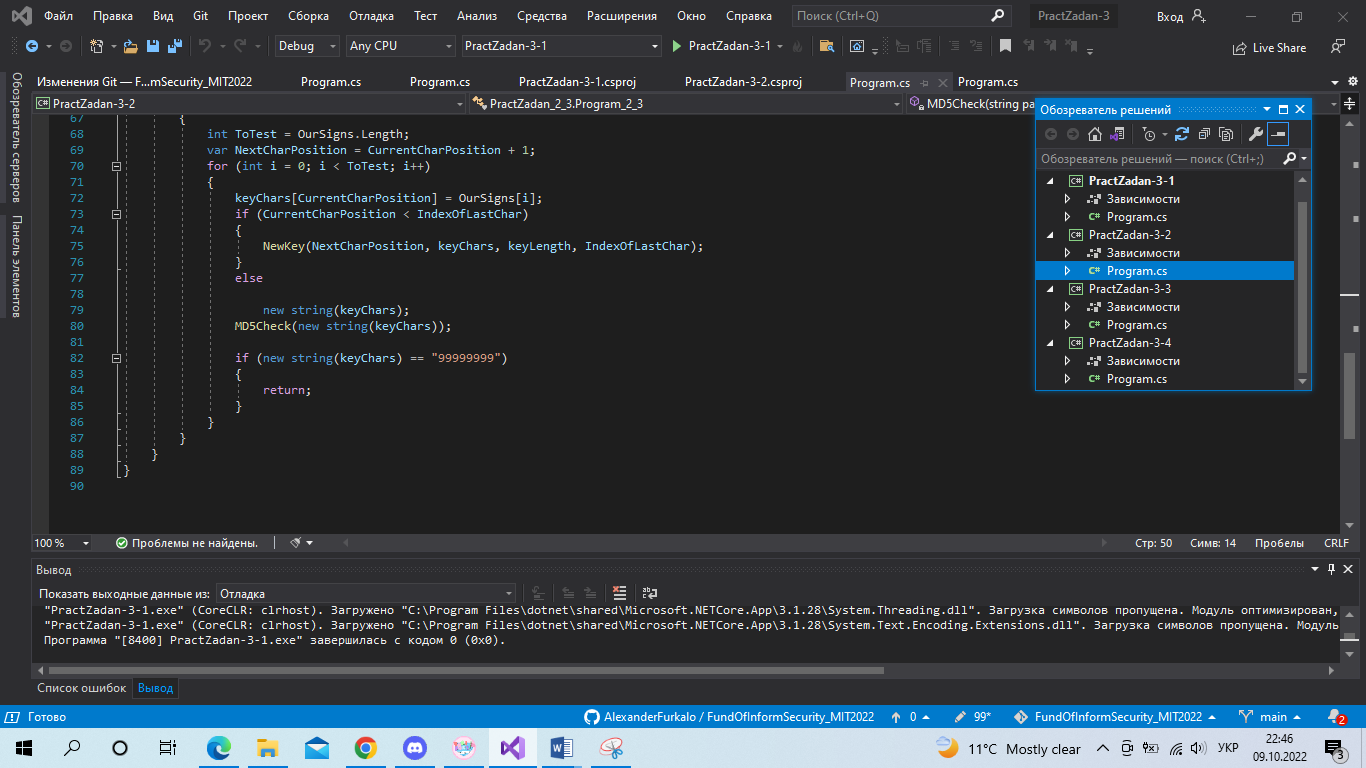
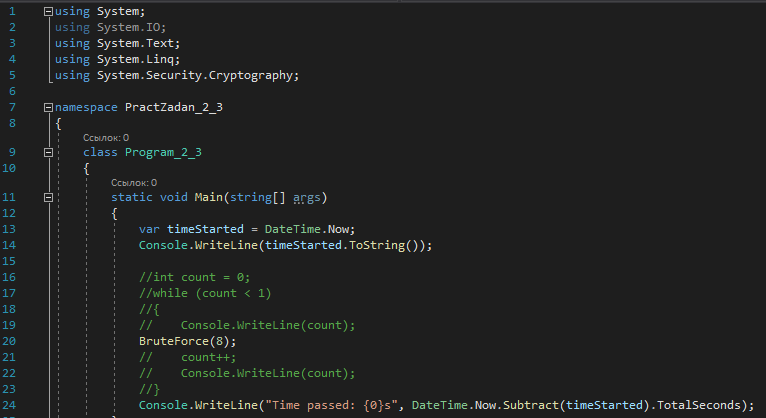
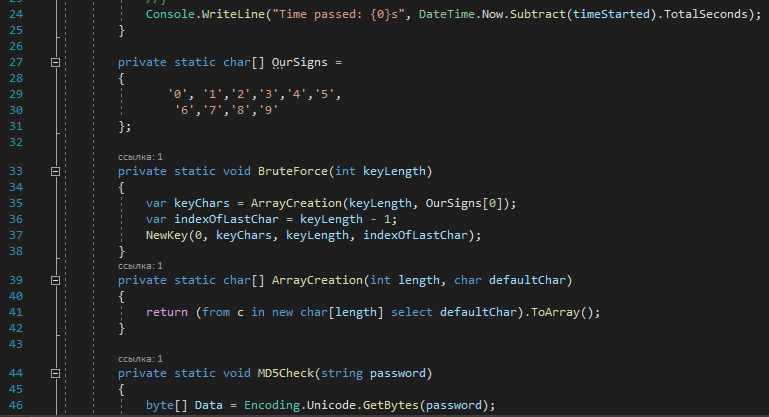
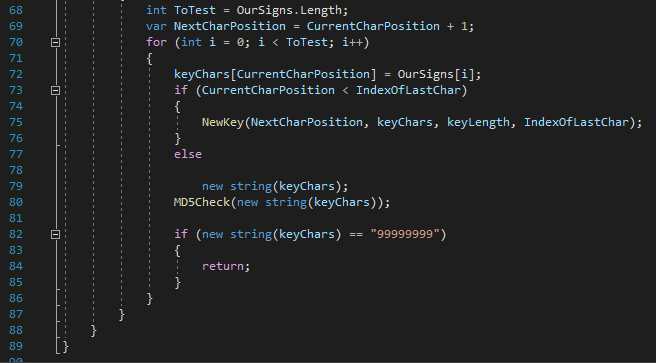
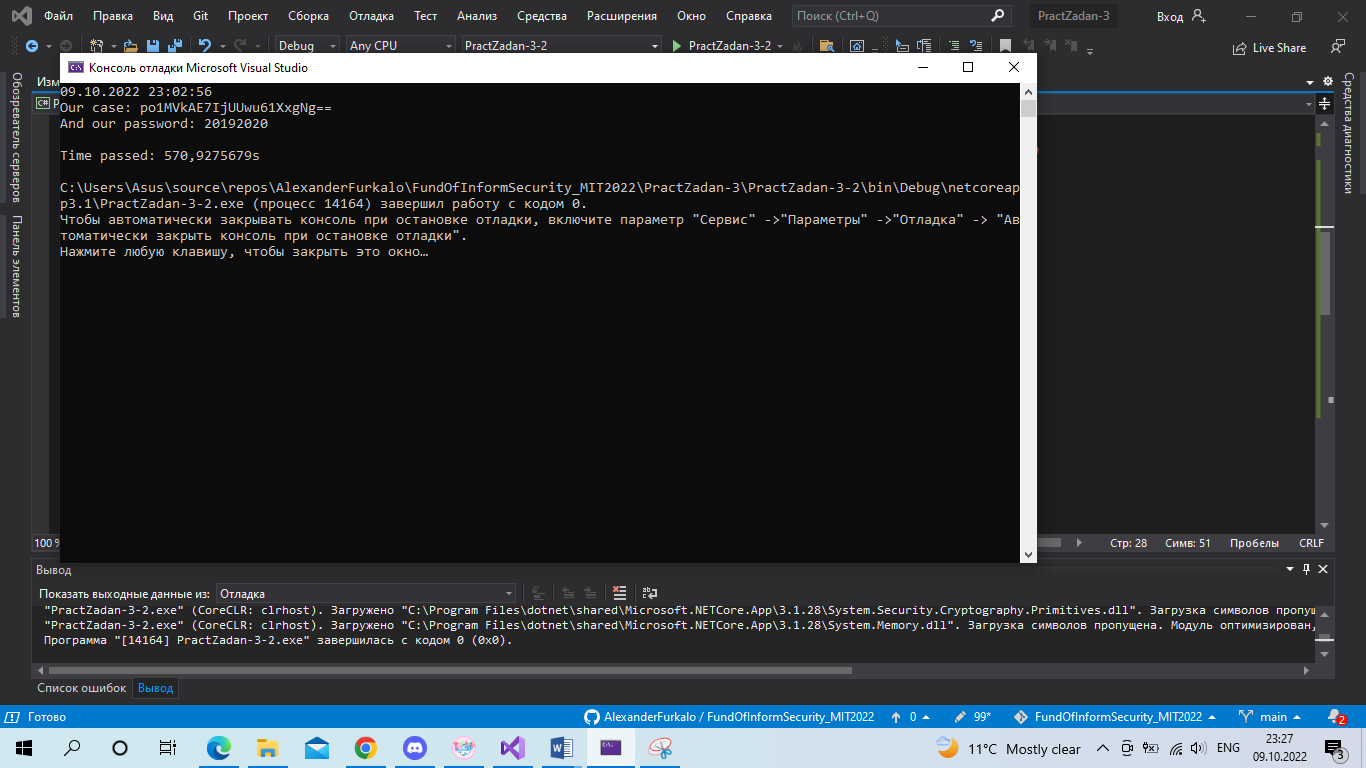
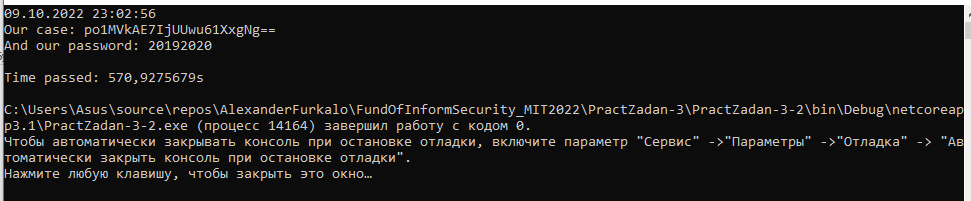
  
  
  


Трохи наблизивши:  
  
  
  
  


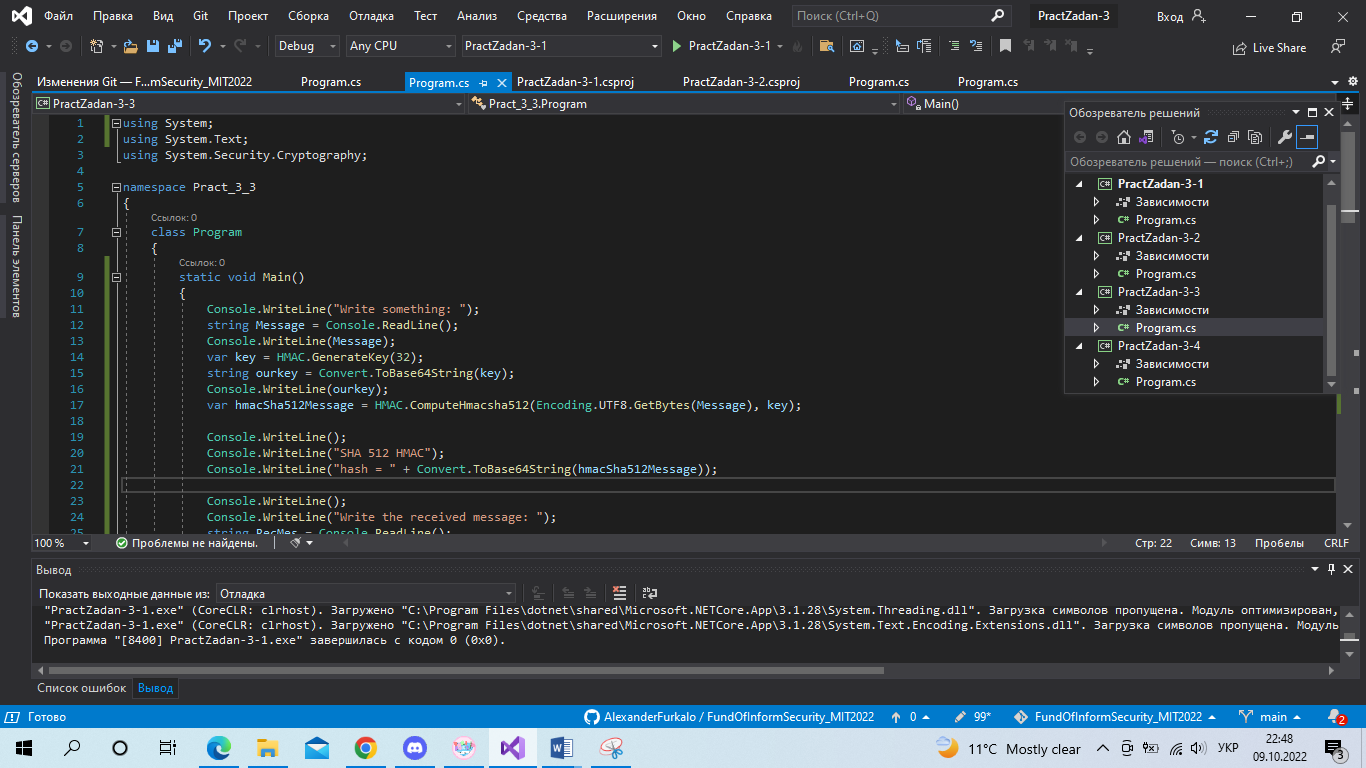
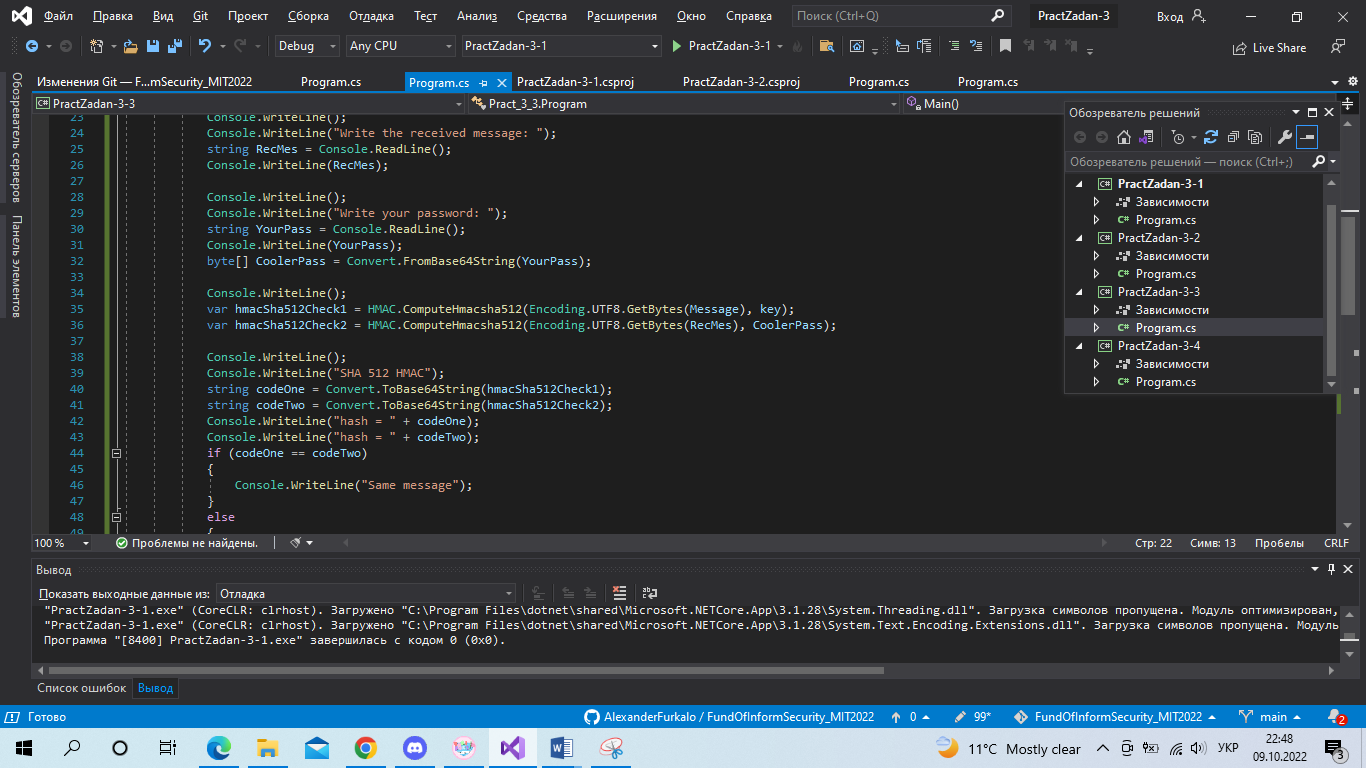
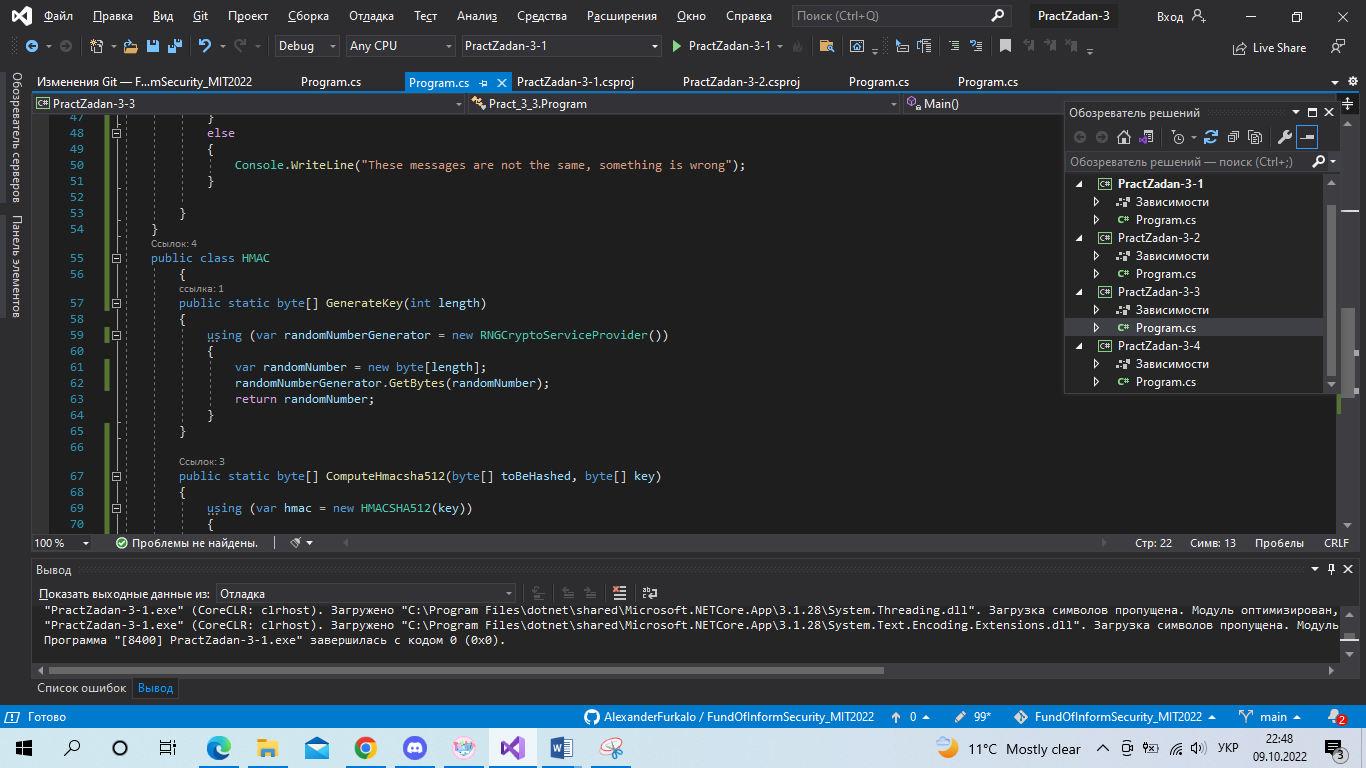
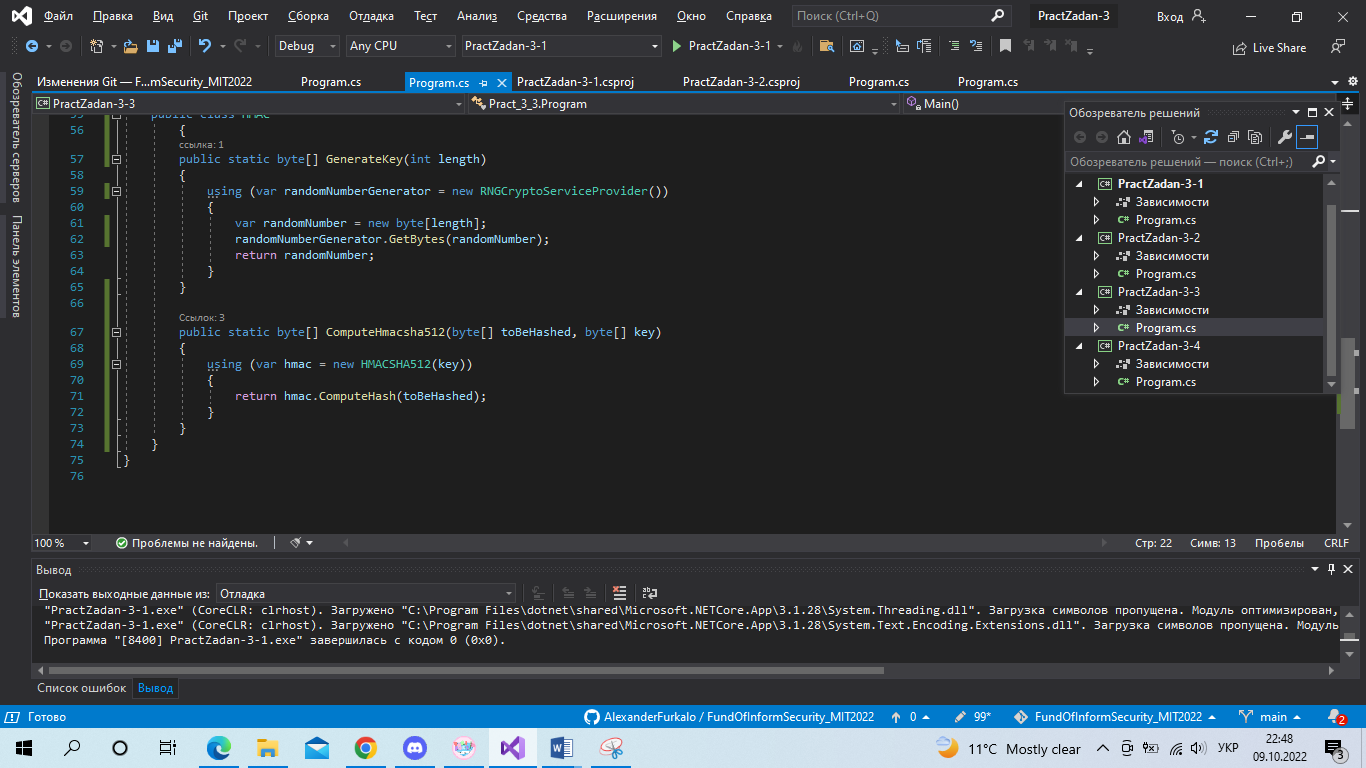
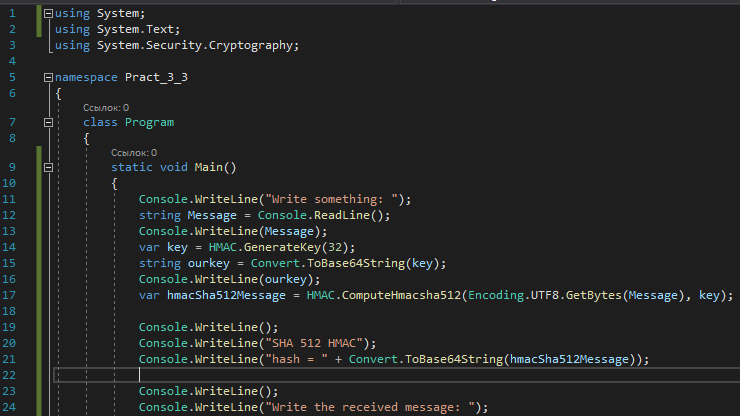
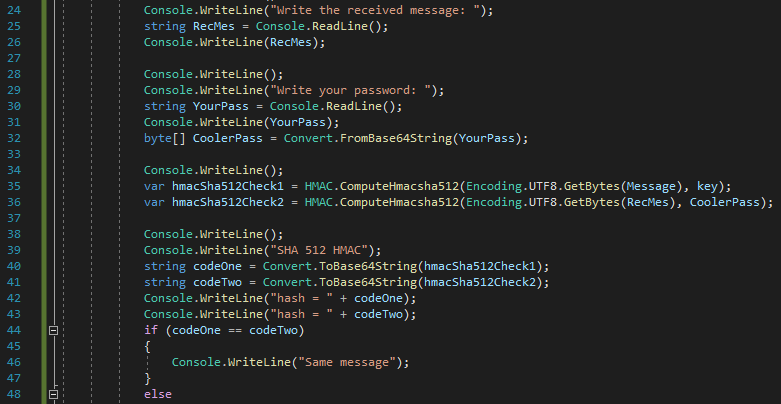
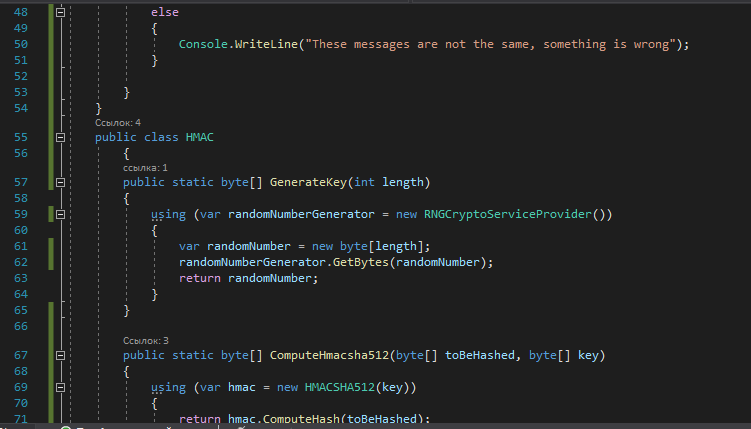
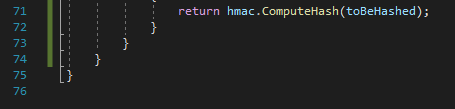
Результати роботи:  
  


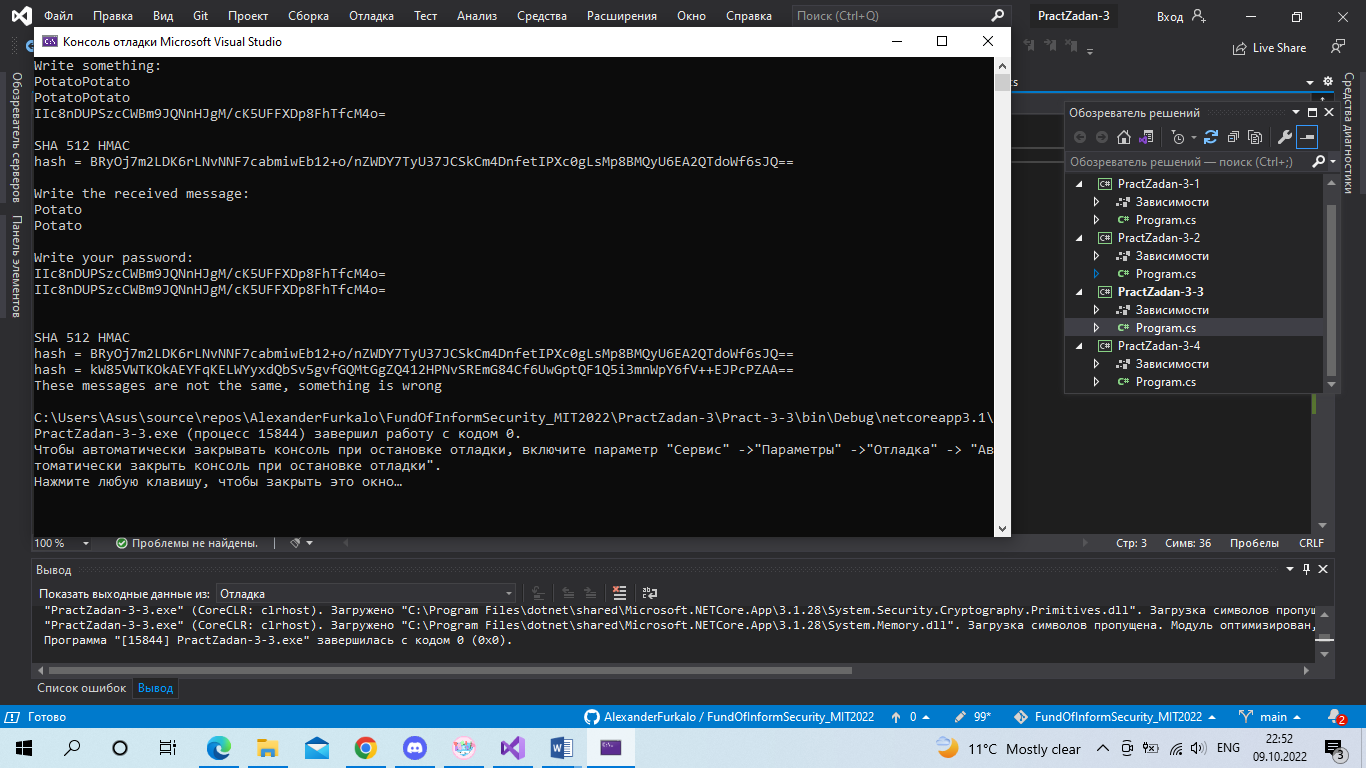
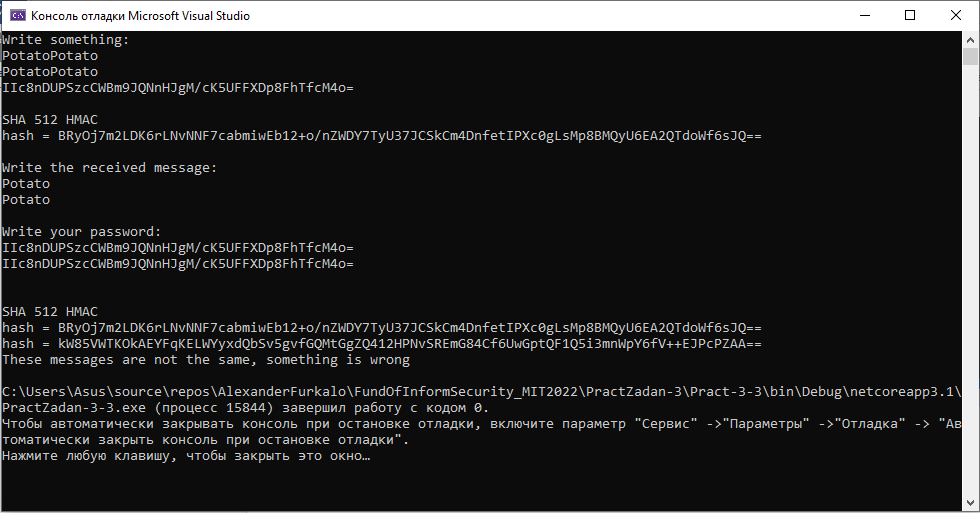
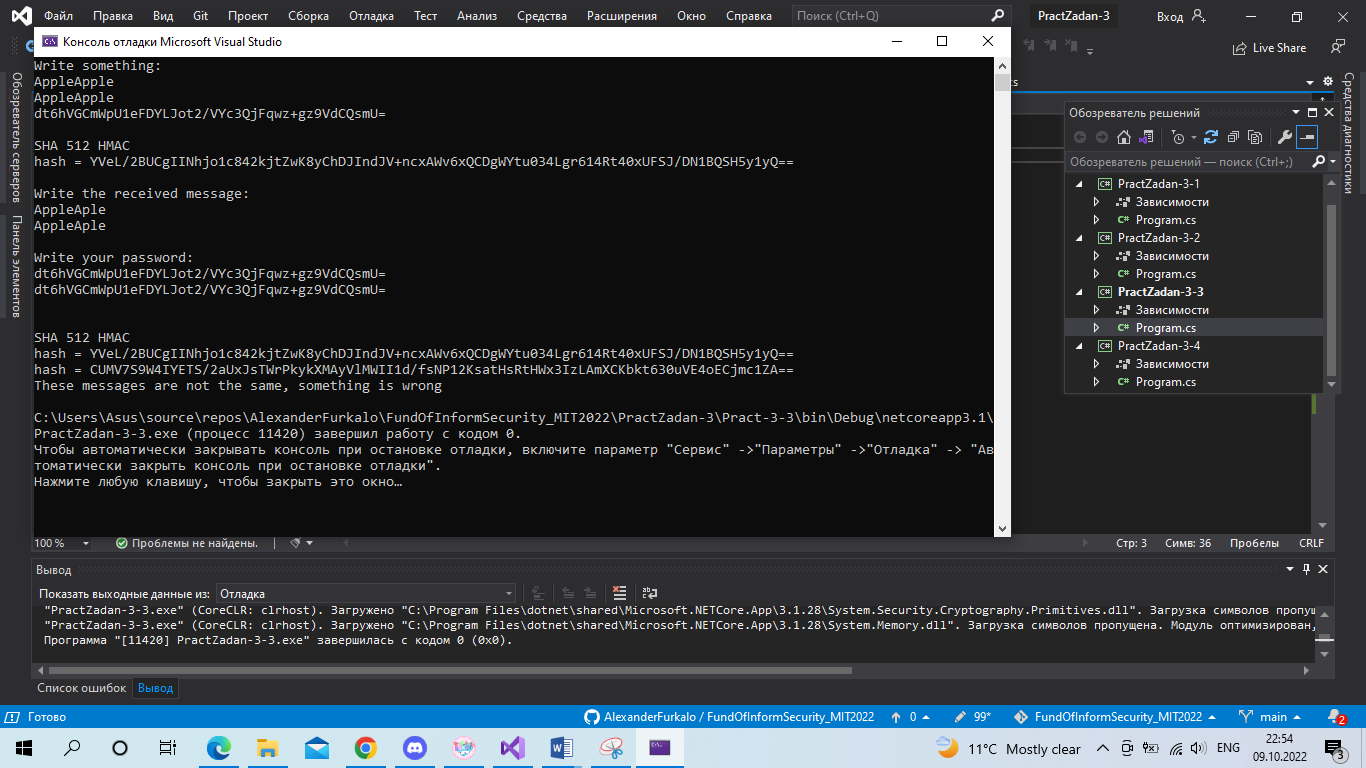
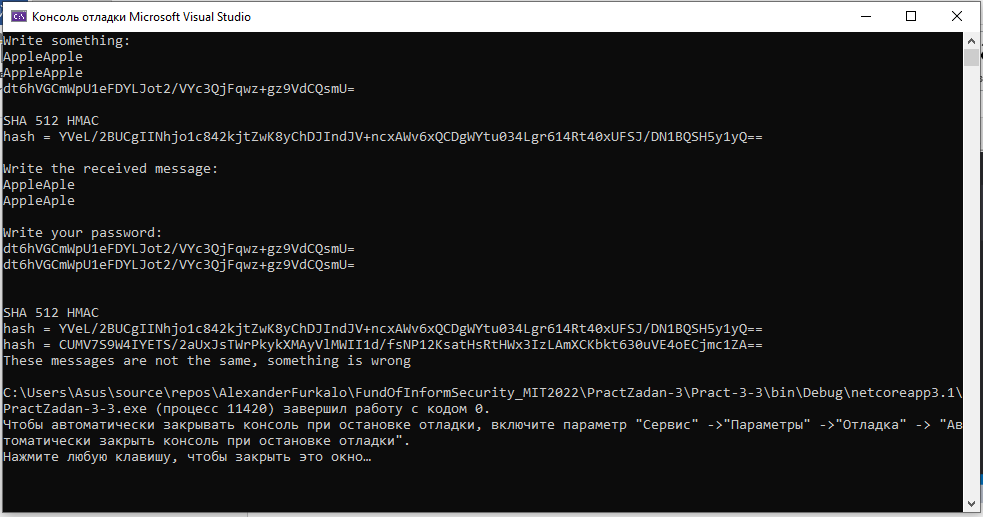
Хеш-функція (або хешування) - перетворення (однобічна функція) бітового рядка довільної довжини в бітовий рядок (блок) фіксованої довжини (як правило, 128-512 біт - хеш-код, також називають хешем або дайджестом повідомлення) з такими властивостями, як:  
1) хеш-код повинен легко обчислюватися для довільного вхідного повідомлення;  
2) незначна зміна вхідного повідомлення має призводити до тотальної зміни значення хеш-функції;  
3) Не повинно бути можливості за прийнятний час знайти повідомлення, що має заданий хеш-код (має бути стійкою в розумінні обернення);  
4) Хеш-функція має бути стійкою в розумінні виявлення колізій першого (для заданого повідомлення m повинно бути обчислювально неможливо підібрати інше повідомлення n , для якого h(n) = h(m) ) та другого (повинно бути обчислювально неможливо підібрати пару повідомлень, що мають однаковий хеш) роду.  
Хешування застосовується для порівняння: якщо у двох масивів даних хеш-коди різні, масиви гарантовано розрізняються; якщо однакові – масиви, швидше за все, однакові.

У цьому завданні ми порівняли хеш-коди за всіма відомими алгоритмами: MD5, Sha5, Sha256, Sha384, Sha512. Кількість символів у підсумковому хеш-коді збільшується, де MD5 має найменше символів, а Sha512 - найбільше. Різні тексти мають різні хеш-коди.  
MD5 не рекомендується для нових проектів, оскільки не стійкий до колізій, але швидкий у роботі. Альтернативою алгоритму MD5 є сімейство алгоритмів Secure Hash Algorithm (SHA). На вхід SHA алгоритму подаються деякі дані, проводяться обчислення, та повертається хеш-код. Концепція така ж, як і у MD5 алгоритму, але алгоритми суттєво відрізняються. SHA хеш-коди більш довші та більш стійкі. Який саме із 4-х алгоритмів SHA використовувати залежить від преференцій швидкодії, обсягу пам’яті для зберігання хеш-коду та стійкості до колізій.

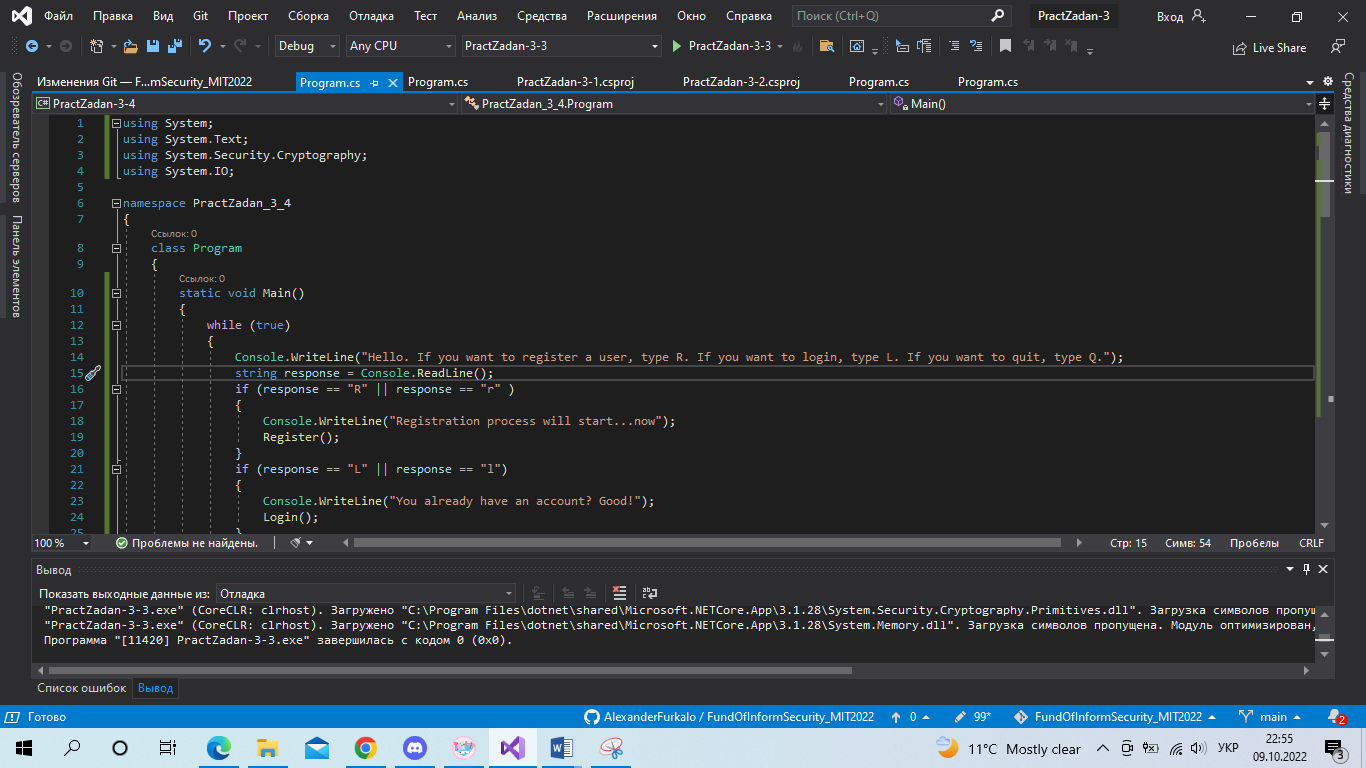
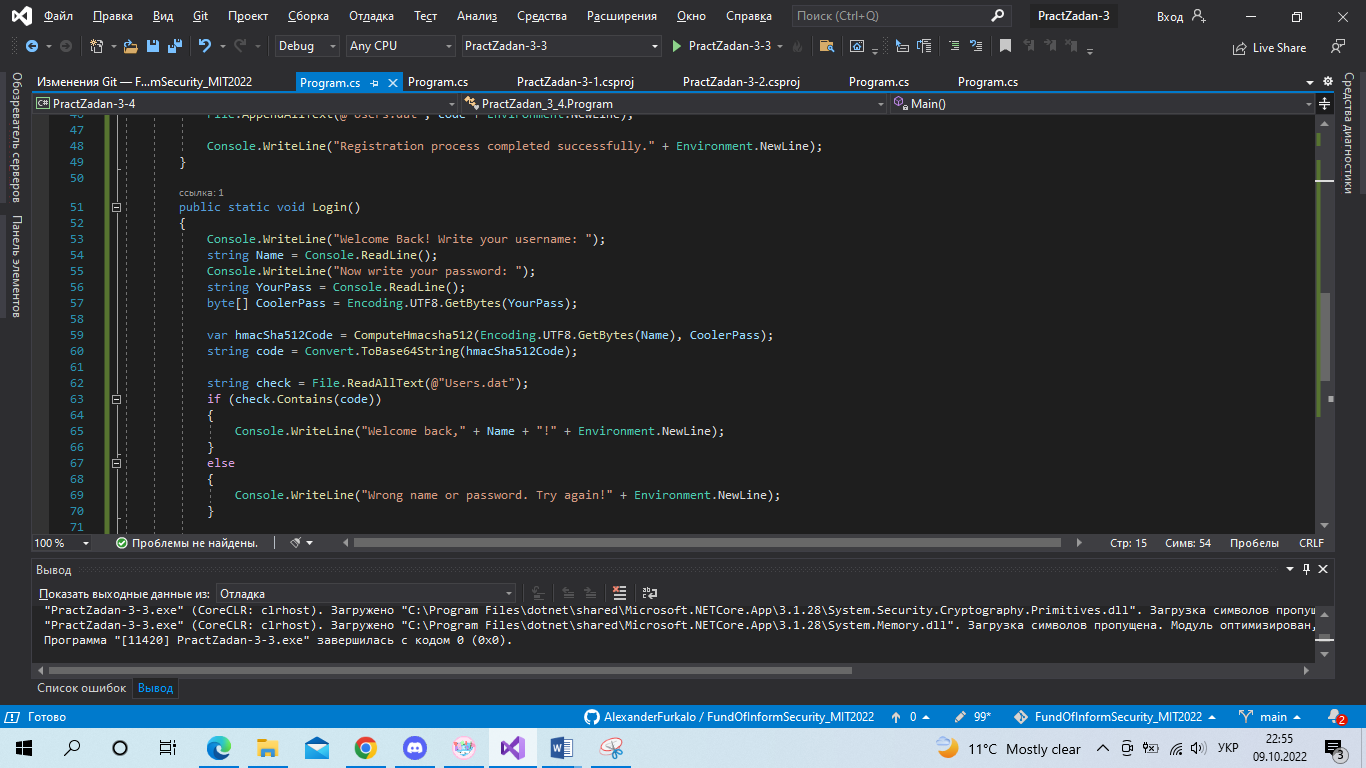
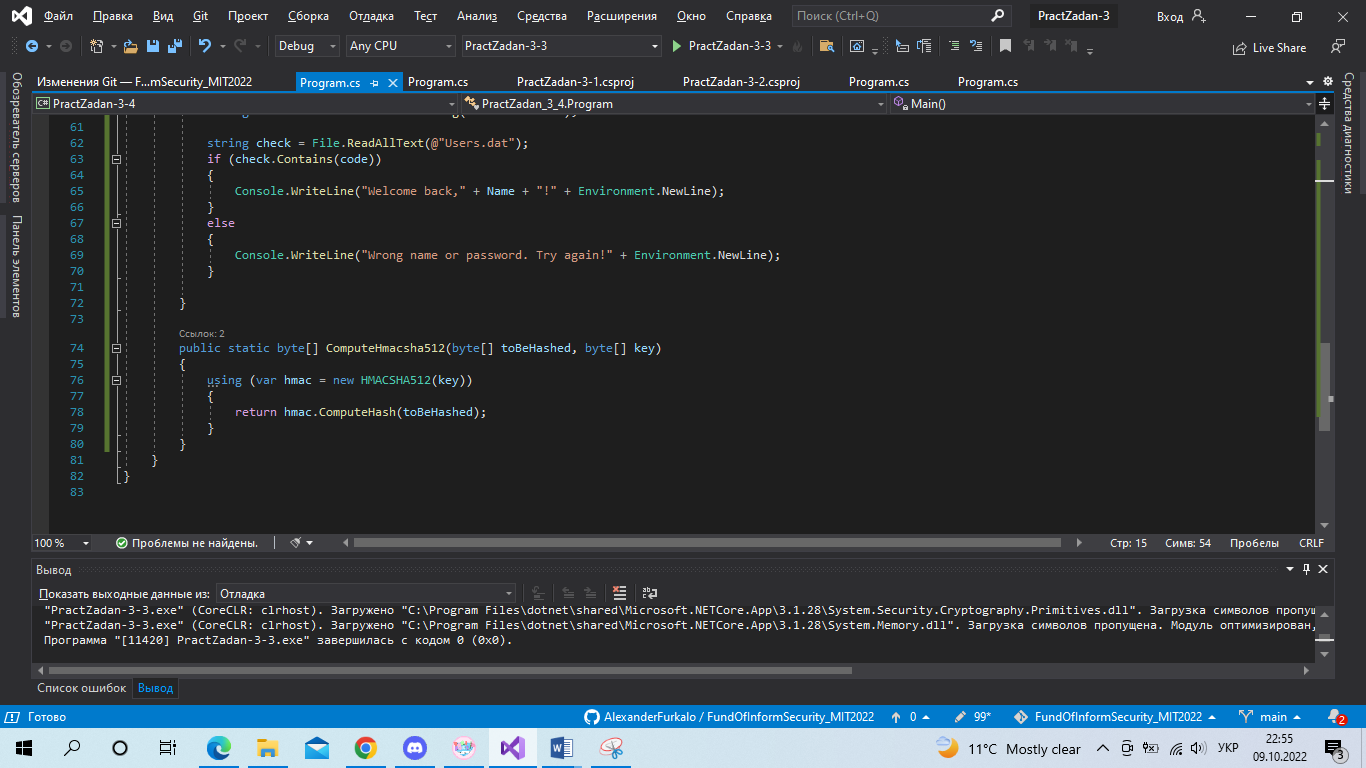
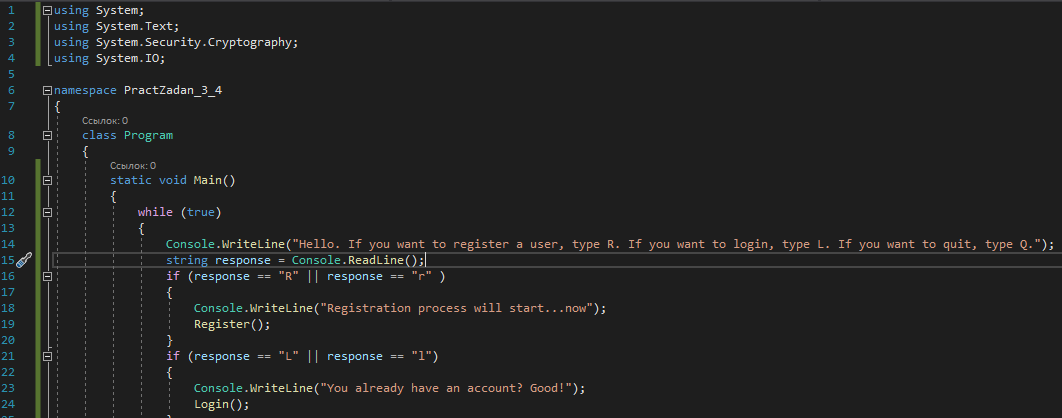
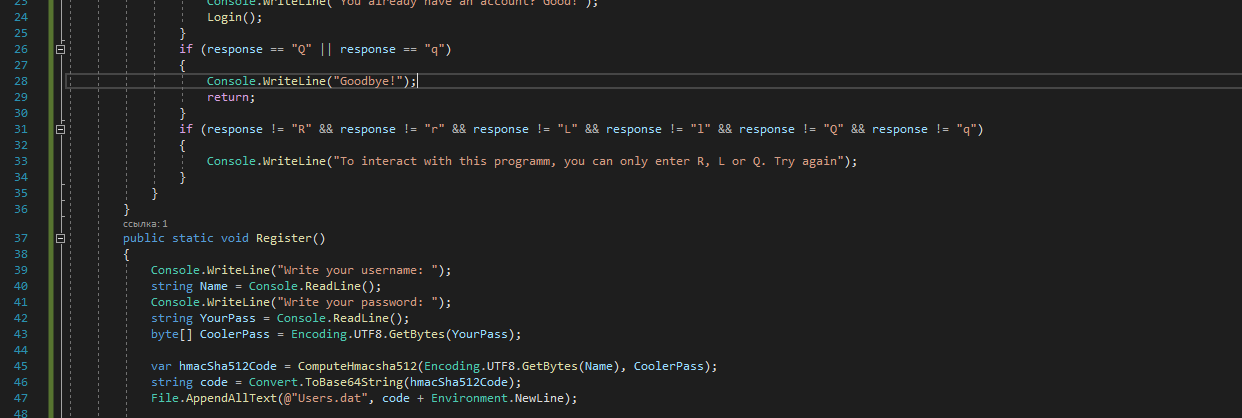
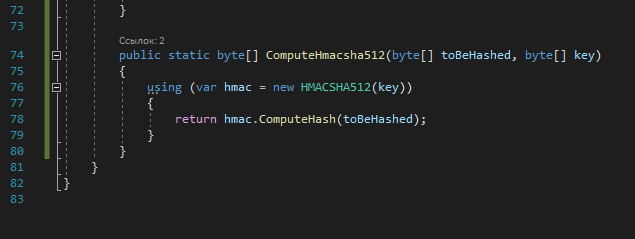
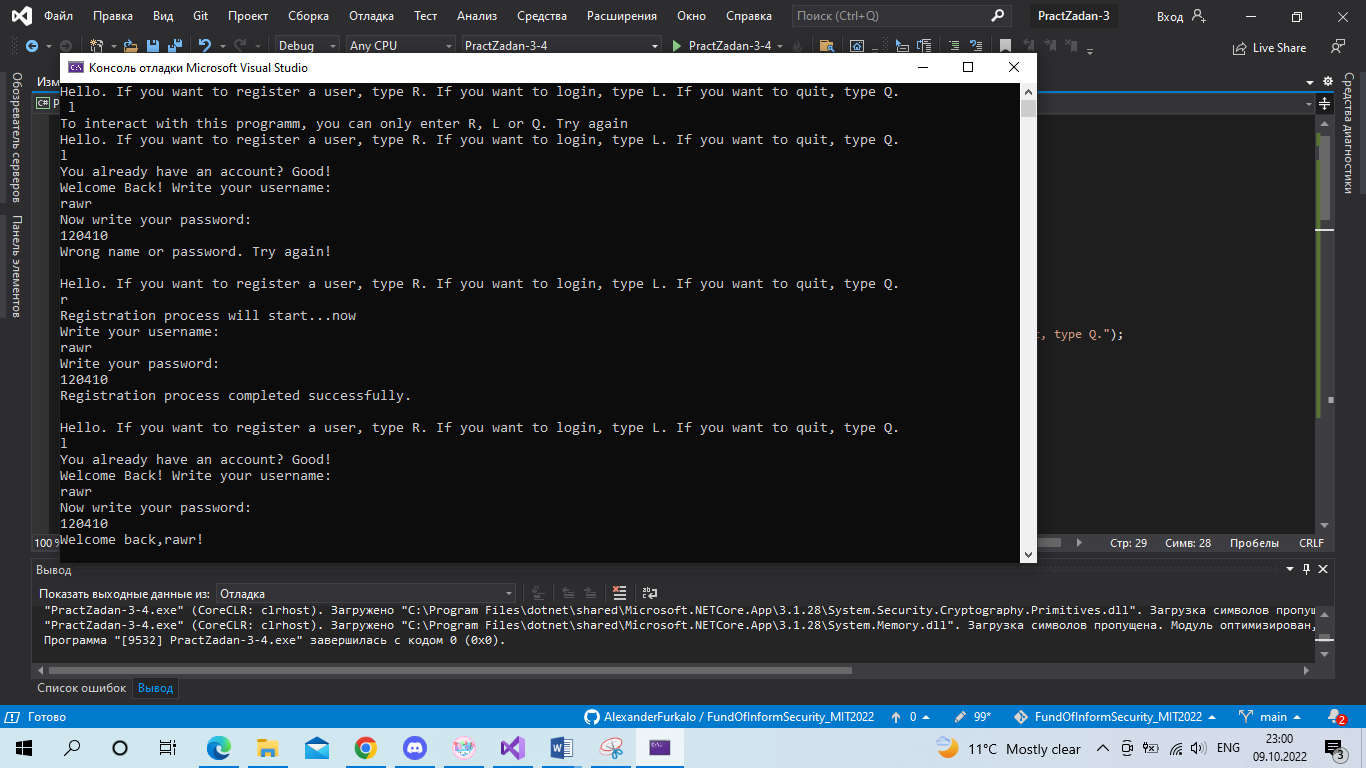
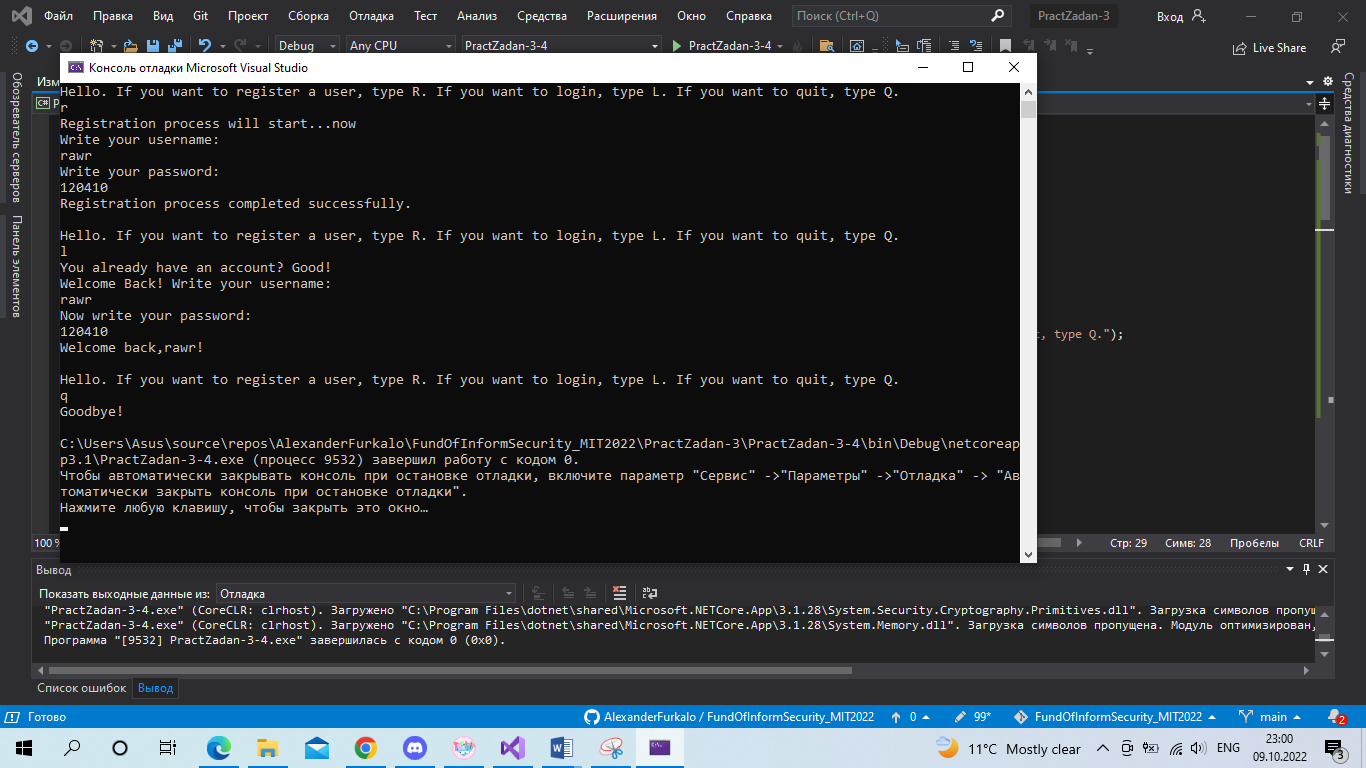
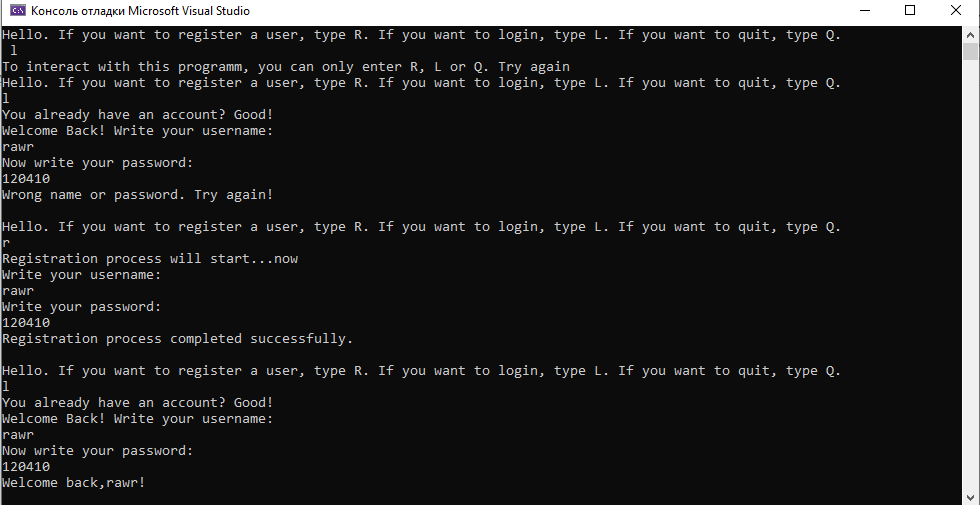
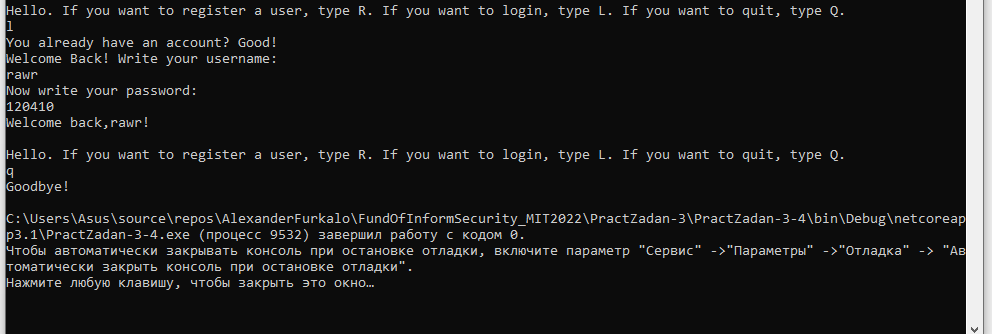
Друге завдання:  
  
  
  
  
Наблизивши:  
  
  
  
  
Результати виконання:  
  


Код цієї програми особливо не відрізняється від попередньої варіації у другій лабораторній, але на цей раз ми шукаємо хеш-код. Лише за приблизно 10 хвилин ми змогли перебрати всі варіанти пароля на 8 знаків із цифр – цей пароль 20192020. Він є простою комбінацією цифрових значень двох років, і, крім того, складається лише з цифр, так що це вкрай ненадійний пароль і знаходиться досить легко.  
Так як криптостійкість відсутня, існує можливість легко підібрати повідомлення на заздалегідь відому контрольну суму. Саме тому такі швидкі і прості алгоритми використовуються тільки для захисту від ненавмисного перекручування інформації (про це йдеться у третьому завданні).   
Не доведено існування повністю криптостійких (необоротних) хеш-функцій, для яких обчислення якогось прообразу заданого значення хеш-функції теоретично неможливе. Зазвичай знаходження зворотного значення є лише обчислювально складним завданням.

Третє завдання:  
  
  
  
  
Наблизивши:  
  
  
  


Результати виконання:  
  
  
  


У цьому завданні нам потрібно було створити програму, яка б обчислювала хеш-код автентифікації повідомлення, та потім мала можливість порівняти цей код з хеш-кодом автентифікації іншого повідомлення. Якщо повідомлення те саме, виводиться “Same Message”. Якщо ні – “These messages are not the same, something is wrong”. Щоб “надіслати” повідомлення, досить щось написати, а потім функція сама згенерує пароль через GenerateKey. Щоб перевірити, потрібно спочатку ввести якесь повідомлення, а потім повторити пароль.  
HMAC (Hash Message Authentication Code) – це комбінація односторонньої функції хешування із криптографічним ключем. HMAC аналогічно до звичайних хеш-функцій повертає хеш-код повідомлення за винятком того, що секретний ключ є частиною самого повідомлення. HMAC дозволяє провести автентифікацію повідомлення тому, що лише особа, яка знає секретний ключ, може згенерувати такий хеш-код.

Четверте завдання:  
  
  
  
  
  
Трохи наблизивши:  
  
  
  
  
Результати:  
  
  
  
  
Нам було потрібно написати програму для реєстрації користувача за логіном/паролем та авторизації шляхом співставлення відповідних логінів і паролів. Програма дає можливість вибору між трьома опціями - реєстрацією користувача, авторизації та виходом із програми. Під час реєстрації користувачеві потрібно придумати свій особистий логін і пароль - потім ці дані використовуються для генерації HMAC, і в такому вигляді записуються у файл. Процес авторизації відрізняється тим, що замість записування цього значення воно потім шукається у файлі, куди вони повинні записуватися - якщо подібностей немає, користувачу повідомляють про помилку і повертає в меню; в іншому випадку система його вітає.

Висновки

Під час цієї практичної роботи ми навчилися писати програми, які обчислюють хеш-коди за всіма відомими алгоритмами для заданих даних, та порівняли розміри хеш-кодів та значення для однакових та різних даних; відновили пароль користувача, використовуючи MD5 хеш-код, та зробили висновки про надійність такого пароля; написали програму для обчислення хеш-коду автентифікації повідомлення а також реалізували можливість перевірки автентичності повідомлення; написали програму для реєстрації користувача за логіном/паролем та авторизації шляхом співставлення відповідних логінів і паролів.  
Ми дізналися, що таке хеш-код, його властивості і як він може використовуватися (наприклад, як механізм перевірки цілісності даних або для зберігання паролів у базі даних).