**ЛАБОРАТОРНА РОБОТА № 4**

з дисципліни

«Архітектура комп`ютерів»

Шейко Р. О.

Групи КІ-21-2

**Завдання на лабораторну роботу**

1. Розробити програму, за допомогою якої з заданого рядка отримати рядок згідно варіанту (табл. 4.8). Результат слід вирівняти по правому краю.

Зауваження. Загальна методика вставки бітових рядків виглядає наступним чином: - підігнати до правого краю рядка молодший біт місця вставки в цьому рядку. Для цього використовувати команди циклічного зсуву, щоб зберегти праву частину заданого рядка. Величина зсуву − це номер початкової позиції місця вставки; - зсунути вихідний рядок вправо на кількість бітів, яка дорівнює довжині підрядка, що вставляється; - командою shld вставити підрядок, що вставляється у вихідний підрядок. Перед цим, безумовно, лівий край підрядка, який вставляється, повинен знаходиться біля лівого краю регістру RАХ; - відновити командою циклічного зсуву праву частину заданого рядка.

2. Розробити програму, за допомогою якої з заданого рядка отримати рядок згідно варіанту (табл. 4.8). Бітовий рядок, що вставляється, повинен бути вирівняним до лівого краю.

3. Розробити програму, за допомогою якої з двох рядків отримати один, заданий варіантом з табл. 4.8, взяв за основу раніше розроблені програми.



Рис 1. Варіант завдання

2. Код до програми першого завдання знаходиться у додатку А, другого – Б, а третє у В.

3. Почнемо тестування. Почнемо з першої програми:

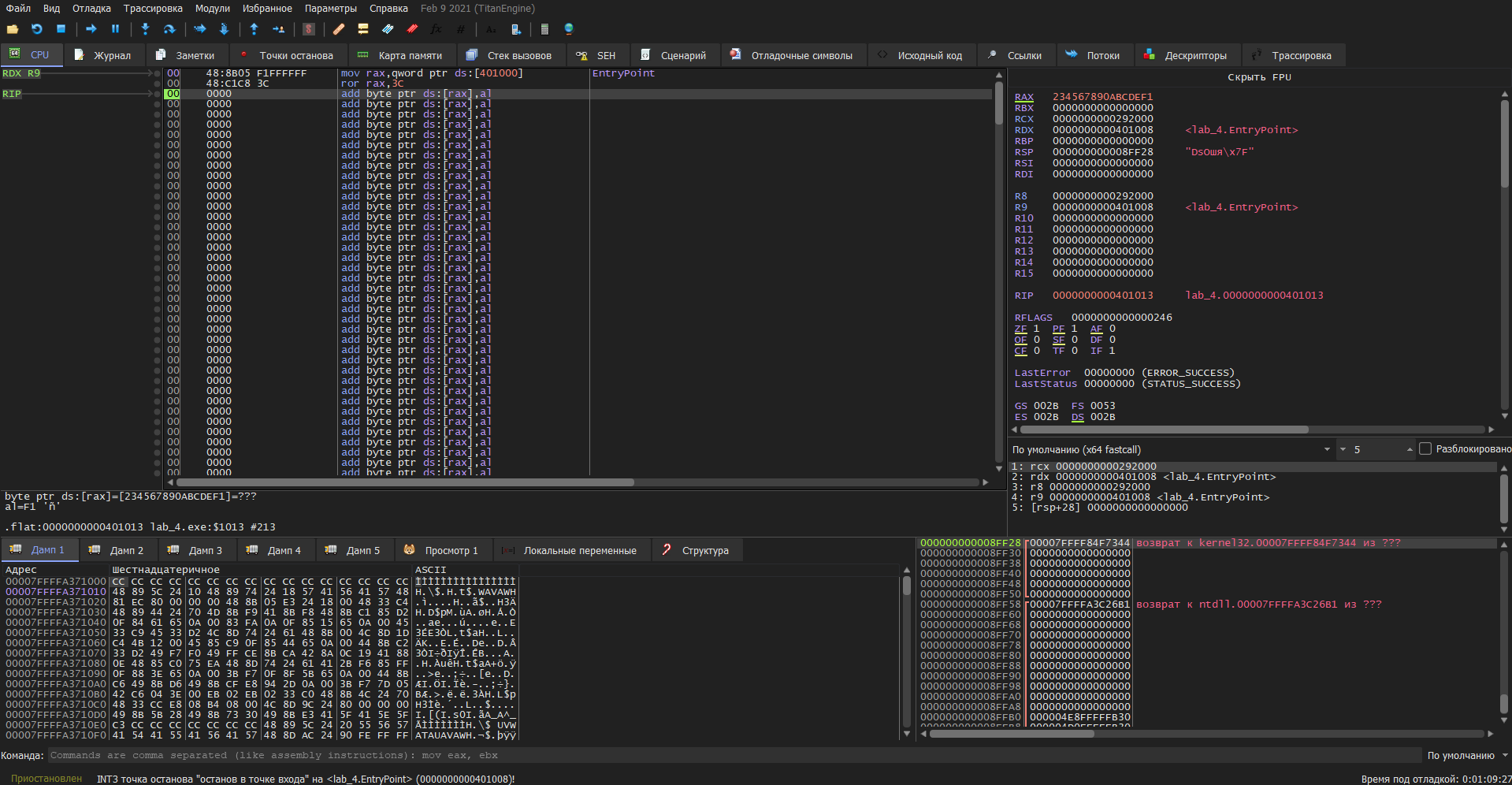
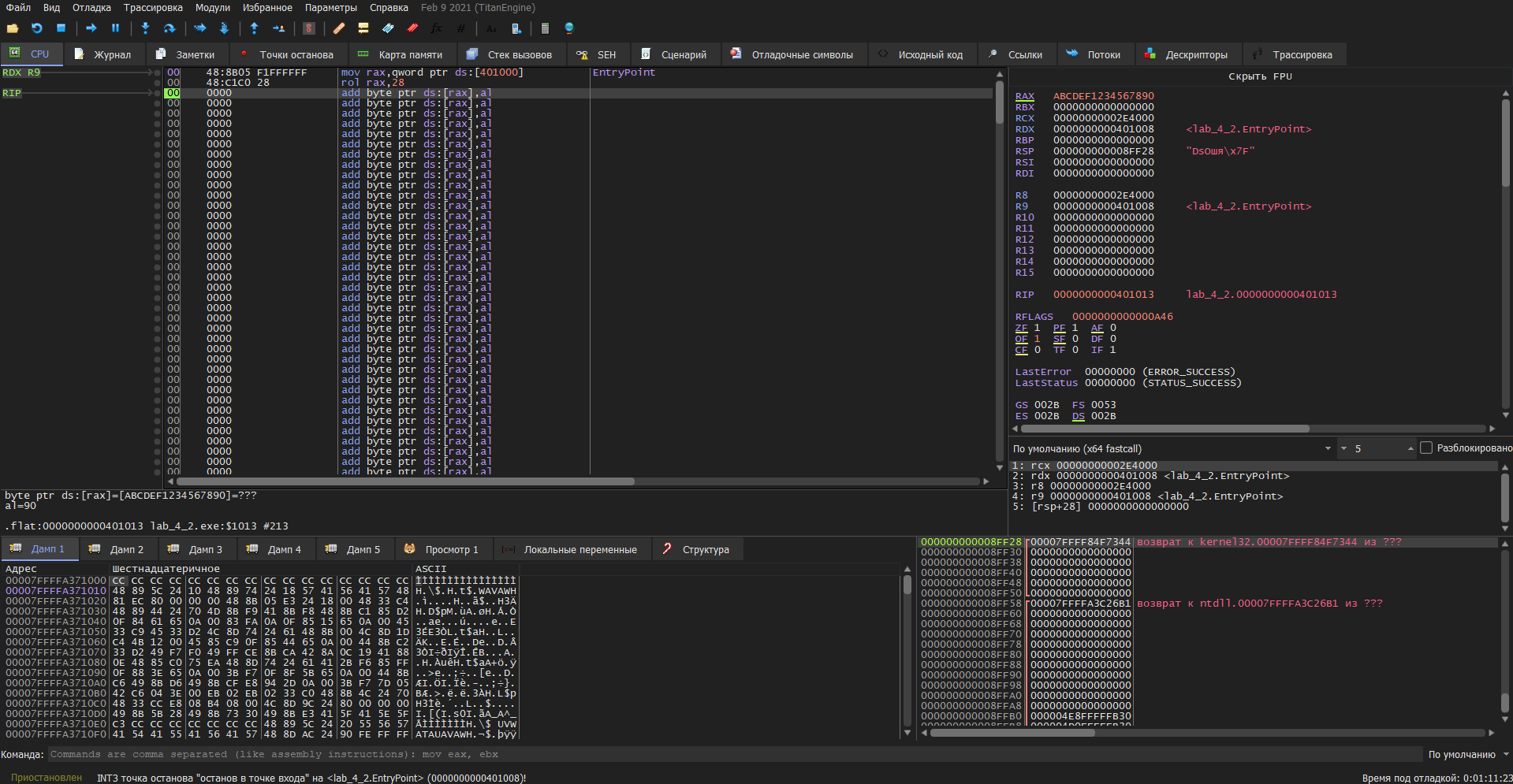


Рис 2. Перше завдання після завершення коду

  
Рис 3. Друге завдання після завершення коду

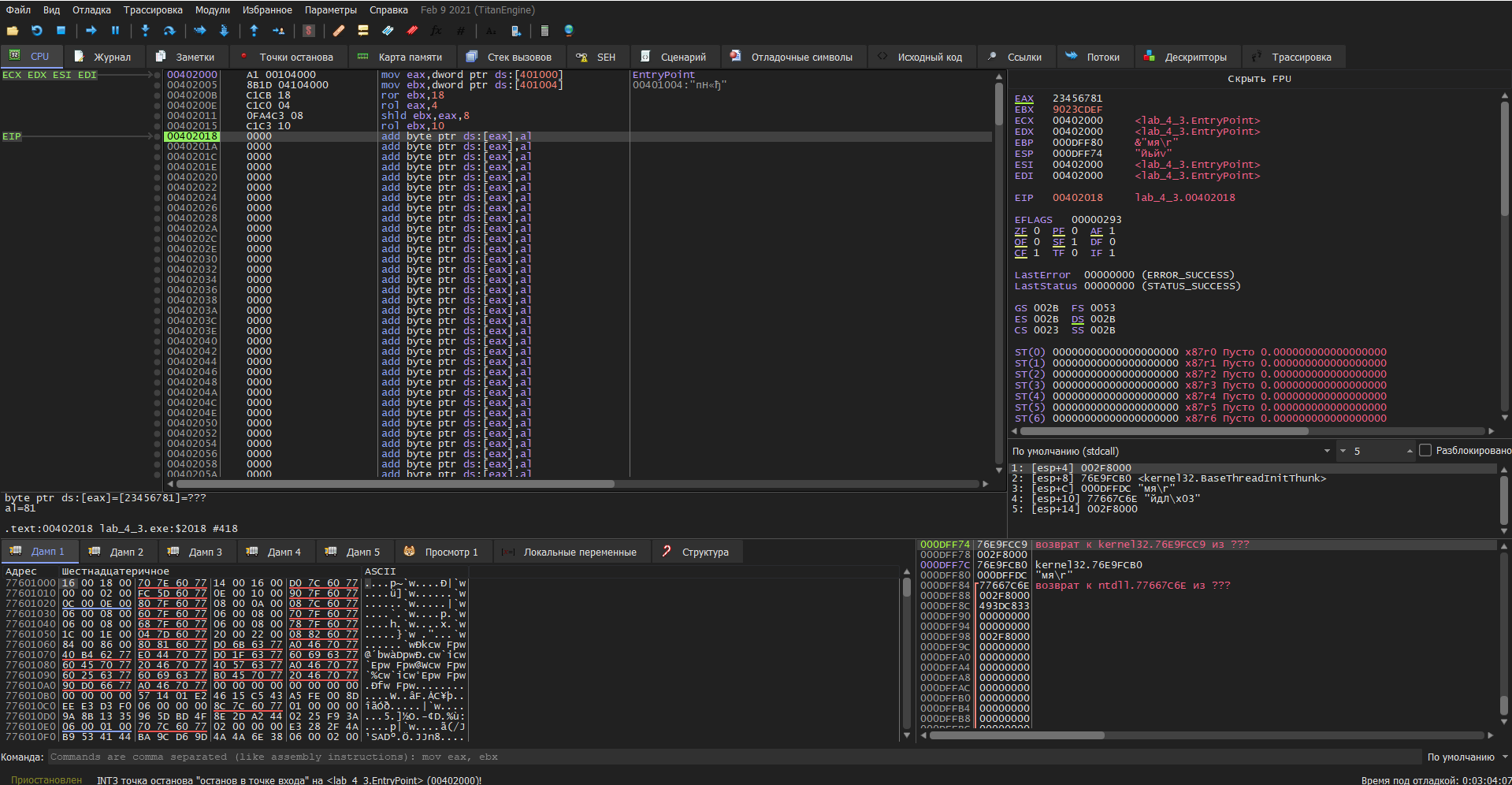


Рис 4. Третє завдання після завершення коду

Додаток А

*Лістинг завдання 1*

*format PE64 console*

*entry start*

*bit\_str dq 1234567890abcdefh*

*start:*

*mov rax,[bit\_str]*

*ror rax,60 ;*

Додаток Б

*Лістинг завдання 2*

*format PE64 console*

*entry start*

*bit\_str dq 1234567890abcdefh*

*start:*

*mov rax,[bit\_str]*

*rol rax,40*

Додаток В

*Лістинг завдання 3*

*format PE console*

*entry start*

*section '.data' data writable*

*bit\_str1 dd 12345678h*

*bit\_str2 dd 90abcdefh*

*section '.text' code executable*

*start:*

*;9023cdef*

*mov eax,[bit\_str1] ; 12345678*

*mov ebx,[bit\_str2] ; 90abcdef*

*ror ebx,24 ; bcdef90a*

*rol eax,4 ; 67812345*

*shld ebx,eax,8 ; ef90a678*

*rol ebx,16 ; 90a678ef*