**ЛАБОРАТОРНА РОБОТА № 6**

з дисципліни

«Архітектура комп`ютерів»

Шейко Р. О.

Групи КІ-21-2

**Завдання на лабораторну роботу**

1. Дослідити програму, наведену на лістингу 4.15 в наступній послідовності:

- ввести програму і отримати їй відповідний машинний код;

- виконати до першої команди коду користувача (inc rbx);

- встановити точку останову на наступній команді після останньої команди коду користувача (jz start);

- здійснити багаторазовий запуск програми і переконатися в тому, що налагоджувач реагує (припиняє виконання програми) лише на ті числа в регістрі RBX, які містять «1» у п'ятому розряді;

- проконтролювати наявність «0» в розряді ZF регістра EFLAGS/FLAGS після припинення виконання програми;

- дослідити виконання програми у режимі трасування;

- змінити програму таким чином, щоб налагоджувач припиняв виконання програми при наявності «0» у п'ятому розряді й «1» у всіх інших розрядах досліджуваного числа

- дослідити виконання модифікованої програми в режимі трасування.

2. Розробити і дослідити програму для запису одиниць в усі розряди регістра RСX, якщо число, сформоване в регістрі RВX більше 3, але менше 8.

- ввести програму і отримати відповідний їй машинний код;

- виконати до кода користувача (Кнопка , або вибір пунктів меню Отладка → Выполнить до пользовательского кода, або натискання клавіш та );

- встановити точку зупинки на наступній команді після останньої команди користувача і здійснити багаторазовий запуск програми;

- ознайомитися зі зміною вмісту регістрів RВX і RСX;

- дослідити виконання програми в режимі трасування.

3. Дослідити програму , наведену в лістингу 4.16 в наступній послідовності:

- ввести програму і отримати відповідний їй машинний код;

- дослідити виконання програми без використання налагоджувача.

4. Скласти програму у мнемокодах для визначення за допомогою циклічного обчислювального процесу кількості двійкових чисел, які мають задані значення (k), (n) і (m) двійкових розрядів із номерами k, n і m з масиву, розміщеного в пам'яті відповідно до індивідуального варіанту N (табл. 4.10). 

Рис 1. Варіант завдання

2. Код знаходиться у додатку А.

3. Почнемо тестування. Проєкти були протестовані. Що були надані в прикладах. Ось вирішення індивідуального завдання:

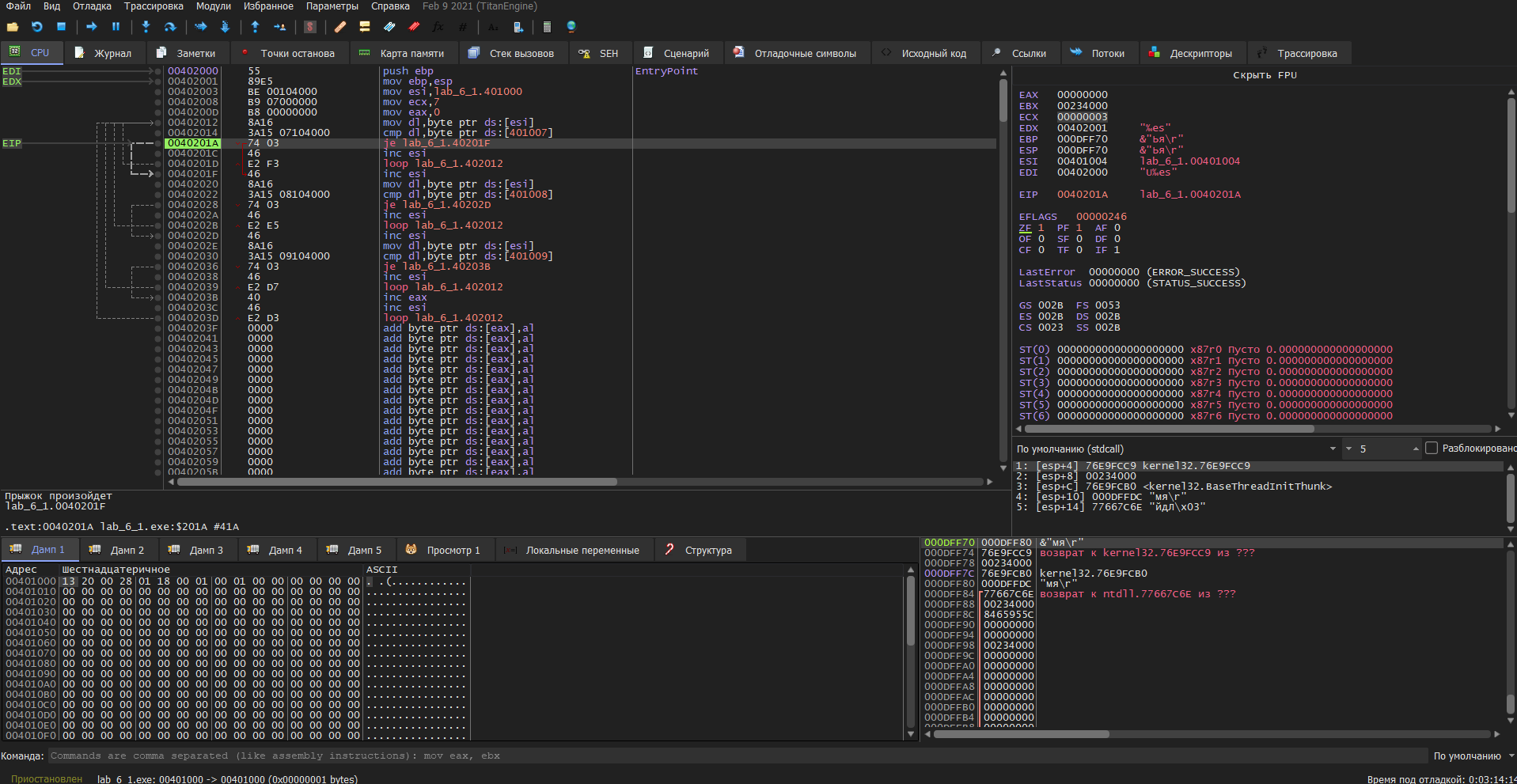


Рис 2. Початок відлагодження програми

Рис 3. Кінець відлагодження програми

Додаток А

*Лістинг завдання*

*format PE64 console*

*entry start*

*section '.data' data writable*

*size = 8*

*source dq 0000010000000000h, 0101000100000000h, 0100010100010101h,0101010101000101h, 0100000101010001h, 0001000101000101h,0101010100000000h, 0000000001010101h, 0000000100000000h,0001010001000100h, 0101010000000000h*

*number dq (number-source)/size ;кількість даних в source*

*mask1 dq 0000000100000000h ; маска 32 розряду*

*mask2 dq 0000000001000000h ; маска 40 розряду*

*mask3 dq 0000000000010000h ; маска 24 розряду*

*section '.text' code executable*

*start:*

*xor rbx,rbx ;Лічильник даних, які зазнали маскування = 0*

*mov rdx,source ;В RDX адреса, з якої починається масив даних*

*mov rcx,[number] ;Завантаження лічильника циклів*

*m1:*

*mov rax,[rdx] ;Завантаження акумулятора з адреси в RDX*

*test rax,[mask1] ;Тестування 48-го розряду*

*jnz m2 ;Перехід якщо у 48-му розряді немає 0*

*test rax,[mask2] ;Тестування 32-го розряду*

*jz m2 ;Перехід на М2, якщо ZF = 1 (32-й розряд = 0)*

*test rax,[mask3] ;Тестування 16-го розряду*

*jnz m2 ;Перехід якщо у 16-му розряді немає 0*

*inc rbx ;Інкремент лічильника даних*

*m2:*

*add rdx,size ;Перехід до наступної адреси пам'яті даних*

*dec rcx ;Зменшення на 1 вмісту лічильника циклів*

*jnz m1 ;Перехід на М1, якщо ZF = 0 (не усі дані)*