# Міністерство освіти та науки України

Національний технічний університет України

“Київський політехнічний інститут ім. Ігоря Сікорського” Факультет інформатики та обчислювальної техніки

**Лабораторна робота №6**

# з дисципліни “Технології програмування для комп'ютерних систем – 3”

Виконала:

студент групи ІВ-91мн Коренко Д. В.

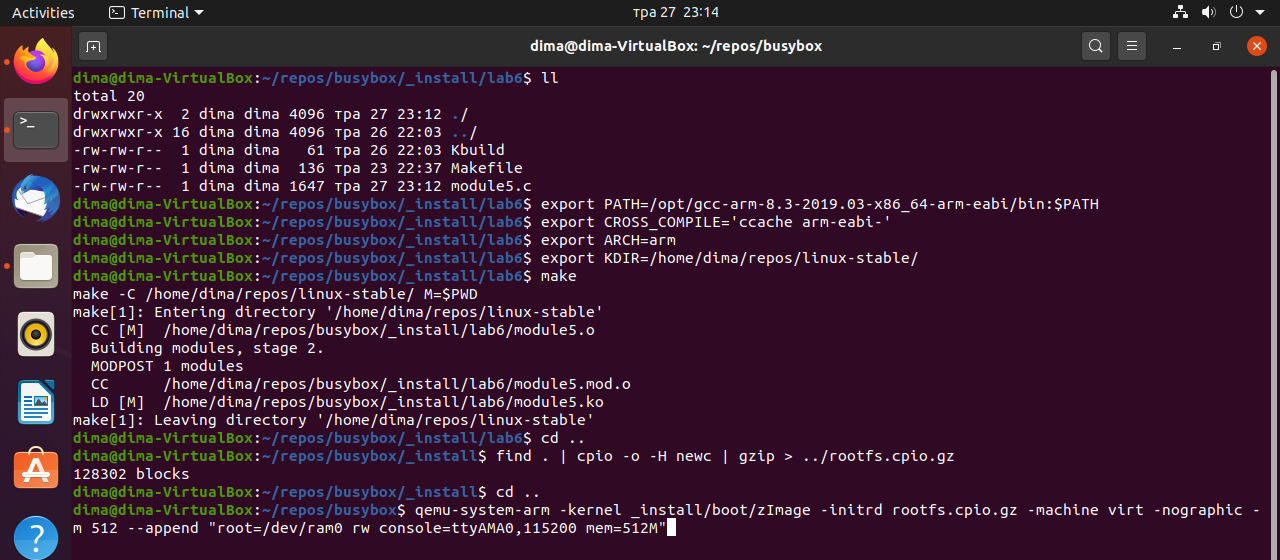
Київ 2020 р.

## Завдання

Ознайомитися із дебагінгом модулю.

## Послідовність виконання роботи

1. Створюємо модуль. Для цього у файловій системі створюємо директорію lab6, у яку додаємо усі необхідні файли для роботи з модулем. Виконуємо команди export для збирання ядра, а також export KDIR=/home/dima/repos/linux-stable/. Виконуємо команду make для збирання модулю. Створимо архів CPIO для rootfs та заархівуємо його за допомогою GZip:



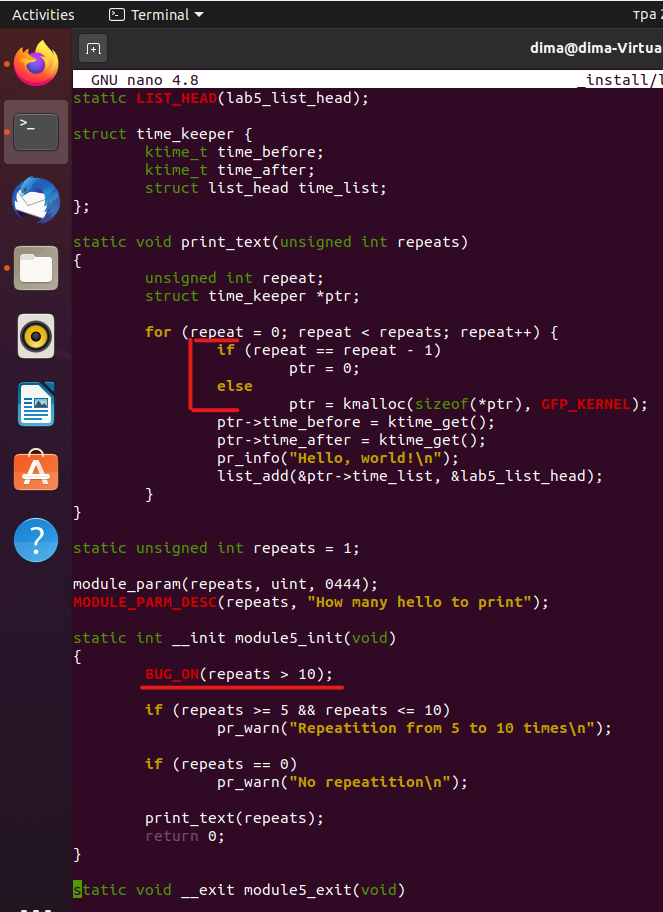
1. Виконання завдання Basic1:

Замінимо виведення повідомлення та повернення -EINVAL для неприпустимого значення параметра викликом функції BUG\_ON().

Додаємо навмисне внесення помилки під час формування останнього елементу списку. До Kbuild додаємо прапорець -g:

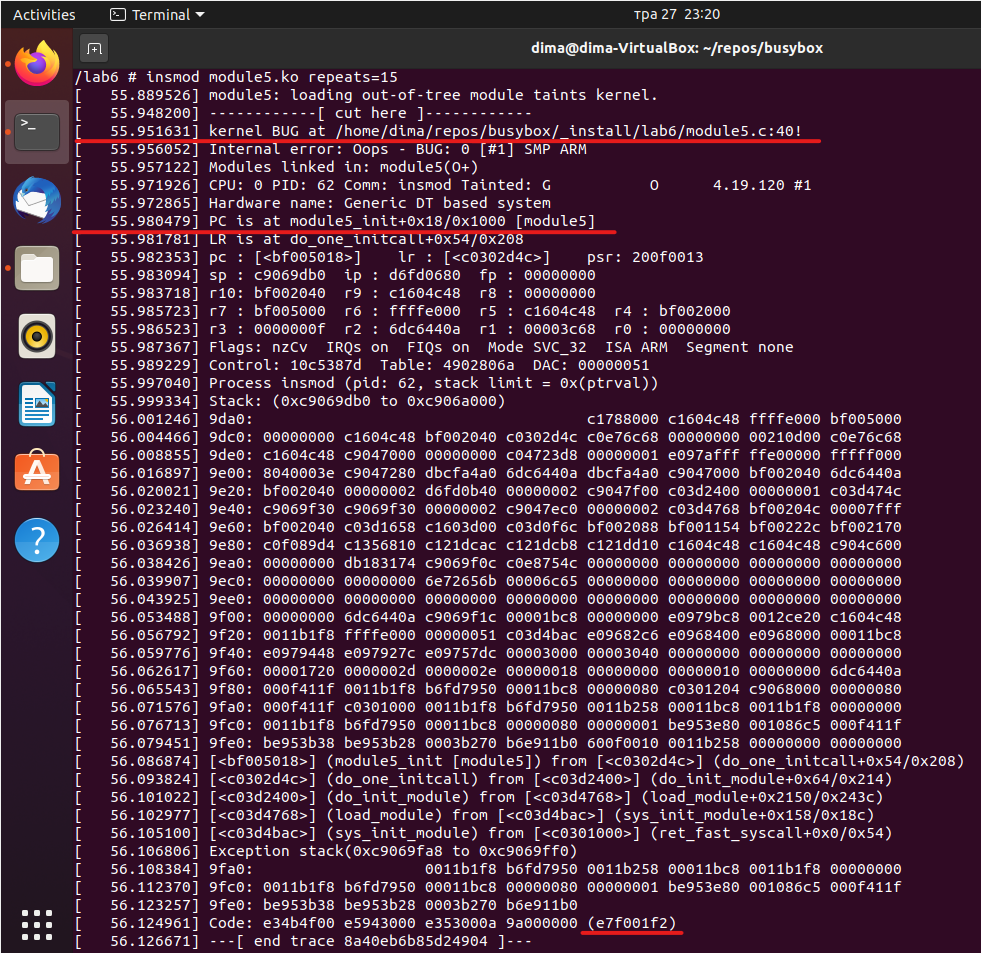


Нижче наведено змінений фрагмент коду:

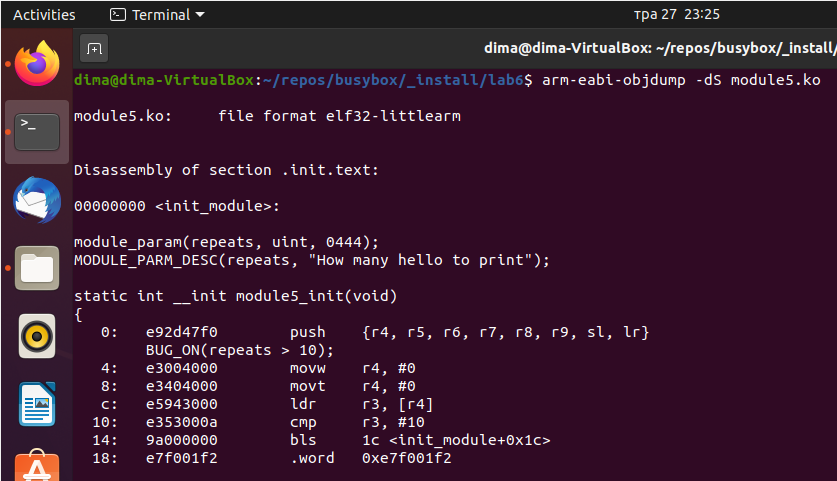


1. Протестуємо роботу модуля:

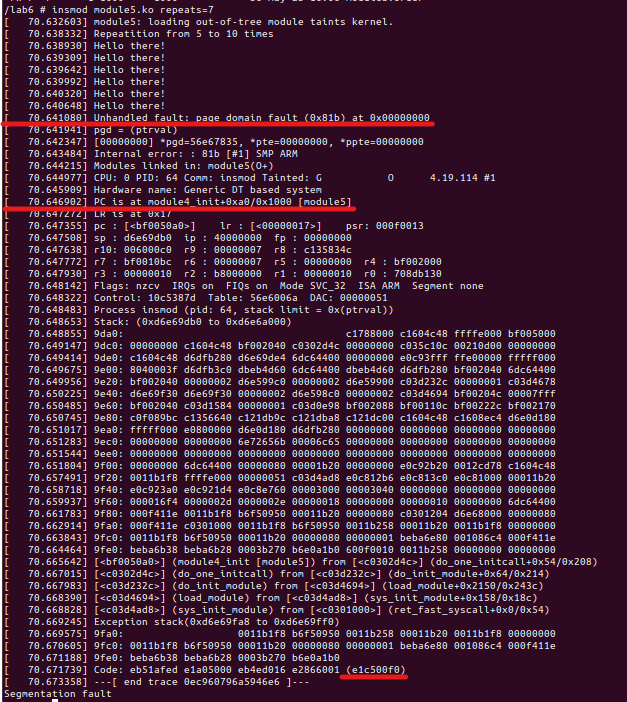
Можна побачити, що завантаження модулю зі значенням параметру, який є більшим за 10 (у прикладі repeats = 12), призводить до виконання макросу BUG\_ON.



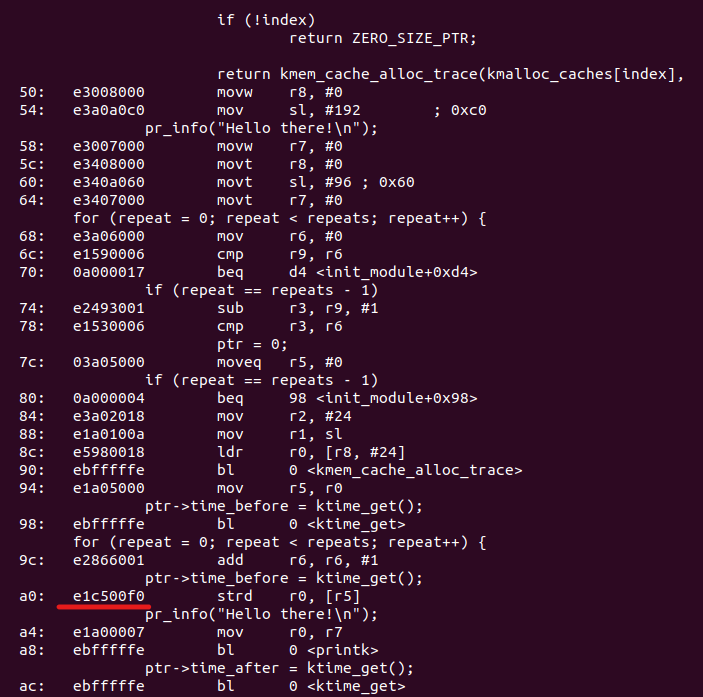
За допомогою утиліти objdump можна побачити, що значення PC та рядку з BUG\_ON є ідентичними.



Якщо ж вести значення параметру менше за 10, то при завантаженні модуля (з параметром, наприклад, repeats=7), побачимо null pointer dereference.



Як результат виконання objdump отримали таке:



1. Виконання завдання Basic2:

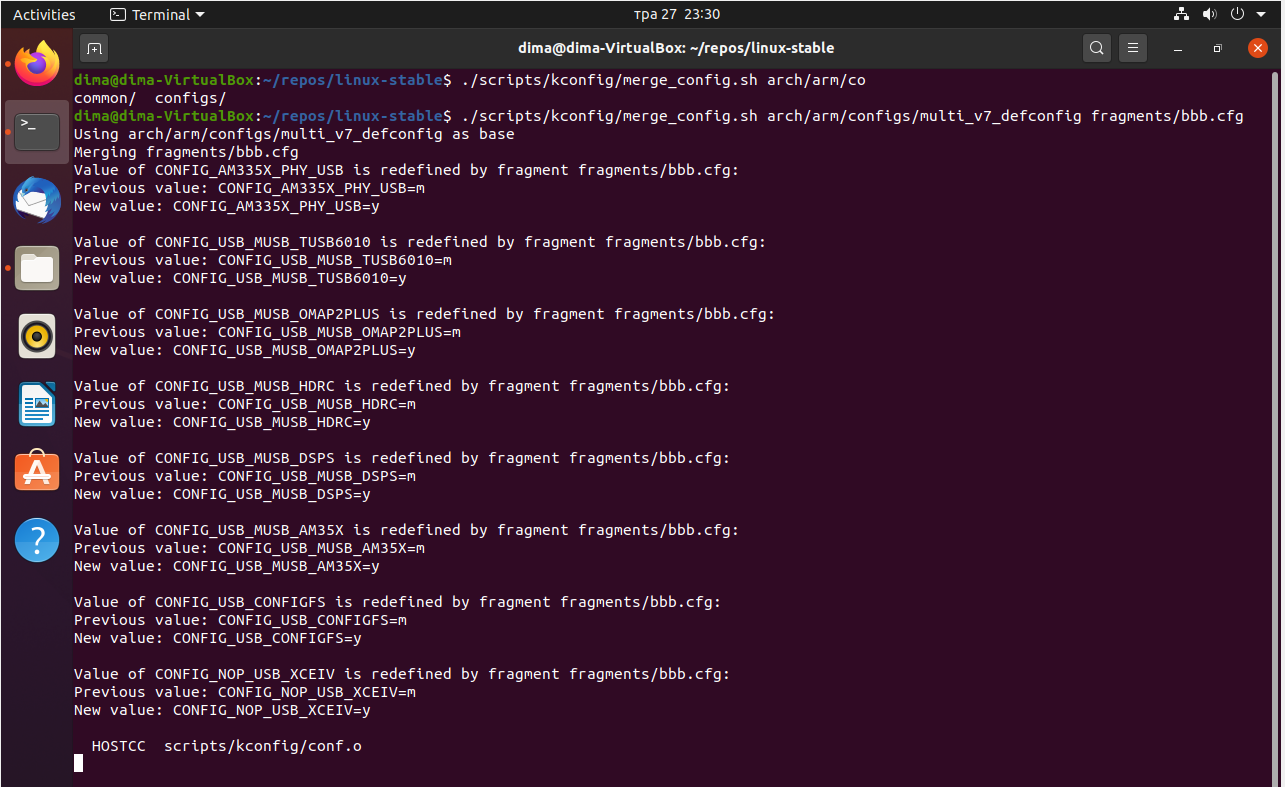
У функції exit модуля друк вмісту списку змінемо на pr\_debug і додамо два виклики

pr\_debug до та після друку списку.

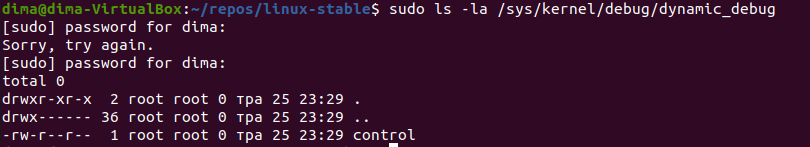
Для того, аби побачити зміни при завантаженні та вивантаженні модуля необхідно додати #define DEBUG на початку файлу та аби рівень логування був 8, аби виводилися debug повідомлення.

Встановлюємо параметр CONFIG\_DYNAMIC\_DEBUG у ~/repos/linux-stable/fragments/bbb.cfg та перезбираємо ядро.

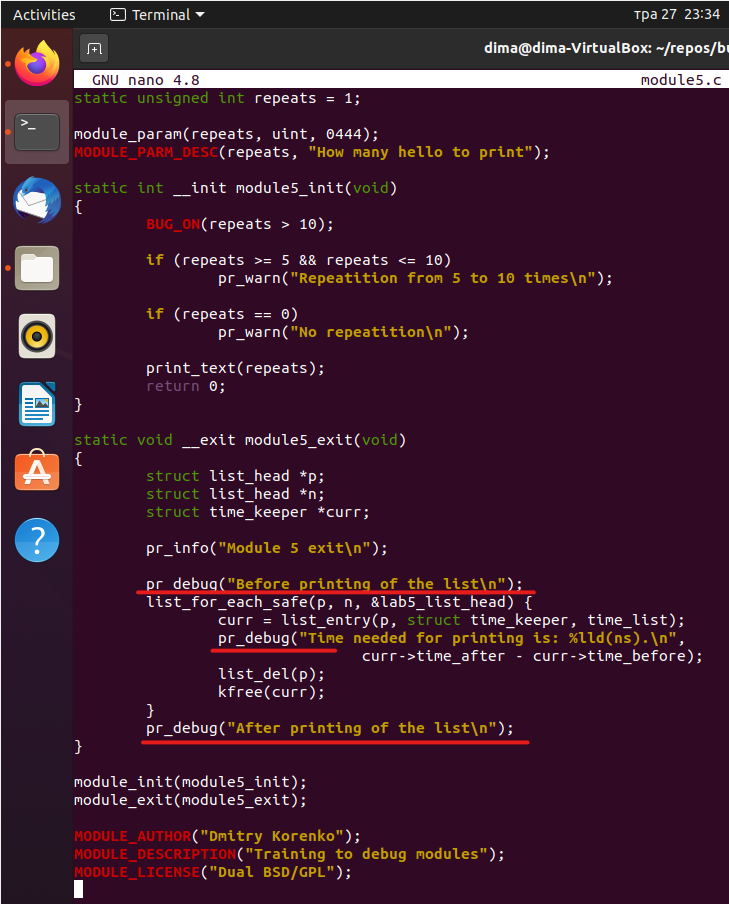




З’явився каталог /sys/kernel/debug/dynamic\_debug:



Нижче наведено змінений фрагмент коду:



Змінюючи прапорці у /sys/kernel/debug/dynamic\_debug/control можна змінювати формат повідомлення для всього модулю та для окремих рядків.

