Завдання розраховане на вже виконане завдання #4, і полягає у модифікації реалізації того завдання.

Зауваження:

1. Для відповідності Linux Kernel Coding Style ознайомтеся зі скриптом checkpatch.pl.
   1. Приклад використання $KDIR/scripts/checkpatch.pl -f файл.c
   2. Позначеного ERROR не повинно бути взагалі. З позначеного WARNING частина вас ще не стосується, проте простих речей на зразок “не треба на початку рядка ставити пробіл, а за ним табуляцію” також не повинно залишатися.
2. При виконанні роботи, не забувайте виконувати необхідні перевірки.

Завдання Basic:

1. Оголосити структуру даних для розміщення у списку, яка крім елемента struct list\_head містить поле типу ktime\_t  
   (include/linux/ktime.h у вашому репозиторії linux-stable).
2. Створити статичну змінну голови списку.
3. Перед кожним друком привітання виділити пам'ять для екземпляра оголошеної структури, занести в неї поточний час ядра, отриманий функцією ktime\_get().
4. У функції hello\_exit() пройти по списку і надрукувати час кожної події в наносекундах, вилучити елемент списку і звільнити виділену пам'ять. Приклад проходу по списку з вилученням елемента є у *appendix*.

На даному етапі досить виділяти пам'ять викликом

ptr = kmalloc(sizeof(\*ptr), GFP\_KERNEL);

і звільнити її викликом

kfree(ptr);

Докладніше у додатку *Allocating Memory*.

Завдання Advanced:

У цьому завданні вся робота зі списком виконується в модулі hello1. Також додати ще одне поле типу ktime\_t і засікати час до та після виклику функції друку, а на вивантаженні модуля надрукувати час, який пішов на кожен друк.

Звіт:

*Протокол має містити лістинг коду (для завдання Basic) або посилання на github/gitlab репозиторій, що містить файли з вихідним кодом та Makefile (для завдання Advanced), а також скріншоти виконання insmod/rmmod.*