Загружаемые модули ядра, ч. 2

ОС. Семинар №11

Закон программирования: ни одна переменная, ни одна функция, ни один тип не могут использоваться без объявления, описания и инициализации.

Мы работаем со структурой ядра через загружаемые модули (в 1м): printk, next_task - проходим по сведениям списку процессов.

В этой лабе: **взаимодействие загружаемых модулей ядра.**

4 файла: md1.c, md2.c, md3.c, md.h

- extern char* md1_data ←
  char* md1_proc() →

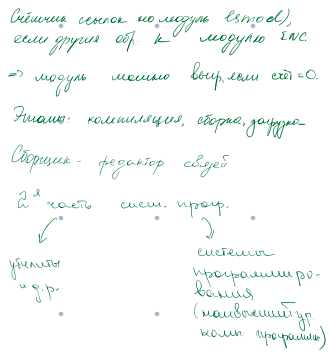
  указывает компилятору, что перем./функция описаны (определены) в другом файле исходного кода.

static char* md_local

ограничивает область видимости внутри 1го файла

EXPORT_SYMBOL( ) - макрос, с помощью которого в ядре осуществляется экспортирование (м.б. доступны для использования другими модулями).

Но для того чтоб другие модули могли использовать экспортированные символы, они должны "знать" их определения (в заголовочном файле).



Если другие модули обращаются к нек. модулю, то с помощью lsmod можно увидеть счётчик

ссылок на модуль

Этапы: компиляция, сборка, загрузка

Модуль можно выгрузить, если счётчик ссылок равен 0 (на него).