

## Задача ht01-1: c/libc/startup-1

Это задание — часть серии заданий на разработку подмножества стандартной библиотеки Си.

Реализуйте библиотечные функции `_exit` и `exit`, а также инициализационный код, который получает управление при запуске программы на выполнение (`_start`), выполняет инициализацию компонент библиотеки, а затем подготавливает аргументы для функции `main` и вызывает ее. После завершения `main` вызывается функция `exit`. Функция `main` должна вызываться со стеком, выровненным по 16 байт.

При инициализации библиотеки после запуска программы последовательно вызываются функции `__init_mem`, `__init_stdin`, `__init_stdout`, только после них вызывается `main`.

Функция `_exit` имеет прототип:

```
void _exit(int status) __attribute__((noreturn));
```

Эта функция просто вызывает соответствующий системный вызов.

Функция `exit` имеет прототип:

```
void exit(int status) __attribute__((noreturn));
```

Эта функция последовательно вызывает функции `__fini_stdout`, `__fini_stdin`, `__fini_mem`, затем передает управление функции `_exit`.

Все функции `__init_*` и `__fini_*` не принимают аргументов. Все функции `__init_*` и `__fini_*` могут отсутствовать в исполняемом файле. В этом случае они, естественно, не вызываются. Чтобы при компоновке программы не возникала ошибка отсутствующих внешних имен, соответствующие функции должны быть объявлены как *слабые* метки, например так:

```
.weak __fini_mem
```

Тогда, если соответствующая метка не объявлена ни в одном компоновочном модуле, она получит значение 0. Например, инструкция:

```
mov $__fini_mem, %eax
```

загрузит на регистр `%eax` адрес начала функции `__fini_mem`, либо 0, если эта функция отсутствует в программе.

Дополнительно (по указанию преподавателя) реализуйте инициализационный код как позиционно-независимый.