

РЕКЛАМА

АРБАН
ГРУППА СТРОИТЕЛЬНЫХ КОМПАНИЙ

**СДАЧА УЖЕ
В ЭТОМ ГОДУ!**

МИКРОРАЙОН
**SCANDIS
OZERO**

**2-комн.
квартира**

69 М²

от 9 453 тыс. руб.

Посмотреть

г. Красноярск,
ООО «СЗ Компания «Арбан»,
тел.: 8 (391) 205-15-15.
Проектная декларация
на сайте наш.дом.рф

- | | |
|---------------------------|---------|
| Лекция 1 | Тест 1 |
| Лекция 2 | Тест 2 |
| Лекция 3 | Тест 3 |
| Лекция 4 | Тест 4 |
| Лекция 5 | Тест 5 |
| Лекция 6 | Тест 6 |
| Лекция 7 | Тест 7 |
| Лекция 8 | Тест 8 |
| Лекция 9 | Тест 9 |
| Лекция 10 | Тест 10 |
| Дополнительный материал 1 | |
| Дополнительный материал 2 | |
| Экзамен | |

Вы можете **поддержать**
этот курс.

УЗНАЙТЕ ОСНОВНЫЕ ФУНКЦИИ EXCEL ЗА 3 ДНЯ!

БОЛЬШЕ НА [INTENSIVE.SKILLBOX.RU](https://intensive.skillbox.ru)

Учитесь и получайте официальные документы БЕСПЛАТНО. Вы можете поддержать наш проект.

ИТ
УНИВЕРСИТЕТ

ля Рейтинг Вопросы Магазины

Сведения об образовательной организации

А | Профессиональная переподготовка | Повышение квалификации | Сертификации

[Регистрация](#) [Вход](#)

Искать

Новости

Мобильное программирование приложений реального времени в стандарте POSIX: Мобильное программирование приложений реального времени в стандарте POSIX [+]

Реклама

РЕКЛАМА  Yandex DataLens

[Записаться](#) | [Вам нравится?](#) Нравится 6 студентам | [Поделиться](#) | [Поддержать программу](#)

Лекция 7: Асинхронный ввод/вывод, рекомендательные интерфейсы

А | версия для печати

[< Лекция 6](#) | | **Лекция 7:** 1 2 3 4 5 | | [Лекция 8 >](#)

Согласно стандарту POSIX-2001, списком можно не только инициировать запросы на асинхронный ввод/вывод, но и ожидать их завершения. Для этого служит функция `aio_suspend()` (см. листинг 7.5).

```
#include < aio.h>
int aio_suspend (
    const struct aiocb *const listio [],
    int nent,
    const struct timespec *timeout);
```

Листинг 7.5. Описание функции aio_suspend().

Поток управления, вызвавший `aio_suspend()` , приостанавливает выполнение на время, не большее, чем задано аргументом `timeout` , до тех пор пока не завершится по крайней мере одна из операций асинхронного ввода/вывода, на которую есть ссылка из массива `listio` (`nent` – число его элементов), или ожидание не будет прервано доставкой сигнала. Если какое-либо из перечисленных условий истинно в момент вызова, он завершается без задержек.

Нулевой результат свидетельствует о завершении по крайней мере одной из *операций ввода/вывода*. Какой именно, как обычно, нужно выяснять индивидуально, пользуясь функциями `aiio_error()` и `aiio_return()`.

Описываемая далее группа функций (см. листинг 7.6) предназначена для согласования состояния буферизованных и хранимых в долговременной памяти данных, обеспечения целостности данных и файлов. Подобные функции необходимы в системах обработки транзакций, чтобы гарантировать определенное состояние долговременной памяти, даже если система аварийно завершит работу. Один элемент из этой группы – функция `msync()` – мы уже рассматривали.

```
#include <unistd.h>
void sync (void);
```

```
#include <unistd.h>
int fsync (int fildes);
```

```
#include <unistd.h>
int fdatsync (int fildes);
```

```
#include <aio.h>
int aio_fsync (int op,
               struct aiocb *aiocbp);
```

Листинг 7.6. Описание функций синхронизации оперативной и долговременной памяти.

Функция `sync()` имеет глобальный характер – она планирует запись в файловые системы всех измененных данных.

Функция `fsync()` обеспечивает запись измененных данных в файл, заданный дескриптором `fd`. Более того, после успешного возврата из функции (с нулевым результатом) окажутся выполненными все стоявшие в очереди к этому файлу запросы на ввод/вывод с гарантией целостности файла.

Функция `fdatsync()` аналогична `fsync()`, но гарантируется лишь *целостность данных* (см. курс [1]).

Бесплатный вебинар «Первые шаги iOS-разработчика»

Узнайте, с чего начать программирование на iOS и выйти на доход 100 000 ₽/месяц!

mobdevfactory.com Реклама

- ого при
- , -
- оваться
- я.



Проиллюстрируем использование функций *асинхронного ввода/вывода* программой, подсчитывающей суммарное число строк в совокупности файлов (см. [листинг 7.7](#)).

```

/* **** */
/* Программа подсчитывает суммарное */
/* число строк - в файлах аргументах */
/* командной строки. */
/* Если аргументы отсутствуют или в */
/* качестве имени задан минус, */
/* читается стандартный ввод. */
/* **** */

```

```
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>
#include <fcntl.h>
#include <signal.h>
#include <string.h>
#include <aio.h>
#include <assert.h>
#include <errno.h>
```

```
int main (int argc, char *argv []) {
/* Указатель на массив */
/* указателей на управляющие */
/* блоки запросов операций */
/* асинхронного чтения */

    struct aiocb **lio_ptr;
    // Общее число строк в файлах
    long nlines = 0;
    // Признак повторного указания
    // стандартного ввода
    int dup_stdin = 0;

    int i;
```

```
/* Сведем случай с отсутствием */
/* аргументов к общему,          */
/* воспользовавшись одним из     */
/* немногих стандартизованных    */
/* файлов                          */
/*                                */
if (argc == 1) {
    argv[0] = "-";
} else {
    argv[0] = "/dev/null";
}
```

```
/* Зарезервируем память, */
/* откроем заданные файлы */
/* и сформируем начальный список */
/* запросов на чтение */
assert((lio_ptr =
(struct aiocb **) malloc
(sizeof(struct aiocb *) * argc))
!= NULL);
```

```
for (i = 0; i < argc; i++) {
    assert ((lio_ptr[i] =
             (struct aiocb *) malloc
             (sizeof (struct aiocb)))
            != NULL);
```

```

if (strcmp (argv [i], "-") == 0) {
    if (dup_stdin == 0) {
        lio_ptr [i] -> aio_fildes =
            STDIN_FILENO;
        dup_stdin = 1;
    } else {
        lio_ptr [i] -> aio_fildes =
            fcntl (STDIN_FILENO,
                F_DUPFD, 0);
    }
} else
    if ((lio_ptr [i] -> aio_fildes =

```

Бесплатный вебинар «Первые шаги iOS-разработчика»

Узнайте, с чего начать программирование на iOS и выйти на доход 100 000 ₽/месяц!
mobdevfactory.com Реклама





```

lio_ptr [i] = NULL;
continue;
}

lio_ptr [i] -> aio_offset = 0;
assert ((lio_ptr [i] -> aio_buf =
        malloc(BUFSIZ)) != NULL);
lio_ptr [i] -> aio_nbytes = BUFSIZ;
lio_ptr [i] -> aio_reqprio = 0;
lio_ptr
    [i] -> aio_sigevent.sigev_notify
        = SIGEV_NONE;
lio_ptr [i] -> aio_lio_opcode =
    LIO_READ;
} /* for (i < argc) */

/* Поехали ... */
assert (lio_listio (LIO_NOWAIT,
    lio_ptr, argc,
    &lio_ptr [0] -> aio_sigevent)
    == 0);

/* Будем ждать завершения */
/* операций ввода/вывода, */
/* обрабатывать прочитанные */
/* данные и */
/* инициировать новые запросы */
/* на чтение */
while (aio_suspend ((
    const struct aiocb **)
    lio_ptr,
    argc, NULL) == 0) {
    /* Выясним, какой запрос */
    /* и как выполнен */

    /* Число недочитанных файлов */
    int nreqs = 0;

    for (i = 0; i < argc; i++) {
        if (lio_ptr [i] == NULL) {
            continue;
        }

        /* Есть обслуживаемые */
        /* запросы */
        nreqs++;

        if (aio_error (lio_ptr [i])
            == EINPROGRESS) {
            continue;
        }
        {
            // Запрос выполнен

            // Число прочитанных байт
            ssize_t nb;

            if ((nb = aio_return
                (lio_ptr [i])) <= 0) {
                /* Дошли до конца файла */
                /* или чтение */
                /* завершилось ошибкой */
                (void) close(lio_ptr
                    [i] -> aio_fildes);
                free ((void *) lio_ptr
                    [i] -> aio_buf);
                free (lio_ptr [i]);
                lio_ptr [i] = NULL;
                nreqs--;
            } else {
                /* Обрабатываем прочитанные */
                /* данные */

```

Бесплатный вебинар «Первые шаги iOS-разработчика»

Узнайте, с чего начать программирование на iOS и выйти на доход 100 000 ₽/месяц!
mobdevfactory.com Реклама

РЕКЛАМА

АРБАН
ГРУППА СТРОИТЕЛЬНЫХ КОМПАНИЙ

**СДАЧА УЖЕ
В ЭТОМ ГОДУ!**

МИКРОРАЙОН
**SCANDIS
OZERO**

**2-комн.
квартира**

69 м²

от 9 453 тыс. руб.

Посмотреть

г. Красноярск,
ООО «СЗ Компания «Арбан»,
тел.: 8 (391) 205-15-15.
Проектная декларация
на сайте наш.дом.рф

```

/* Позиция, где нашли */
/* перевод строки */
char *p1;
p = (char *) lio_ptr [i] -> aio_buf;
while ((p1 =
(char *) memchr
(p, '\n',
nb - (p - (char *)
lio_ptr
[i] -> aio_buf)))
!= NULL) {
nlines++;
p = p1 + 1;
}

/* Иницируем новый */
/* запрос на чтение */
lio_ptr [i] -> aio_offset
+= nb;
(void) aio_read
(lio_ptr [i]);
}
}
} /* for (i < argc) */

/* Остались недочитанные */
/* файлы? */
if (nreqs == 0) {
break;
}
} /* while */

printf ("%ld\n", nlines);

return 0;
}

```

Листинг 7.7. Пример программы, использующей функции асинхронного ввода/вывода.

Отметим списочную форму начального представления и *ожидания* выполнения *запросов* на *асинхронное чтение*. Из технических деталей обратим внимание на пустой *указатель* в качестве элемента массива указателей на структуры типа `aiocb` (функция `lio_listio()` игнорирует такие элементы) и в качестве аргумента `timeout` функции `aio_suspend()` (означает бесконечное *ожидание*).

Второй вариант решения той же задачи (см. листинг 7.8) демонстрирует другой способ уведомления о завершении операции *асинхронного ввода/вывода* – *вызов функции*.

```

/* **** */
/* Программа подсчитывает суммарное число строк */
/* в файлах – аргументах командной строки. */
/* Если аргументы отсутствуют или в качестве имени */
/* задан минус, читается стандартный ввод. */
/* **** */

#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>
#include <fcntl.h>
#include <signal.h>
#include <string.h>
#include <aio.h>
#include <pthread.h>
#include <assert.h>

/* Указатель на массив указателей на управляющие */
/* блоки запросов операций асинхронного чтения */
static struct aiocb **lio_ptr;

/* Общее число строк в файлах */
static long nlines_total = 0;

/* Переменная условия, на которой ждут */
/* окончания обработки всех файлов */
static pthread_cond_t open_files_cond =
PTHREAD_COND_INITIALIZER;

/* Мьютекс. охраняющий доступ */

```

Бесплатный вебинар «Первые шаги iOS-разработчика»

Узнайте, с чего начать программирование на iOS и выйти на доход 100 000 ₽/месяц!
mobdevfactory.com Реклама

РЕКЛАМА

**Золотая
Корона**

**Ден
пер
из Р**

korona

УСЛУГА ПРЕДОСТАВЛЯЕТСЯ
ВЫРСОК, УЛ. КИРОВА, 46
С 01.04.2014 С ПЕРЕРЕЧ
ПЛЕКСНОМ ОБСЛУЖИВА
ГМ СМОТРИТЕ НА ЯНКО
ТЕКНИЧЕСКИЙ ЦЕНТР» (ООО)

РЕКЛАМА

АРБАН

ГРУППА СТРОИТЕЛЬНЫХ КОМПАНИЙ

СДАЧА УЖЕ
В ЭТОМ ГОДУ!

МИКРОРАЙОН

SCANDIS
OZERO



2-комн.
квартира
69 м²

от 9 453 тыс. руб.

Посмотреть

г. Красноярск,
ООО «СЗ Компания «Арбан»,
тел.: 8 (391) 205-15-15.
Проектная декларация
на сайте наш.дом.рф

```
/* **** */
/* Функция, вызываемая */
/* при завершении операции */
/* асинхронного ввода/вывода */
/* **** */

static void end_of_aio_op (union signal sg_vl) {
    int no;          /* Номер выполненного запроса */
                    /* в общем списке */
    ssize_t nb;      /* Число прочитанных байт */
    long nlines = 0; /* Число строк в одной */
                    /* прочитанной порции */

    no = sg_vl.sival_int;

    if ((nb = aio_return (lio_ptr [no])) <= 0) {
        /* Дошли до конца файла */
        /* или чтение завершилось ошибкой */
        (void) close (lio_ptr [no]->aio_fildes);
        free ((void *) lio_ptr [no]->aio_buf);
        free (lio_ptr [no]);
        lio_ptr [no] = NULL;
        (void) pthread_cond_signal (&open_files_cond);
    } else {
        /* Обрабатываем прочитанные данные */
        char *p; /* Текущее начало поиска перевода строки */
        char *p1; /* Позиция, где нашли перевод строки */

        p = (char *) lio_ptr [no]->aio_buf;
        while ((p1 = (char *) memchr (p, '\n',
                                     nb - (p - (char *) lio_ptr [no]->aio_buf)))
               != NULL) {
            nlines++;
            p = p1 + 1;
        }

        /* Прибавим локальную сумму к общей */
        (void) pthread_mutex_lock (&nlines_mutex);
        nlines_total += nlines;
        (void) pthread_mutex_unlock (&nlines_mutex);

        /* Иницируем новый запрос на чтение */
        lio_ptr [no]->aio_offset += nb;
        (void) aio_read (lio_ptr [no]);
    }
}

/* **** */
/* Функция проверяет, сколько осталось открытых файлов */
/* **** */

static int n_open_files (int nfiles) {
    int nof = 0; /* Число открытых файлов */
    int i;

    for (i = 0; i < nfiles; i++) {
        if (lio_ptr [i] != NULL) {
            nof++;
        }
    }

    return (nof);
}

/* **** */
/* Обработка аргументов командной строки, */
/* инициация начальных запросов на чтение, */
/* ожидание завершения обработки файлов, */
/* вывод результатов */
/* **** */

int main (int argc, char *argv []) {
    int dup_stdin = 0; /* Признак повторного указания */
                    /* стандартного ввода */

    int i;

    /* Блокировка стандартного ввода/вывода и ошибок */
}
```

Бесплатный вебинар «Первые шаги iOS-разработчика»

Узнайте, с чего начать программирование на iOS и выйти на доход 100 000 ₽/месяц!

mobdevfactory.com Реклама

РЕКЛАМА

Модуль

Полный курс для тех, кто хочет продавать на маркетплейсах. Мгновенные выплаты.

АО КБ «Мир» Лицензия

РЕКЛАМА

АРБАН
ГРУППА СТРОИТЕЛЬНЫХ КОМПАНИЙ

**СДАЧА УЖЕ
В ЭТОМ ГОДУ!**

МИКРОРАЙОН
**SCANDIS
OZERO**



**2-комн.
квартира**

69 м²

от 9 453 тыс. руб.

Посмотреть

г. Красноярск,
ООО «СЗ Компания «Арбан»,
тел.: 8 (391) 205-15-15.
Проектная декларация
на сайте наш.дом.рф

```

    argv [0] = "-";
} else {
    argv [0] = "/dev/null";
}

/* Зарезервируем память, откроем заданные файлы */
/* и инициализируем начальные запросы на чтение */
assert ((lio_ptr = (struct aiocb **) malloc
        (sizeof (struct aiocb *) * argc)) != NULL);

for (i = 0; i < argc; i++) {
    assert ((lio_ptr [i] = (struct aiocb *) malloc
            (sizeof (struct aiocb))) != NULL);

    if (strcmp (argv [i], "-") == 0) {
        if (dup_stdin == 0) {
            lio_ptr [i]->aio_fildes = STDIN_FILENO;
            dup_stdin = 1;
        } else {
            lio_ptr [i]->aio_fildes = fcntl (STDIN_FILENO,
                    F_DUPFD, 0);
        }
    } else if ((lio_ptr [i]->aio_fildes = open
            (argv [i], O_RDONLY)) == -1) {
        perror ("OPEN");
        free (lio_ptr [i]);
        lio_ptr [i] = NULL;
        continue;
    }

    lio_ptr [i]->aio_offset = 0;
    assert ((lio_ptr [i]->aio_buf = malloc (BUFSIZ)) != NULL);
    lio_ptr [i]->aio_nbytes = BUFSIZ;
    lio_ptr [i]->aio_reqprio = 0;

    lio_ptr [i]->aio_sigevent.sigev_notify = SIGEV_THREAD;
    lio_ptr [i]->aio_sigevent.sigev_signo = SIGRTMIN;
    lio_ptr [i]->aio_sigevent.sigev_value.sival_int = i;
    lio_ptr [i]->aio_sigevent.sigev_notify_function =
        end_of_aio_op;
    lio_ptr [i]->aio_sigevent.sigev_notify_attributes =
        NULL;

    /* Запрос готов, можно отправлять его на выполнение */
    (void) aio_read (lio_ptr [i]);
} /* for (i < argc) */

/* Дождемся завершения обработки всех указанных */
/* в командной строке файлов */
while (n_open_files (argc) > 0) {
    (void) pthread_cond_wait (&open_files_cond,
            &nlines_mutex);
}

printf ("%ld\n", nlines_total);

return 0;
}

```

Листинг 7.8. Модифицированный вариант программы, использующей функции асинхронного ввода/вывода.

В модифицированном варианте можно отметить применение *переменных условия* как средства *ожидания завершения* "вдвойне асинхронной" обработки файлов (по вводу/выводу и по выполнению функций процессирования прочитанных данных в рамках специально создаваемых потоков управления). Когда очередной *файл* закрывается, то для генерации *уведомления о возможном завершении* всей обработки вызывается функция `pthread_cond_signal()`. Обратим внимание на то, что в этот момент вызывающий *поток* не является владельцем мьютекса, ассоциированного с *переменной условия*, но в данном случае это и не требуется.

Дальше >>

< Лекция 6 || Лекция 7: 1 2 3 4 5 || Лекция 8 >

Бесплатный вебинар «Первые шаги iOS-разработчика»

Узнайте, с чего начать программирование на iOS и выйти на доход 100 000 ₽/месяц!
mobdevfactory.com Реклама

РЕКЛАМА

АРБАН
ГРУППА СТРОИТЕЛЬНЫХ КОМПАНИЙ

**СДАЧА УЖЕ
В ЭТОМ ГОДУ!**

МИКРОРАЙОН
**SCANDIS
OZERO**



**2-комн.
квартира
69 м²**


от 9 453 тыс. руб.

[Посмотреть](#)

г. Красноярск,
ООО «СЗ Компания «Арбан»,
тел.: 8 (391) 205-15-15.
Проектная декларация
на сайте наш.дом.рф

Реклама


Реклама



Курс «C#-разработчик»

3 месяца бесплатно - первый платеж через 90 дней.
Освойте C# с нуля до уровня Pro.

[Программа курса](#) | [Топ преподаватели](#)
[Процесс обучения](#) | [Записаться](#)


 skillbox.ru

[Узнать больше](#)

РЕКЛАМА

самолет **плюс**

**Открой свой бизнес
под крылом
сильного бренда
ГРУППЫ САМОЛЕТ**



Бесплатный вебинар «Первые шаги iOS-разработчика»

Узнайте, с чего начать программирование на iOS и выйти на доход 100 000 ₽/месяц!
mobdevfactory.com Реклама