

# Сетевое взаимодействие на языке Ляпас

## Содержание

<b>1</b>	<b>Описание функций</b>	<b>2</b>
<b>2</b>	<b>Примеры использования</b>	<b>4</b>

## 1 Описание функций

`socket(t/s)` - эта подпрограмма используется для создания сокета. На вход необходимо передать тип канала связи с сокетом, который должен быть использован. Подпрограмма возвращает целое положительное число - номер дескриптора или -1 в случае ошибки.

Таблица 1: Значения type

type	значение
SOCK_STREAM	1
SOCK_DGRAM	2

`bind(s,F1/e)` - используется сервером для присваивания сокету имени. До выполнения этой подпрограммы сокет недоступен программам-клиентам. Первый аргумент - сокет-дескриптор. Второй параметр - указатель на комплекс, описывающий sockaddr. В случае успешного выполнения подпрограмма возвращает значение равное 0, иначе - код ошибки.

Таблица 2: Комплекс F1

биты	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
значение	domain	port		IP address				0	0	0	0	0	0	0	0	0

`set_sockaddr_port(F1,p)` - подпрограмма, позволяющая заполнить поле комплекса sockaddr значением port.

`set_sockaddr_ip(F1,F2)` - подпрограмма, позволяющая заполнить поле комплекса sockaddr значением, взятым из комплекса F2, соответствующим IP адресу. В случае, если комплекс заполнен нулями, для установления соединения будут использованы все IP-адреса сервера.

`listen(s,b/e)` - используется сервером, чтобы уведомить операционную систему, что он ожидает запросы на установление соединения на данном сокете. Первый аргумент - сокет для прослушивания, второй аргумент - целое положительное число, определяющее как много запросов может быть принято на сокет. В случае успешного выполнения подпрограмма возвращает значение равное 0, иначе - код ошибки.

`accept(s,F1/s)` - используется сервером для принятия соединения на сокет. Сокет должен быть уже слушающим в момент вызова подпрограммы. Если сервер устанавливает соединение с клиентом, то подпрограмма accept возвращает новый сокет-дескриптор, через который и происходит общение клиента с сервером, в противном случае -1. Пока устанавливается соединение клиента с сервером, подпрограмма accept блокирует другие запросы соединения с данным сервером, а после установления связи "прослушивание" запросов возобновляется. Первый аргумент подпрограммы - сокет-дескриптор для принятия соединений от клиентов. Второй аргумент - комплекс F1, для соответствующего домена. Второй аргумент заполняются соответствующими значениями в момент установления соединения клиента с сервером и позволяют серверу точно определить, с каким именно клиентом он общается.

`connect(s,F1/e)` - данная подпрограмма используется клиентом для установления соединения с сервером. Первый аргумент - сокет клиента, второй - комплекс F1, соответствующий серверу. Если соединение прошло успешно возвращается значение равное нулю, иначе - код ошибки.

Таблица 3: Значения e

Значение	Описание ошибки
1	Функция заблокирована межсетевым экраном
2	Сокета с заданным номером не существует
4	Системный вызов был прерван сигналом
9	Некорректный номер дескриптора
11	В данный момент функция не может быть выполнена, попробуйте позже
12	Недостаточно свободного места для создания сокета
13	Запрещено создание сокета с заданными параметрами
14	Некорректный логический адрес параметра
22	Некорректное значение аргумента
23	Переполнение таблицы дескрипторов
24	Слишком много открытых файлов
30	Файловая система доступна только на чтение
36	Комплекс F1 слишком длинный
71	Ошибка протокола
88	Значение аргумента s, не является дескриптором
91	Тип сокета не поддерживает протокол, выбранный сервером
93	Данный тип протокола не поддерживается
95	Функция не поддерживает данные типы сокетов
98	Сетевой адрес уже используется
99	Требуемый сетевой адрес не может быть использован
101	Сеть недоступна
103	Соединение разорвано
105	Недостаточно свободного места для создания сокета
106	Соединение на сокете уже произошло
110	Истечение времени ожидания
111	Не обнаружен слушающий порт сервера
114	Сокет является неблокирующим, а предыдущая попытка установить соединение ещё не завершена
115	Сокет является неблокирующим, а соединение не может быть установлено в данный момент

## 2 Примеры использования

### Эхо-сервер

```
main(/)
  Os Oe Oc Oa
  @+F1(16) @+F2(16) @+F3(16) @+F4(10)
  @'0.0.0.0'>F2
  *set__sockaddr__port(F1,1195/)
  *set__sockaddr__ip(F1,F2/)
  *socket(1/s)
  *bind(s,F1/e)
  *listen(s,100/e)
§1 *accept(s,F3/c)
  *freadf(c,F4,0,5/a)
  *fwritef(c,F4,0,5/a)
  *close(c/)
  →1
  *close(s/)
  **
```

### Эхо-клиент

```
main(/)
  Os Oe Oa
  @+F1(16) @+F2(16) @+F3(5) @+F4(5)
  @'192.168.203.1'>F2
  @'echo'>F3
  *set__sockaddr__port(F1,1195/)
  *set__sockaddr__ip(F1,F2/)
  *socket(1/s)
  *connect(s,F1/e)
  *fwritef(s,F3,0,5/a)
  *freadf(s,F4,0,5/a)
  *fwritef(1,F4,0,5/a)
  *close(s/)
  **
```