

Использование системного вызова `mknod` для создания FIFO

Прототип системного вызова

```
#include <sys/stat.h>
#include <unistd.h>

int mknod(char *path, int mode, int dev);
```

Описание системного вызова

Нашей целью является не полное описание системного вызова `mknod`, а только описание его использования для создания FIFO. Поэтому мы будем рассматривать не все возможные варианты задания параметров, а только те из них, которые соответствуют этой специфической деятельности.

Параметр `dev` является несущественным в нашей ситуации и мы будем всегда полагать его равным 0.

Параметр `path` является указателем на строку, содержащую полное или относительное имя файла, который будет являться меткой FIFO на диске. Для успешного создания FIFO файла с таким именем перед вызовом не должно существовать.

Параметр `mode` устанавливает атрибуты прав доступа различных категорий пользователей к FIFO. Этот параметр задается как результат побитовой операции "или" значения `S_IFIFO`, указывающего, что системный вызов должен создать FIFO, и некоторой суммы следующих восьмеричных значений:

- 0400** - Разрешено чтение для пользователя, создавшего FIFO.
- 0200** - Разрешена запись для пользователя, создавшего FIFO.
- 0040** - Разрешено чтение для группы пользователя, создавшего FIFO.
- 0020** - Разрешена запись для группы пользователя, создавшего FIFO.
- 0004** - Разрешено чтение для всех остальных пользователей
- 0002** - Разрешена запись для всех остальных пользователей

При создании FIFO реально устанавливаемые права доступа получаются из стандартной комбинации параметра `mode` и маски создания файлов текущего процесса `umask`, а именно - они равны `(0777 & mode) & ~umask`.

При успешном создании FIFO системный вызов возвращает значение 0, при неуспешном - отрицательное значение.