

## Некоторые системные вызовы

**Fork** – порождение нового процесса имеет следующий формат:

**Pid** = fork();

**Int** pid;

Системный вызов `fork` служит для создания нового процесса. Новый процесс (процесс потомок) является полной копией процесса предка за исключением того что процесс потомок имеет свой идентификатор процесса PID.

Новый процесс имеет собственную копию таблицы дескрипторов открытых файлов.

Но эти дескрипторы ссылаются на те же самые файлы, что и дескрипторы процесса предка. В случае успешного завершения системного вызова `fork` возвращает значение 0 порожденному процессу и идентификатор нового процесса – породившему. В случае неуспешного завершения, возвращается `minus1 ( -1 )`.

**nice** – изменение приоритета процесса

**int** nice(int *inc*);

**nice** добавляет значение аргумента *inc* к значению `nice`-приоритета вызывающего процесса (большее значение `nice` означает меньший приоритет). Только супер-пользователь может задавать отрицательные значения или определять увеличение приоритета. При удачном завершении вызова возвращаемое значение равно нулю. При ошибке оно равно -1.

**Wait** – Ожидание завершения процесса-потомка

**int** wait (int \**status*)

**Wait** возвращает идентификатор процесса, изменившего статус и через переменную *status* – код статуса. Тип изменения статуса: потомок завершился, потомок остановлен по сигналу или продолжил работу.