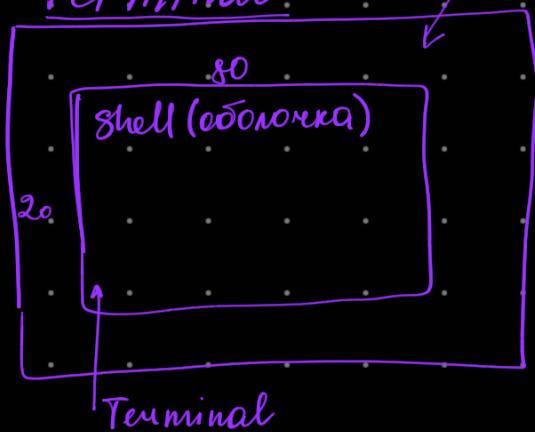


# Сессия 1

- 1. ОС, POSIX/C
- 2. ООП / git / C++
- Terminal



Desktop

основы  
sh  
bash

while (!eof)  
{  
 print ("prompt")  
 getline();  
 pause();  
 execute();  
}

Секция с строками

command      |  
                  |  
                  args

hello      a      b      c  
              простой/мног

- internal (bash) (например: help)
- external (& programs) (pwd)

echo (и inter., и external)

если нужно повторить, то будет internal

| use /bin/echo  
/bin/echo

bin  
use  
bin  
echo

(/bin/  
use) - большинство команд из этих

Число получение помощью по int. с помощью help (man)

no exten. с помощью man

man

- Така:
- 1) man(prog)
  - 2) syscalls
  - 3) libprog

printf (в нескольких языках)

Число к определенному языку man 3 printf

apropos - помощь, осущ. поиск по описанию программ

~~Компилятор~~  
GNU - ~~GNU is not Unix~~

- gcc -
- clang (с помощью Apple)

Вход компилятора gcc hello.c

- пишет
  - компилирует
  - пишет
- запись в файл a.out

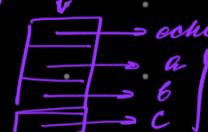
Число выполнено ошибки. нужно использовать инструкции

gcc -Wall hello.c -o hello  
режим записи

echo → myecho

int main( int ac, char\* a[ ] )  
число  
кот. со значениями

массив  
строк



вывод через пробел и перевод строки

-n последний символ не в бордюре (можно! echo -n . <sup>нажат</sup> <sub>перевод</sub> )

discored. gg/rEqNB3

личное сообщение → kill



## Документация

fopen :  
-fopen(3)...

### SYNOPSIS

#include  $\Rightarrow$  copy & paste

FILE \*fopen(path).

type type

### DESCRIPTION

path

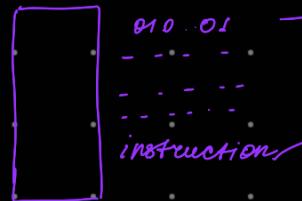
Поиск в manual проходием с помощью /path повторной  $\rightarrow$

## ERRORS

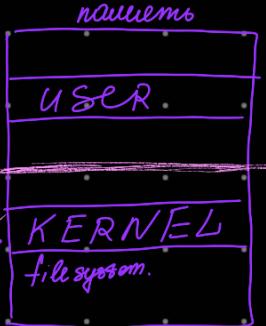
в 90% случаев отладочное число

## SEE ALSO

## file-programm



execute



запуск

process	table
U	K
[pid]	ab, c

CPU context

context t

user context

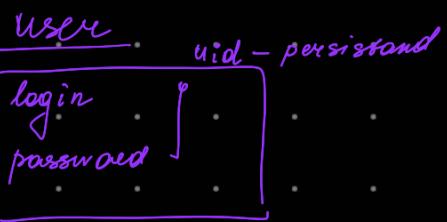
kernel context  
(egno)

fopen  $\xrightarrow{\text{Linux}}$  open  
~~Windows~~ CreateFile

syscall - getpid()

(pid)

... init = /etc/init.d



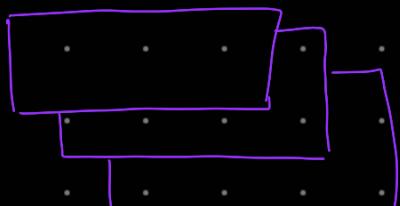
uid = 1056

primary group



gid (oh) = gid (process)

supplementary groups



$\in$  list (groups)

\$ id

\$ root

myid

Kak freemint

getuid() - user

getgid() - group

getgroups() - s. groups

o namespace namespace

pwd - names o namespaces

group

getpwuid(  
uid)

getpwuid()  
getgrgid()

getpwent("root")  
getgrouplist()

- Могут смотреться с мес., что uid/group есть, а имени нет.  
Если имен нет, то выдают ошибку прошу uid.
- Программа не должна падать, если некорректно введена

```
if argc == 1
{
    getuid()
    getgroups()
    getgid()
}

else
{
    getpwnam
    getgrouplist
    getgr...
}

print(...)
```

path - унаследован

римо введеное command 0-terminated

/ - указывает один из трех на группы

/user/bin - absolute

use /bin/ls - relative

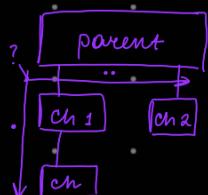
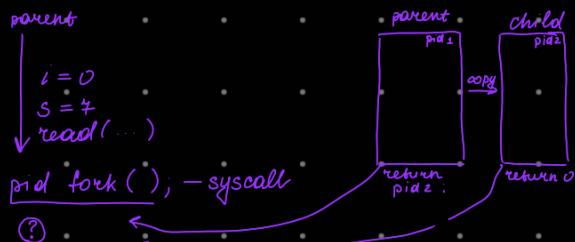
current directory / (могут указать /home/username)

./hello - запускаем из текущей директории  
↑↑↑↑  
они лежат parent

Базируется на текущем  
+ наследуете из предыдущего

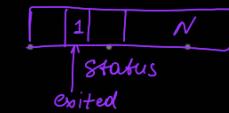






wait(&st); // block (будет ждать и не вспыльчика, когда дочерний процесс завершится)  
// статус завершения

main()
 exit(N);
 return N;
 g

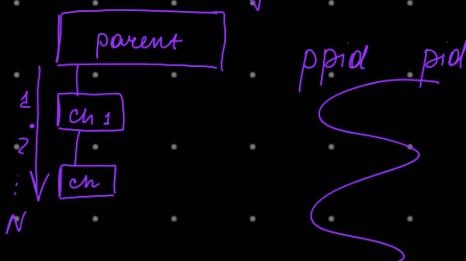


WIFEXITED(st)  
(true/false)

WEXITSTATUS(st)

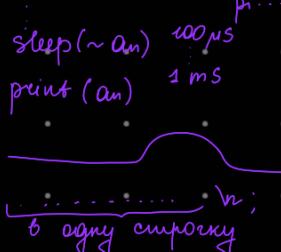
} макрос, чтобы упростить статус

называем N



②

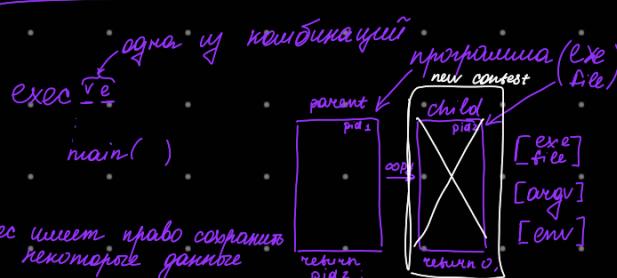
Sleepsort  $\langle a_1, a_2, \dots, a_n \rangle$



strace ./a.out

Write ("Hello world")

strace -f ./a.out



(из группы фундаментальных)

execvp

execclp

выбираем -l, если ошибки; если успешно, то не выб

On

mytime ls //child2 execvp  
time = 1.043

//parent

struct gettimeofday (&t)

)

KAK TEST (mytime sleep 5)

использует программу, где

использование времени где вебога

не должно падать, если программа

