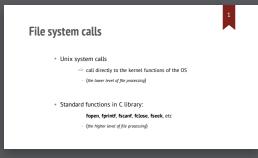
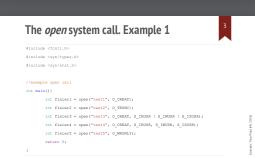
## Operating systems 1, Lecture 6

Sanda-Maria AVRAM, Ph. D.





```
The open system call. Example 3

**include 
**include <pr
```

```
The creat system call Example

#include 
#incl
```

```
Functions read and write

#include cunistd.h>

#size_t read(int fd, void *buf, size_t count);

#size_t write(int fd, void *buf, size_t count);

Return: 0 = EOF

-1 = error

>0 = number of bytes operated.
```

```
Functions read and write. Example 2

...

//example of reading a sentence and adding a word
int main(){
    int rister = open("cont1", O_RDMR);
    char buff[256];
    read(fision; buff, 1);
    int i = 0;
    while(buff[i+]|a".") read(fision; buff + i, 1);
    buff(i) = 0;
    varies(i, buff, 1);
    write(fision; "Bub", 3);
    return 0;
}
```

```
Function (seek, Example 2

#include cfonti,ho
#include constant,ho
#include cays/types,ho
#
```

```
System calls dup and dup2

#include contact.h>

int dup(int oldfd);
int dup2(int oldfd, int newfd);

Duplicates the file descriptor oldfd as follows:

- oldfd and newfd refer to the same physical file;

- the access mode is retained from the opening;

- both descriptors share the same current pointer in the file.

Return: -1 - error;

>0 - the new descriptor.
```

```
- the access mode is retained from the opening;
- both descriptors share the same current pointer in the file.

Return: -1 - error;
>0 - the new descriptor.

##Include <atclib.h>
##Include <atclib.h
```

```
The open system call. Example 2

#include <fort.in>
```

```
The creat system call

#include <pys/types.hb
#include <pys/tatt.hb
#include <forti.nb

int creat(const char * pathname, mode;t mode);

/* equivalent to specifying the options: O_MEGRITIO_THORC for the open function */
```





```
Functions read and write. Example 3

...

#include datio.hb

// read write example

int main(){

int fisier = open("const", 0_NRONLY | 0_RBONLY); // not equivalent to 0_RBON

char buff256|;

buff(0) = 'A';

printf("ta\n", read(fisier, buff, 1)=-17"Error": "No error");

printf("ta\n", write(fisier, buff, 1)=-17"Error": "No error");

raturn 0;

1
```

```
Char buff(1);

| lawak(fisier, 7, SEEE_COD);
| read(fisier, buff, 1);
| print("%", "buff);

| lawak(fisier, buff, 1);
| print("%", "buff);

| Definite ("%", "buff);

| Definite ("%", "buff);
| Definite ("%", "buff);
| Definite ("%", "buff);
| Definite ("%", "buff);
| Definite ("%", "buff);
| Definite ("%", "buff);
| Definite ("%", "buff);
| Definite ("%", "buff);
| Definite ("%", "buff);
| Definite ("%", "%");
| Definite ("%");
| Definite ("
```

```
# Almolude Capy/Types.Do

# Almolude Capy/Types.Do

# Almolude Capy/Types.Do

# Include Capy/Types.Do

# ...

| Include Capy/T
```

```
Standard functions in the C library

**Include <atdio.h>

**FILE **topen(connt char *path, connt char *mode);
int fclose(FILE *fp);

int fprint(FILE *stream, connt char *format, ...);
int facan(FILE *stream, connt char *format, ...);
size_tread(void *ptr, size_t size_t nemeb, FILE *stream);
size_t fread(void *ptr, size_t size_t nemeb, FILE *stream);
int facan(FILE *stream, long offset, int whence);

ment.
```

```
The dup2 system call. Example 1

#include cfontlibe

int main(){

int main(){

int fisier! = open("fisier!", O_NNONIXY);

dup2(fisier), 1);

dup2(1, 2);

print("Example dup2 \n");

perror("Example secorth");

return 0;

}
```

```
The fcntl system call

finctions -(unistd.h)>
finctions -(contin.h)

int fenet(int fa, int cnd, ... /* arg */ );

Provides or changes properties of an open file;

cmd: F_DUFFD -duplicating a descriptor
F_GETFD (or F_SETID) - file descriptor flags
F_GETFL (or F_SETID) - file status flags
```







