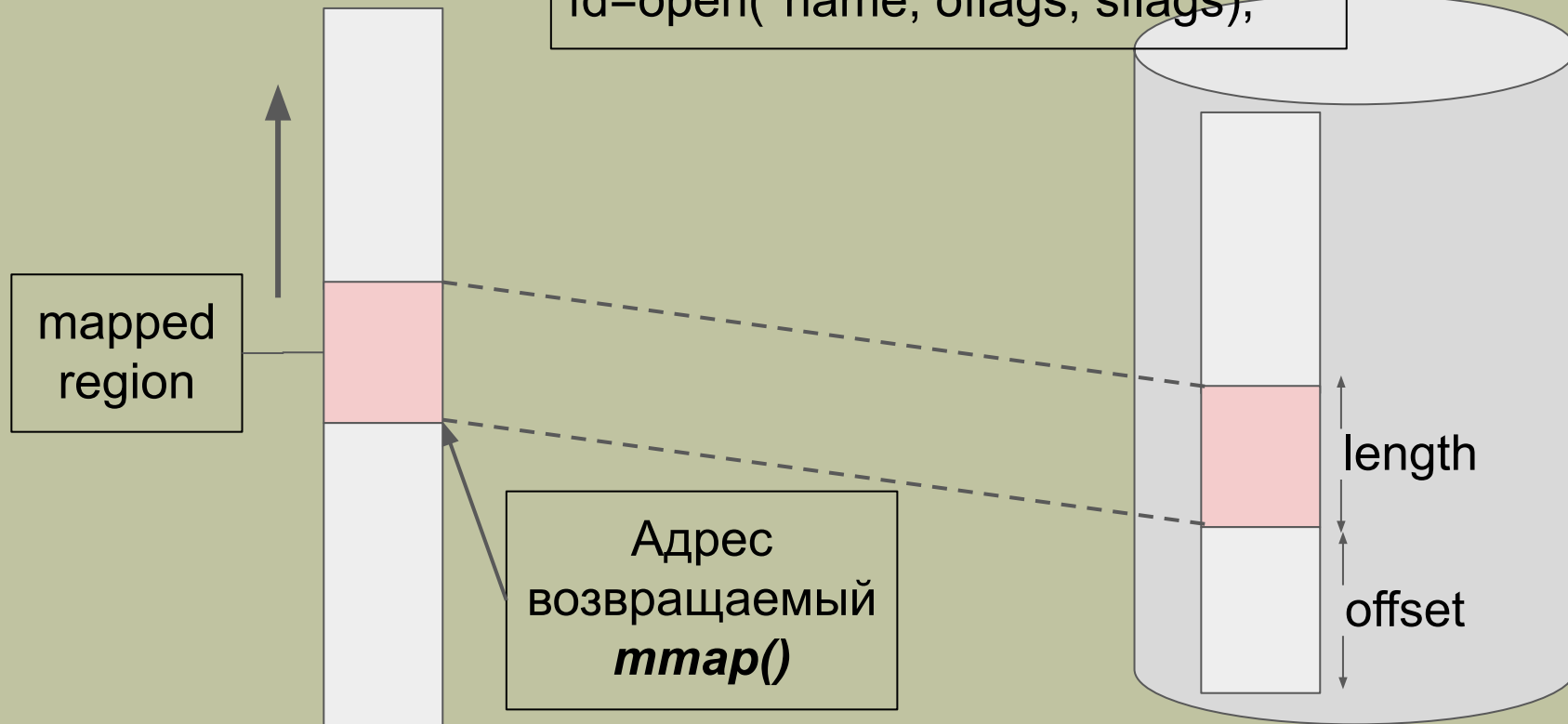# Лекция 13

- Отображение файлов в память.
- Выделение разделяемой памяти. Именованные семафоры *POSIX*.

Адресное пространство процесса

```
pAddress=mmap(*addr,length,prot,flags,
              fd, offset);
```

fd=open(*name, oflags, sflags);

mapped region

Адрес возвращаемый *mmap()*

length

offset

prot: PROT_READ, PROT_WRITE, PROT_EXEC

flags: MAP_PRIVATE, MAP_SHARED

oflags: O_RDONLY, O_WRONLY, O_RDWR, O_CREAT, O_APPEND,...

oflags: S_IRUSR, S_IWUSR, S_IXUSR,
        S_IRGRP, S_IWGRP, S_IXUSR,
        S_IROTH, S_IWOTH, S_IXOTH

```
> ls -l
-rw-r--r-- 1 malkov users     69 Dec  6 12:16 s_test
```

```
>chmod   760 s_test
>ls -l
-rwxrw---- 1 malkov users     69 Dec  6 12:17 s_test
```

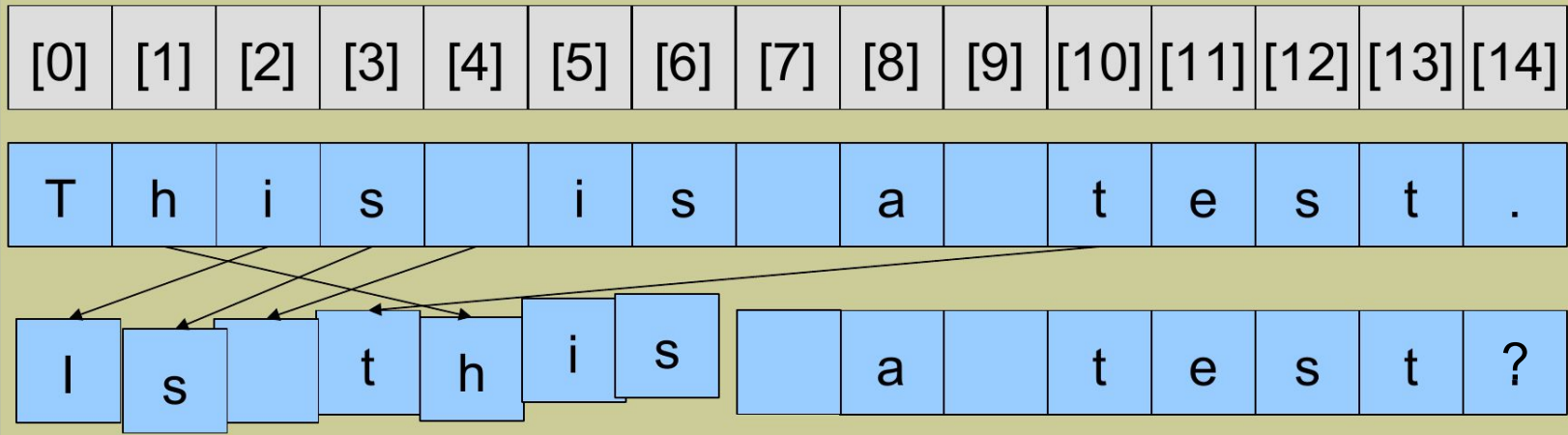S_IRUSR | S_IWUSR | S_IXUSR |
S_IRGRP, S_IWGRP

```
int fd;          //дескриптор файла
FILE* fp;     //файловый поток (указатель на структуру)
fd=open(*name, oflags, sflags);
read(fd,...);
write(fd,…);
lseek(fd,...);
fp=fopen(*name, mode); //mode: r,w,a,r+w+
fread(...,fp);
fwrite(…,fp);
fseek(fp,...);
fd=fileno(fp);
fp=fdopen(fd, mode);
close(fd);
fclose(fp);
```

```c
#include <stdlib.h>
#include <stdio.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <time.h>
#include <unistd.h>
#include <sys/mman.h>

int main(int argc, char* const argv[]){
  int fd;
  struct stat stat_file;
  char dummy;
  char* map_address;
```

lab13b.c

```c
fd=open("test.txt", O_RDWR | O_CREAT,
        S_IRUSR | S_IWUSR | S_IRGRP);
  if (fd == -1)
    fprintf(stderr, "open\n");

  if(fstat(fd, &stat_file))
    fprintf(stderr, "fstat\n");

  map_address=(char*)mmap(0,stat_file.st_size,
            PROT_READ | PROT_WRITE,
            MAP_PRIVATE, fd, 0);

  if (map_address == MAP_FAILED)
    fprintf(stderr, "mmap\n");
```

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]
T	h	i	s		i	s		a		t	e	s	t	.

I s   t h i s   a   t e s t ?

```
dummy=map_address[1];
map_address[0]=map_address[5]-0x20;
map_address[1]=map_address[3];
map_address[2]=map_address[4];
map_address[3]=map_address[10];
map_address[4]=dummy;
map_address[14]=63;
```

```
  write(fd, map_address, stat_file.st_size);

  munmap(map_address, stat_file.st_size);

  close(fd);

  return 0;
}
```

```c
#include <stdio.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <unistd.h>
#include <sys/mman.h>

int main(int argc, char* const argv[]){
  int fd;
  struct stat stat_file;
  char dummy;
  char* map_address;
```

lab13c-1.c

```c
fd=open("test_shared.txt", O_RDWR | O_CREAT,
        S_IRUSR | S_IWUSR | S_IRGRP);
if (fd == -1)
  fprintf(stderr, "open\n");

map_address=(char*)mmap(0,256,
            PROT_READ | PROT_WRITE,
            MAP_SHARED, fd, 0);

if (map_address == MAP_FAILED)
  fprintf(stderr, "mmap\n");
```

```
 write(fd, "Hello, students!\n", sizeof("Hello, students!\n"));
 getc(stdin);

 munmap(map_address, 256);
 close(fd);

 return 0;
}
```

```c
#include <stdio.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <unistd.h>
#include <string.h>
#include <sys/mman.h>

int main(int argc, char* const argv[]){
  int fd;
  struct stat stat_file;
  char dummy;
  char* map_address;
  char buf[256];

  memset(buf,0,256);
```

lab13c-2.c

```c
fd=open("test_shared.txt", O_RDWR);
 if (fd == -1)
   fprintf(stderr, "open\n");

 map_address=(char*)mmap(0,256,
           PROT_READ | PROT_WRITE,
           MAP_SHARED, fd, 0);

 if (map_address == MAP_FAILED)
   fprintf(stderr, "mmap\n");
```

```
  write(fileno(stdout), map_address, sizeof(buf));

  munmap(map_address, 256);
  close(fd);

  return 0;
}
```

```
00600000-00601000 r--p 00000000 08:13 1690022690   ~/lab13c-1
00601000-00602000 rw-p 00001000 08:13 1690022690   ~/lab13c-1
0077a000-0079b000 rw-p 00000000 00:00 0            [heap]

7fd912f68000-7fd912f8d000 r-xp 00000000 00:2d 18855
                              /lib64/ld-2.26.so
7fd91318c000-7fd91318d000 rw-s 00000000 08:13 1690064245
                    ~/Lecture13/mapipc/test_shared.txt
7fd91318d000-7fd91318e000 r--p 00025000 00:2d 18855
                              /lib64/ld-2.26.so
```

```
00600000-00601000 r--p 00000000 08:13 1690022690  ~/lab13c-1
00601000-00602000 rw-p 00001000 08:13 1690022690  ~/lab13c-1
0077a000-0079b000 rw-p 00000000 00:00 0            [heap]

7fd912f61000-7fd912f64000 rw-p 001b4000 00:2d 18863
                              /lib64/libc-2.26.so
7fd912f64000-7fd912f68000 rw-p 00000000 00:00 0
7fd91318c000-7fd91318d000 rw-s 00000000 08:13 1690064245
                              ~/test_shared.txt

7fd91318e000-7fd91318f000 rw-p 00026000 00:2d 18855
                              /lib64/ld-2.26.so
```

```c
#include <stdio.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <unistd.h>
#include <sys/mman.h>
int main(){
  int fd;
  fd=shm_open("common_region",
          O_RDWR | O_CREAT,
          S_IRUSR | S_IWUSR | S_IRGRP);
  write(fd, "Hello, students!\n", sizeof("Hello, students!\n"));
  getc(stdin);
  close(fd);
  return 0;
}
```

lab13d-1.c

```c
#include <unistd.h>
#include <string.h>
#include <sys/mman.h>
#include <sys/stat.h>
#include <fcntl.h>
#define SIZE_BUF 256

int main(int argc, char* const argv[]){
  int fd;
  char buf[SIZE_BUF];

  fd=shm_open("common_region", O_RDWR,
         S_IRUSR | S_IWUSR | S_IRGRP);
```

lab13d-2.c

```
  memset(buf,0, SIZE_BUF);
  read(fd, buf,SIZE_BUF);
  write(fileno(stdout), buf, SIZE_BUF);


  close(fd);
  return 0;
}
```

```
mapshm> gcc lab13d-1.c -lrt -o lab13d-1
```

```
mapshm> gcc lab13d-2.c -lrt -o lab13d-2
```

```c
#include <semaphore.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <sys/mman.h>

int main( void ) {
  int n=0;
  int fd;
  char* sh;
  sem_t  *sem;
```

lab13e-1.c

```
fd=shm_open("common_region",
        O_RDWR | O_CREAT,
        S_IRUSR | S_IWUSR | S_IRGRP);
 if (fd == -1)
   fprintf(stderr, "shm_open\n");

ftruncate(fd, 6);

sh=(char*)mmap(0,6,
            PROT_READ | PROT_WRITE,
            MAP_SHARED, fd, 0);
 if (sh == MAP_FAILED)
   fprintf(stderr, "mmap\n");

memset(sh,0,6);
```

```c
sem=sem_open("common_sem", O_CREAT,
        S_IRUSR | S_IWUSR | S_IRGRP, 1);
if (sem == SEM_FAILED)
  fprintf(stderr, "sem_open");

while(n++<200){
 sem_wait(sem);
  //write(fileno(stdout),sh, 6);
 printf("String: %s\n",sh);
 sem_post(sem);
 usleep(100);
}
```

```c
  shm_unlink("common_region");
  munmap(sh, 6);

  sem_unlink("common_sem");
  sem_close(sem);
  return 0;
}
```

```
mapshm> gcc lab13e-1.c -lpthread -lrt -o lab13e-1
```

```
mapshm> gcc lab13e-2.c -lpthread -lrt -o lab13e-2
```

```c
#include <semaphore.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <sys/mman.h>

int main(){
  int n=0;
  int counter=0;
  int fd;
  char* sh;
  sem_t  *sem;
```

lab13e-2.c

```c
fd=shm_open("common_region",
        O_RDWR | O_CREAT,S_IRUSR | S_IWUSR | S_IRGRP);

 sh=(char*)mmap(0,6,
            PROT_READ | PROT_WRITE,
            MAP_SHARED, fd, 0);
 memset(sh,0,6);


 sem=sem_open("common_sem", 0);
```

```
while ( n++<200 ){
  sem_wait(sem);
  if(counter%2){
    sh[0]='H';sh[1]='e';sh[2]='l';sh[3]='l';sh[4]='o';sh[5]='\0';
  }
  else{
    sh[0]='B';sh[1]='y';sh[2]='e';sh[3]='_';sh[4]='u';sh[5]='\0';
  }
  sem_post(sem);
  counter++;
  usleep(100);
 }
 munmap(sh, 6);
 return 0;
}
```

```
00602000-00603000 rw-p 00002000 08:13 21720831  ~/lab13e-1
0163d000-0165e000 rw-p 00000000 00:00 0            [heap]
7fa3f8a44000-7fa3f8bf5000 r-xp 00000000 00:2d 18863
                                  /lib64/libc-2.26.so
77fa3f9019000-7fa3f901a000 rw-p 0001a000 00:2d 18889
                                  /lib64/libpthread-2.26.so
7fa3f9225000-7fa3f9226000 rw-p 00007000 00:2d 18893
                                  /lib64/librt-2.26.so
7fa3f9226000-7fa3f924b000 r-xp 00000000 00:2d 18855
                                  /lib64/ld-2.26.so


7fa3f9444000-7fa3f9445000 rw-s 00000000 00:17 79254
                                  /dev/shm/common_region
```

```
00400000-00401000 r-xp 00000000 08:13 21720846   ~/lab13e-2
........................................................................................................
7f34c6dee000-7f34c6def000 rw-s 00000000 00:17 79255
                              /dev/shm/sem.common_sem
7f34c6def000-7f34c6df0000 rw-s 00000000 00:17 79254
                              /dev/shm/common_region

........................................................................................................
```