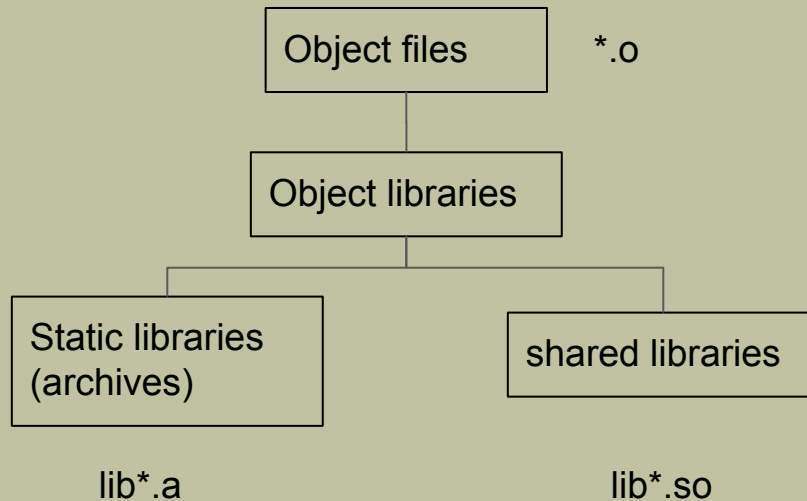


Лекция 5

- Библиотеки объектных модулей (архивы и разделяемые библиотеки).
- ELF (***E**xecutable and **L**inking **F**ormat*) файлы.

Статические библиотеки - архивы, и библиотеки динамической компоновки.



```
~/Lab5> cat lab5-1.c
```

```
double fun1(double x, int n){  
    return n*x;  
}
```

```
int fun2(int n){  
    int m;  
    m=n*n;  
    return m;  
}
```

```
double y=12.4;
```

```
~/Lab5> cat lab5-2.c
```

```
double z=2.87;
```

```
double gun1(){  
    int a[3]={2,3,4};  
    return (double)(a[0]+a[1]+a[2])/3.0;  
}
```

~/Lab5> cat lab5.c

```
#include <stdio.h>
double fun1(double, int);
int fun2(int);
double gun1();
extern double y;
extern double z;
int main(){
    fprintf(stdout, "%g\t%d\t%g\t%g\t%g\n",
        fun1(0.1,123), fun2(8), y,gun1(),z);
return 0;
}
```

```
~/Lab5> vim lab5-1.c
~/Lab5> vim lab5-2.c
~/Lab5> gcc -c lab5-1.c lab5-2.c
~/Lab5> ar cr liblab5.a *.o
~/Lab5> vim lab5.c
~/Lab5> gcc -c lab5.c
~/Lab5> gcc lab5.o -L. -llab5 -o lab5
~/Lab5> ./lab5
12.3      64      12.4      3      2.87
```

```
~/Lab5> gcc -c -fPIC -Wall lab5-*.c
~/Lab5> gcc -shared lab5-*.o -o liblab5.so
~/Lab5> gcc -c lab5.c
~/Lab5> gcc lab5.o -L. -llab5 -o lab5s
~/Lab5> export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:.
~/Lab5> ./lab5s
12.3      64      12.4      3      2.87
```

```
~Lab5> cat lab5d.c
```

```
#include <stdio.h>
#include <stdlib.h>
#include <dlfcn.h>
typedef double (*fun) (double, int);
typedef double (*gun) ();
extern double y,z;

int main() {
    gun g;
```



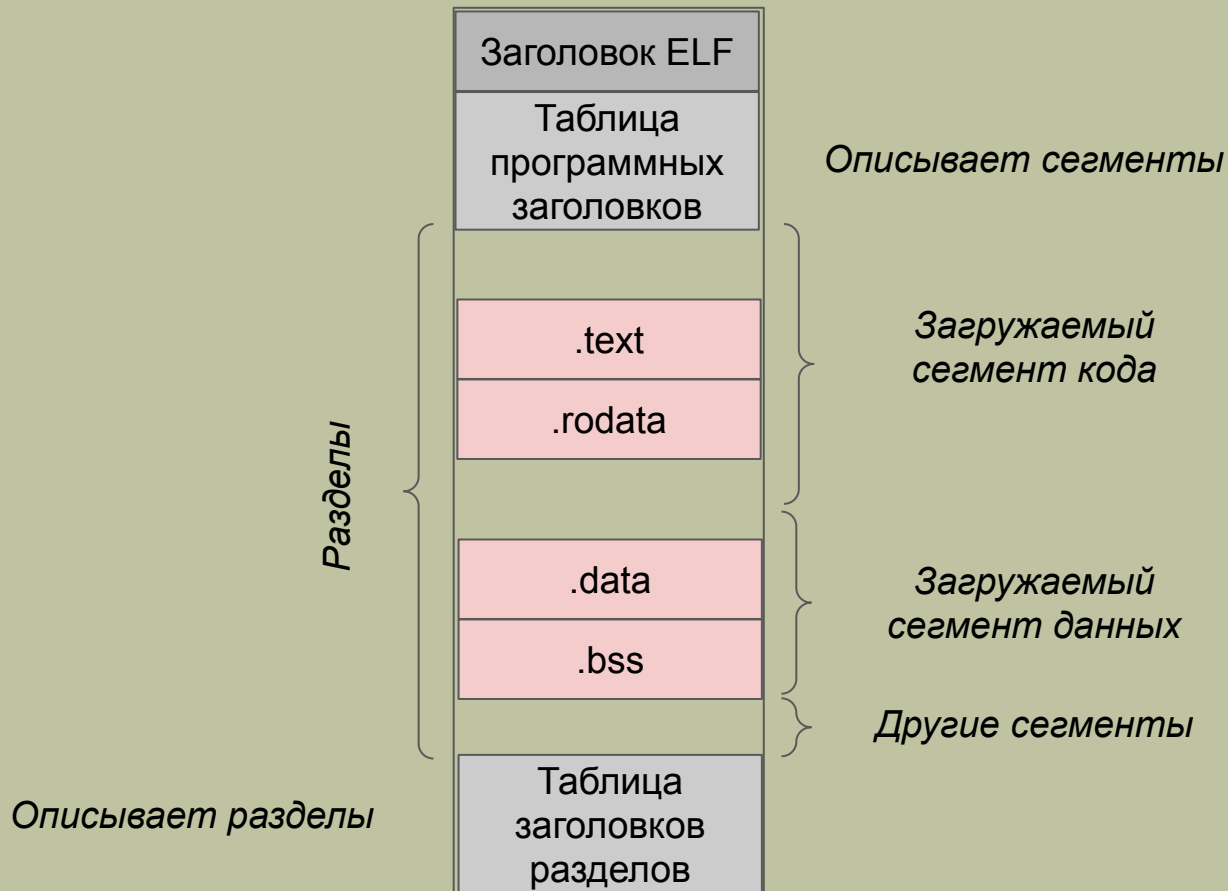
```
void* h=dlopen("liblab5.so", RTLD_LAZY);  
    fprintf(stdout, "%g\t%g\n", y,z);  
    g=(gun)dlsym(h, "gun1");  
    fprintf(stdout, "%g\t%d\t%g\n",  
        ((fun)dlsym(h, "fun1"))(0.1,123),  
        ((int (*)(int))dlsym(h, "fun2"))(8),  
        g());  
  
    dlclose(h);  
    return 0;  
}
```

```
~Lab5> gcc lab5d.c -L. -llab5 -ldl -o lab5d
~/Lab5> export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:.
~/Lab5> ./lab5d
12.4      2.87
12.3      64      3
```

```
~/Lab5> ldd lab5s
        linux-vdso.so.1 (0x00007ffdb8de8000)
        liblab5.so => not found
        libc.so.6 => /lib64/libc.so.6
(0x00007fc5dfc5f000)
        /lib64/ld-linux-x86-64.so.2
(0x00007fc5e001a000)
```

```
/Lab5> ldd lab5d
        linux-vdso.so.1 (0x00007fff037bc000)
        liblab5.so => not found
        libdl.so.2 => /lib64/libdl.so.2
(0x00007f1fb3b05000)
        libc.so.6 => /lib64/libc.so.6
(0x00007f1fb374a000)
        /lib64/ld-linux-x86-64.so.2
(0x00007f1fb3d09000)
```

Структура ELF файла



```
Lab5> readelf -h liblab5.so
```

```
  Magic:      7f 45 4c 46 02 01 01 00 00 00 00 00 00
00 00 00 00
```

```
Класс:                                ELF64
```

```
.....
Тип:                                DYN (Совм. исп. объектный файл)
```

```
Машина:                            Advanced Micro Devices X86-64
```

```
.....
Начало заголовков программы:64 (байт в файле)
```

```
Size of this header:                64 (bytes)
```

```
Size of program headers:            56 (bytes)
```

```
Number of program headers:           7
```

Lab5> readelf -l liblab5.so

Заголовки программы:

Тип	Смещ.	Вирт.адр	Физ.адр	
	Рзм.фйл	Рзм.пм	Флаги	Выравн
LOAD	0x0000000000000000	0x0000000000000000	0x0000000000000000	
	0x0000000000000078c	0x0000000000000078c	R E	0x200000
LOAD	0x00000000000000e30	0x0000000000200e30	0x0000000000200e30	
	0x00000000000000200	0x00000000000000208	RW	0x200000
DYNAMIC	0x00000000000000e40	0x0000000000200e40	0x0000000000200e40	
	0x000000000000001a0	0x000000000000001a0	RW	0x8
NOTE	0x000000000000001c8	0x000000000000001c8	0x000000000000001c8	
	0x00000000000000024	0x00000000000000024	R	0x4
GNU_EH_FRAME	0x00000000000000698	0x00000000000000698	0x00000000000000698	
	0x00000000000000034	0x00000000000000034	R	0x4
GNU_STACK	0x00000000000000000	0x00000000000000000	0x00000000000000000	
	0x00000000000000000	0x00000000000000000	RW	0x10
GNU_RELRO	0x00000000000000e30	0x0000000000200e30	0x0000000000200e30	
	0x000000000000001d0	0x000000000000001d0	R	0x1

~Lab5> dumpelf liblab5.so

```
.phdrs = {  
/* Program Header #0 0x40 */  
{  
    .p_type      = 1           , /* [PT_LOAD] */  
    .p_offset    = 0           , /* (bytes into file) */  
    .p_vaddr     = 0x0         , /* (virtual addr at runtime)  
*/  
    .p_paddr     = 0x0         , /* (physical addr at runtime)  
*/  
    .p_filesz    = 1932        , /* (bytes in file) */  
    .p_memsz     = 1932        , /* (bytes in mem at runtime)  
*/  
    .p_flags     = 0x5         , /* PF_R | PF_X */  
    .p_align     = 2097152     , /* (min mem alignment in  
bytes) */  
},
```



```
#include <elf.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int main(int argc, char** argv) {
    const char* elfFile=argv[1];
    Elf64_Ehdr header;
    Elf64_Phdr phheader;
    int i;
    FILE* file = fopen(elfFile, "rb");
```

```
fread(&header, sizeof(header), 1, file);  
fclose(file);
```

```
for(i=0;i<16;i++)  
    fprintf(stdout, "%x\t", header.e_ident[i]);  
fprintf(stdout, "\n");
```

```
fprintf(stdout, "type: %x\t machine: %x\n",  
        header.e_type, header.e_machine);  
fprintf(stdout, "e_phoff: %x\n",  
        header.e_phoff);  
fprintf(stdout, "e_phnum: %d\n",  
        header.e_phnum);
```

```
file = fopen(elfFile, "rb");

fseek(file, header.e_phoff, SEEK_SET);
for (i=0; i<header.e_phnum; i++) {
    if (i>0)
        fseek(file,
                header.e_phoff+header.e_phentsize*i,
                SEEK_SET);
    fread(&phheader, header.e_phentsize, 1,
          file);
    fprintf(stdout, "%x\t%x\t%x\t%x\n",
            phheader.p_type, phheader.p_offset,
            phheader.p_vaddr, phheader.p_paddr);
}
```

```
fprintf(stdout, "%x\t%x\t%x\t%x\n",  
        phheader.p_filesz, phheader.p_memsz,  
        phheader.p_flags, phheader.p_align);  
fprintf(stdout, "\n");  
}  
  
fclose(file);  
return 0;  
}
```

```
/Lab5> ./lab5-elf liblab5.so
```

7f	45	4c	46	2	1
----	----	----	----	---	---

1	0	0	0	0	0
---	---	---	---	---	---

0	0
---	---

0	0
---	---

```
type: 3 machine: 3e
```

```
e_phoff: 40
```

```
e_phnum: 7
```

1	0	0	0
----------	---	---	---

78c	78c	5	200000
-----	-----	----------	--------

1	e30	200e30	200e30
----------	-----	--------	--------

200	208	6	200000
-----	-----	----------	--------

2	e40	200e40	200e40
1a0	1a0	6	8

4	1c8	1c8	1c8
24	24	4	4

6474e550		698	698	698
34	34	4	4	

6474e551		0	0	0
0	0	6	10	

6474e552		e30	200e30	200e30
1d0	1d0	4	1	

..... •
p_flags This member holds a bit mask of
flags relevant to the segment:

PF_X	An executable segment.
PF_W	A writable segment.
PF_R	A readable segment.

.....