

## Задачи с lseek()

Задача 1. Какво ще се изпише на стандартния изход след изпълнението на

...

```
int main(int argc, const char *argv[])
{
    int fd = open("file", O_WRONLY|O_CREAT|O_TRUNC, 0644);
    ssize_t wbytes;
    wbytes = write (fd, "1234", 4);
    off_t offset;
    offset = lseek(fd, 0, SEEK_END);
    printf("size of file after 1st write is %d\n", (int)offset);
    offset = lseek(fd, 2, SEEK_END);
    printf("offset after 1st lseek is %d\n", (int)offset);
    offset = lseek(fd, 0, SEEK_END);
    printf("size of file after 1st lseek is %d\n", (int)offset);
    offset = lseek(fd, 5, SEEK_END);
    printf("offset after 2nd lseek is %d\n", (int)offset);
    wbytes = write(fd, "5678", 4);
    offset = lseek(fd, 0, SEEK_END);
    printf("size of file after 2nd write is %d\n", (int)offset);
    return 0;
}
```

Задача 2. Какво ще се изпише на стандартния изход и какво ще бъде съдържанието на файла file след изпълнението на

...

```
int main(int argc, const char *argv[])
{
    int fd = open("file", O_CREAT|O_TRUNC|O_RDWR, 0644);
    write(fd, "1234567", 7);
    off_t offset = lseek(fd, -3, SEEK_END);
    write(fd, "890", 2);
    char a[20];
    ssize_t r = read(fd, a,
                    -( lseek(fd, 0, SEEK_CUR) - lseek(fd, 0, SEEK_END) ));
    printf("%d\n",r);
    write(fd, "123", 4);
    return 0;
}
```

Задача 3. Да се напише програма, която извежда на стандартния изход последните N байта от съдържанието на файл. N и името на файла се подават като аргументи от командния ред. Приемаме, че съдържанието на файла няма да се променя по време на изпълнението на програмата.

Задача 4. Да се напише програма, която извежда на стандартния изход последните N реда от съдържанието на файл. N и името на файла се подават като аргументи от командния ред. Приемаме, че съдържанието на файла няма да се променя по време на изпълнението на програмата.