

1.Скачиваем библиотеку *boost* с помощью утилиты **wget**

```
$wget -P ~/Downloads https://sourceforge.net/projects/boost/files/boost/1.69.0/boost_1_69_0.tar.gz
```

2.Разархивируйте скачанный файл в директорию ~/boost\_1\_69\_0

```
$tar -C ~/ -xf ~/Downloads/boost_1_69_0.tar.gz
```

3.Подсчитываем количество файлов в директории ~/boost\_1\_69\_0 **не включая** вложенные директории.

```
tree boost_1_69_0/ -L 1
```

6 directories, 12 files

4.Подсчитываем количество файлов в директории ~/boost\_1\_69\_0 **включая** вложенные директории.

```
tree boost_1_69_0/ -L 1
```

5637 directories, 61191 files

5.Подсчитываем количество заголовочных файлов, файлов с расширением .cpp, сколько остальных файлов (не заголовочных и не .cpp)

```
tree boost_1_69_0/ -P "*.cpp"
```

5637 directories, 13774 files

```
tree boost_1_69_0/ -P "*.hpp"
```

5637 directories, 14912 files

```
tree boost_1_69_0/ -I "*.*pp|*.hpp"
```

5637 directories, 32505 files

6.Найдем полный путь до файла any.hpp внутри библиотеки *boost*.

```
[andrey@PC]~$ find boost_1_69_0/ -name "any.hpp"
boost_1_69_0/boost/fusion/algorithm/query/detail/any.hpp
boost_1_69_0/boost/fusion/algorithm/query/any.hpp
boost_1_69_0/boost/fusion/include/any.hpp
boost_1_69_0/boost/type_erasure/any.hpp
boost_1_69_0/boost/spirit/home/support/algorithm/any.hpp
boost_1_69_0/boost/xpressive/detail/utility/any.hpp
boost_1_69_0/boost/any.hpp
boost_1_69_0/boost/proto/detail/any.hpp
boost_1_69_0/boost/hana/fwd/any.hpp
boost_1_69_0/boost/hana/any.hpp
```

7. Выведем в консоль все файлы, где упоминается последовательность boost::asio.

```
[andrey@PC]~  
└─ $cd boost_1_69_0  
[andrey@PC]~/boost_1_69_0  
└─ $grep -rl "boost::asio"  
boostasio/detail/impl/buffer_sequence_adapter.hpp  
boostasio/detail/impl/descriptor_ops.hpp  
boostasio/detail/impl/dev_poll_reactor.hpp  
boostasio/detail/impl/dev_poll_reactor.hpp  
boostasio/detail/impl/epoll_reactor.hpp  
boostasio/detail/impl/eventfd_select_interrupter.hpp  
boostasio/detail/impl/handler_tracking.hpp  
boostasio/detail/impl/kqueue_reactor.hpp  
boostasio/detail/impl/pipe_select_interrupter.hpp  
boostasio/detail/impl posix_event.hpp  
boostasio/detail/impl posix_mutex.hpp  
boostasio/detail/impl posix_thread.hpp  
boostasio/detail/impl posix_tss_ptr.hpp  
boostasio/detail/impl/reactive_descriptor_service.hpp  
boostasio/detail/impl/reactive_serial_port_service.hpp  
boostasio/detail/impl/reactive_socket_service_base.hpp  
boostasio/detail/impl/resolver_service_base.hpp  
boostasio/detail/impl/scheduler.hpp  
boostasio/detail/impl/select_reactor.hpp
```

8. Скомпилируем *boost*.

```
./bootstrap.sh  
.b2
```

9. Перенесем все скомпилированные на предыдущем шаге статические библиотеки в директорию ~/boost-libs

```
mv stage/lib ~/boost-libs
```

10. Подсчитайте сколько занимает дискового пространства каждый файл в этой директории.

```
cd ~/boost-libs
```

```
[andrey@PC] -[~/boost-libs]
└─ $ls -s
total 51760
  4 libboost_atomic.a
  4 libboost_atomic.so
 16 libboost_atomic.so.1.69.0
244 libboost_chrono.a
  4 libboost_chrono.so
 60 libboost_chrono.so.1.69.0
152 libboost_container.a
  4 libboost_container.so
108 libboost_container.so.1.69.0
 20 libboost_context.a
```

11. Найдем *top10* самых "тяжёлых".

```
[andrey@PC] - [~/boost-libs]
└─ $ ls -sh -S
total 51M
4,6M libboost_wave.a
4,4M libboost_log.a
2,9M libboost_math_tr1.a
2,9M libboost_math_tr1l.a
2,8M libboost_regex.a
2,8M libboost_math_tr1f.a
2,7M libboost_log_setup.a
2,4M libboost_test_exec_monitor.a
2,4M libboost_unit_test_framework.a
2,1M libboost_locale.a
1,7M libboost_program_options.a
1,5M libboost_wave.so.1.69.0
1,3M libboost_log.so.1.69.0
1,2M libboost_serialization.a
1,2M libboost_regex.so.1.69.0
```