

# ***Boost C++ library***

## **What is Boost?**

The Boost C++ library is a collection of modern libraries based on the C++ standard. The source code is released under the [Boost Software License](#), which allows anyone to use, modify, and distribute the libraries for free. The libraries are platform independent and support most popular compilers, as well as many that are less well known. These libraries installed on the cluster are header only libraries. However, loading the libraries will load a specific version of GCC compiler.

Boost official documentation:

[Documentation](#)

## **Versions Available:**

The following versions are available on the cluster:

- Boost – 1.55.0
- Boost – 1.60.0
- Boost – 1.66.0
- Boost – 1.77.0

Researchers can install specific versions of boost library on local environment by following the installation on official website or by contacting HPC help line.

Libraries for python can be installed by using Conda- refer to conda documentation.

## **How to load Boost?**

To see the available version of Boost on the HPC, use the following commands:

```
module avail boost
```

To load Boost, use the following command,

```
module load boost/1.77.0
```

To verify if the module and dependencies are loaded correctly, use the following command.

```
module list
```

This should list all the software and dependencies that are loaded.

## How to use Boost?

To use boost, it must be included in the header section of your code,

This example is taken from boost official [documentation](#) :

```
#include <boost/lambda/lambda.hpp>
#include <iostream>
#include <iterator>
#include <algorithm>

int main()
{
    using namespace boost::lambda;
    typedef std::istream_iterator<int> in;

    std::for_each(
        in(std::cin), in(), std::cout << (_1 * 3) << " " );
}
```

The header section contains a command to use the boost library.

To compile this code, use g++ compiler which loads automatically when loading boost,

```
g++ test.cpp -o test
```

To link and compile the specific boost binary library, pass a flag at compile time. You can even link your own compiled library- follow the instructions on boost [documentation](#).

This example is taken from boost [documentation](#):

```
#include <boost/regex.hpp>
#include <iostream>
#include <string>

int main()
{
    std::string line;
    boost::regex pat( "^Subject: (Re: |Aw: )*(.*)" );

    while (std::cin)
    {
        std::getline(std::cin, line);
        boost::smatch matches;
        if (boost::regex_match(line, matches, pat))
            std::cout << matches[2] << std::endl;
    }
}
```

This example uses Boost regex library.

To compile use the following syntax:

```
g++ test.cpp -o test -lboost_regex
```

Follow the documentation for additional information.

***Where to find help?***

If you are stuck on some part or need help at any point, please contact OIT at the following address.

<https://ua-app01.ua.edu/researchComputingPortal/public/oitHelp>