

# Using vcpkg to couple Boost.Multiprecision in Visual Studio

## 1. Download and Install Boost:

- The package manager **vcpkg** is one of the easiest ways to get **Boost** on Windows.
- Open a Command Prompt or PowerShell.
- If you don't have vcpkg, you'll need to clone the vcpkg repository from GitHub. Follow the instructions here: <https://github.com/microsoft/vcpkg>
- Once vcpkg is set up, open a Command Prompt or PowerShell and run:
- `vcpkg install boost --triplet x64-windows`

## 2. Integrate Boost with Visual Studio:

- In the same command prompt where you installed boost using vcpkg, run this command:
- `vcpkg integrate install`
- This command configures Visual Studio to find the Boost headers and libraries.

## 3. Download the Multiprecision Package:

- To get the Multiprecision part of Boost, use vcpkg:
- `vcpkg install boost-multiprecision --triplet x64-windows`

#### 4. Brief Example:

- Open Visual Studio and create a new C++ console project.
- Replace the contents of your main .cpp file with the following:

```
#include <boost/multiprecision/cpp_int.hpp>
#include <boost/multiprecision/cpp_dec_float.hpp>
#include <iostream>
#include <math.h>
#include <iomanip>

using namespace boost::multiprecision;
using namespace std;

int main()
{
    // 2^100 Specify the number of decimal digits for cpp_dec_float.
    cpp_dec_float_100 n = pow(cpp_dec_float_100(2), 100);
    cout << fixed;
    cout << "Value of 2^100: " << n << "\n";

    // Calculate the factorial of 50 using Boost.Multiprecision
    cpp_int factorial = 1;
    for (int i = 1; i <= 50; ++i) {
        factorial *= i;
    }

    cout << "Factorial of 50 is: " << factorial << endl;
}
```

- Run the program. It will calculate and print  $2^{100}$  and the factorial of 50, a pair of very large numbers that standard C++ types can't handle.