Exercise – Hashing

Try and implement your own HashFunction namespace that contains Hash Functions.

Use the following as a starting point:

**HashFunction.h**:

#include <functional>

namespace HashFunction {

typedef std::function< unsigned int(const char\*, unsigned int)> HashFunc;

// implementation of a basic addition hash

unsigned int badHash( const char\* data, unsigned int length );

// ADD YOUR FUNCTIONS HERE

// a helper to access a default hash function

static HashFunc default = badHash;

}

**HashFunction.cpp**:

#include "HashFunction.h"

namespace HashFunction {

// implementation of a basic addition hash

unsigned int badHash( const char\* data, unsigned int length ) {

unsigned int hash = 0;

for (unsigned int i = 0 ; i < length ; ++i)

hash += data[ i ];

return hash;

}

}

Add additional Hash Functions to your namespace and set the default to an improved method rather than badHash. Use <http://www.partow.net/programming/hashfunctions/> for reference if needed.

To replace the default simply assign HashFunction::default to a function other than badHash.