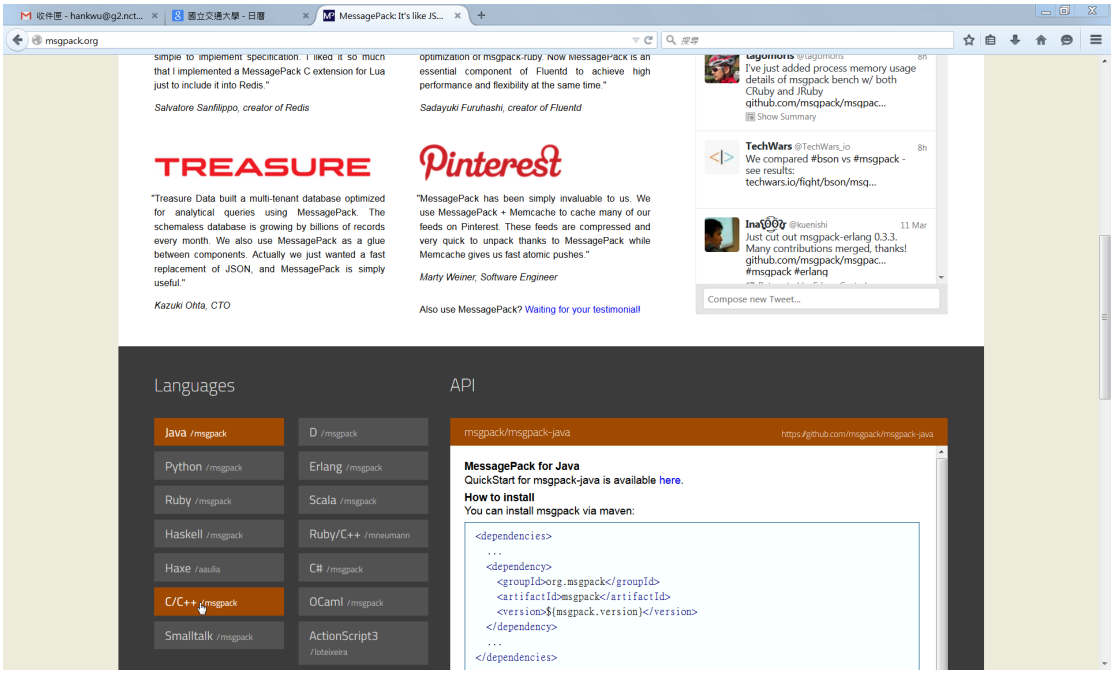
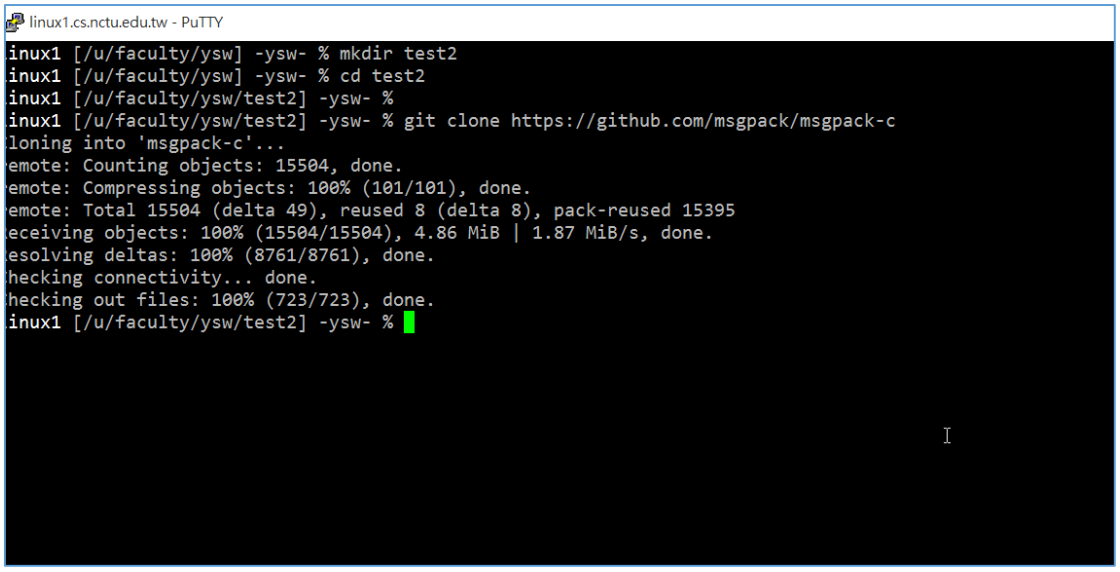
1. We use msgpack for packing input/output data. Its link is available from the official website

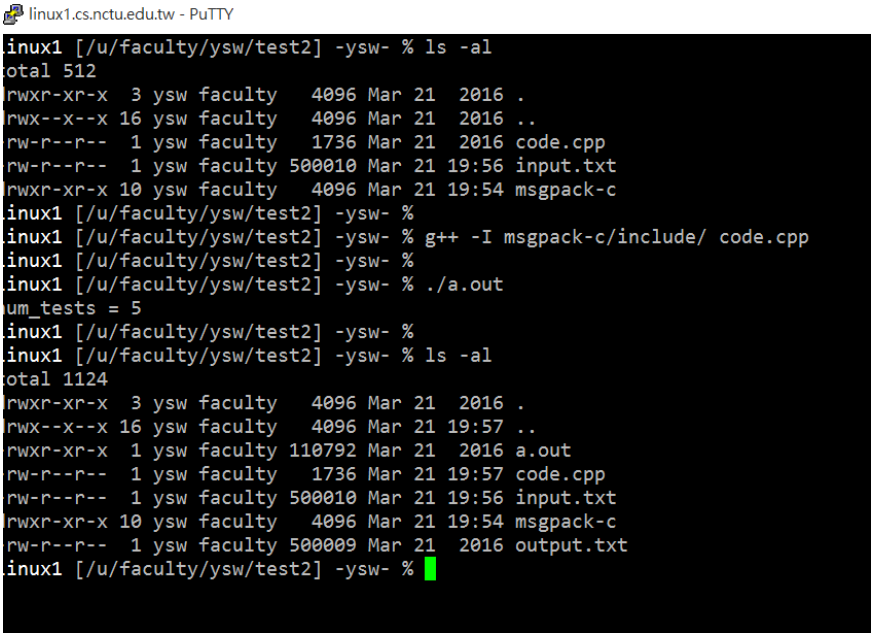


2. Download msgpack-c from github

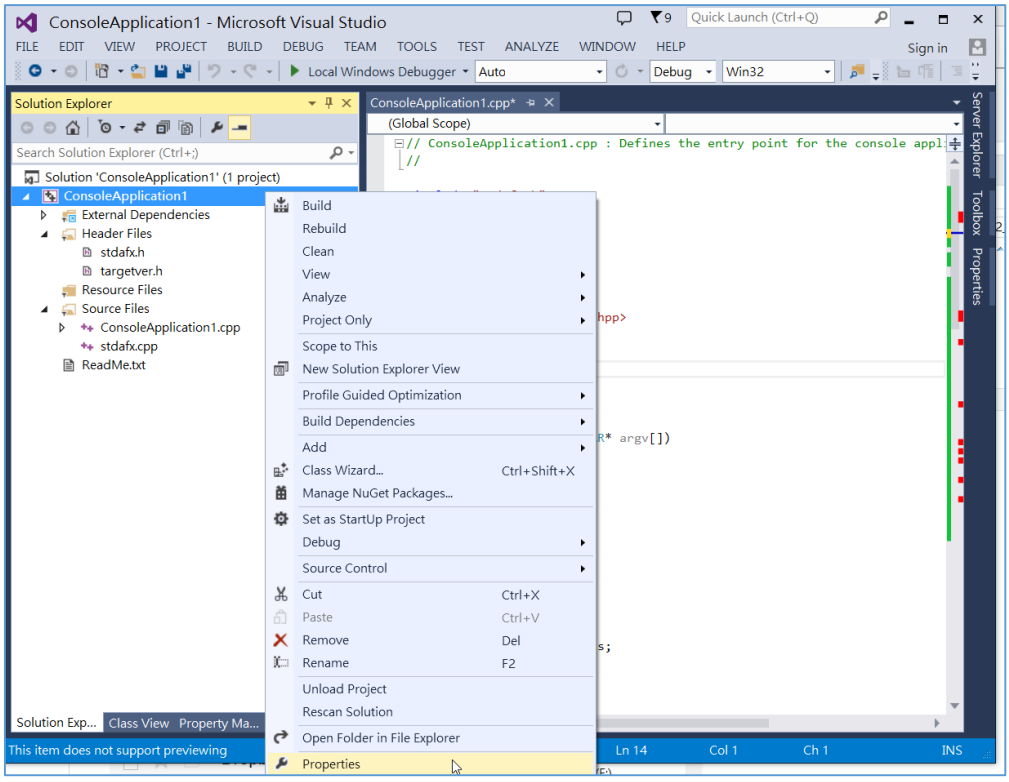
git clone https://github.com/msgpack/msgpack-c

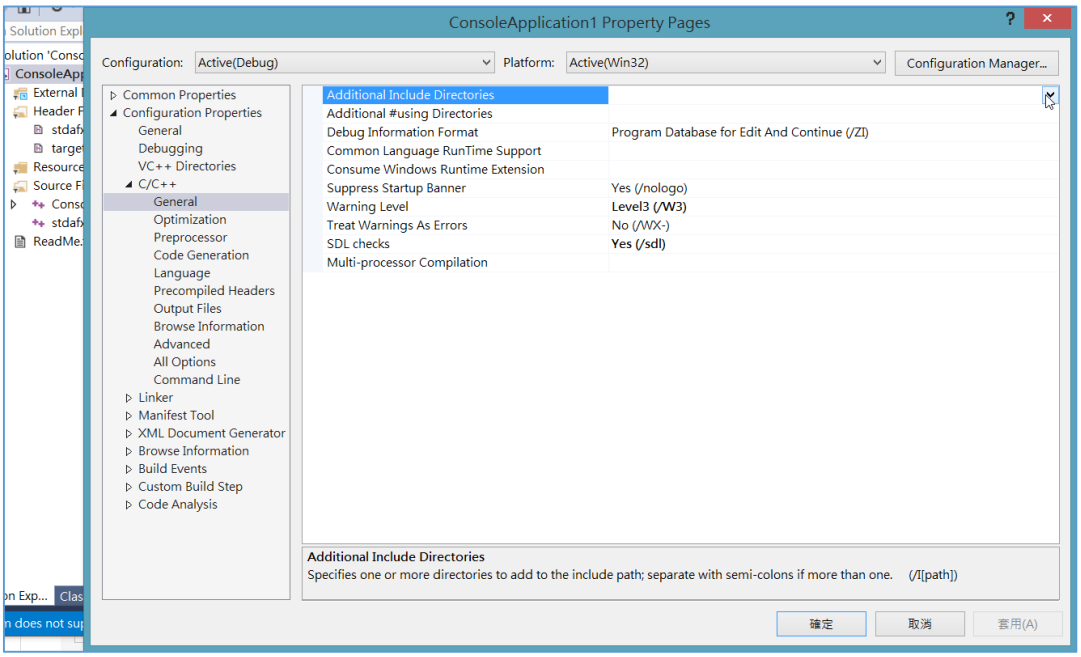


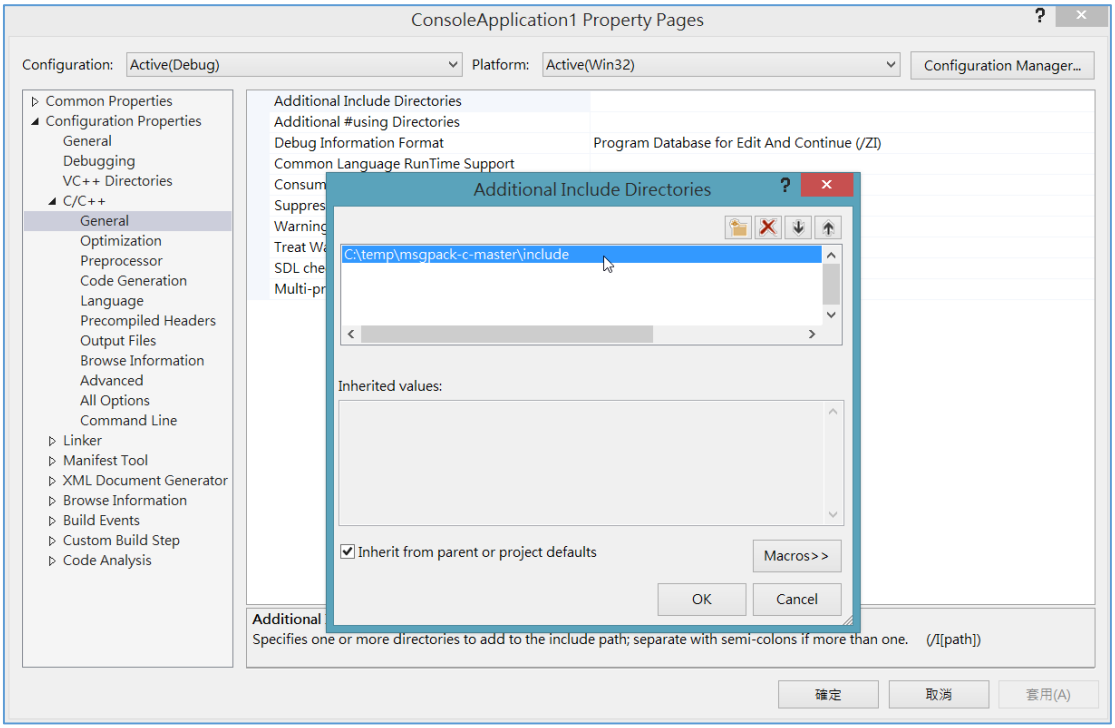
3. Compile your code

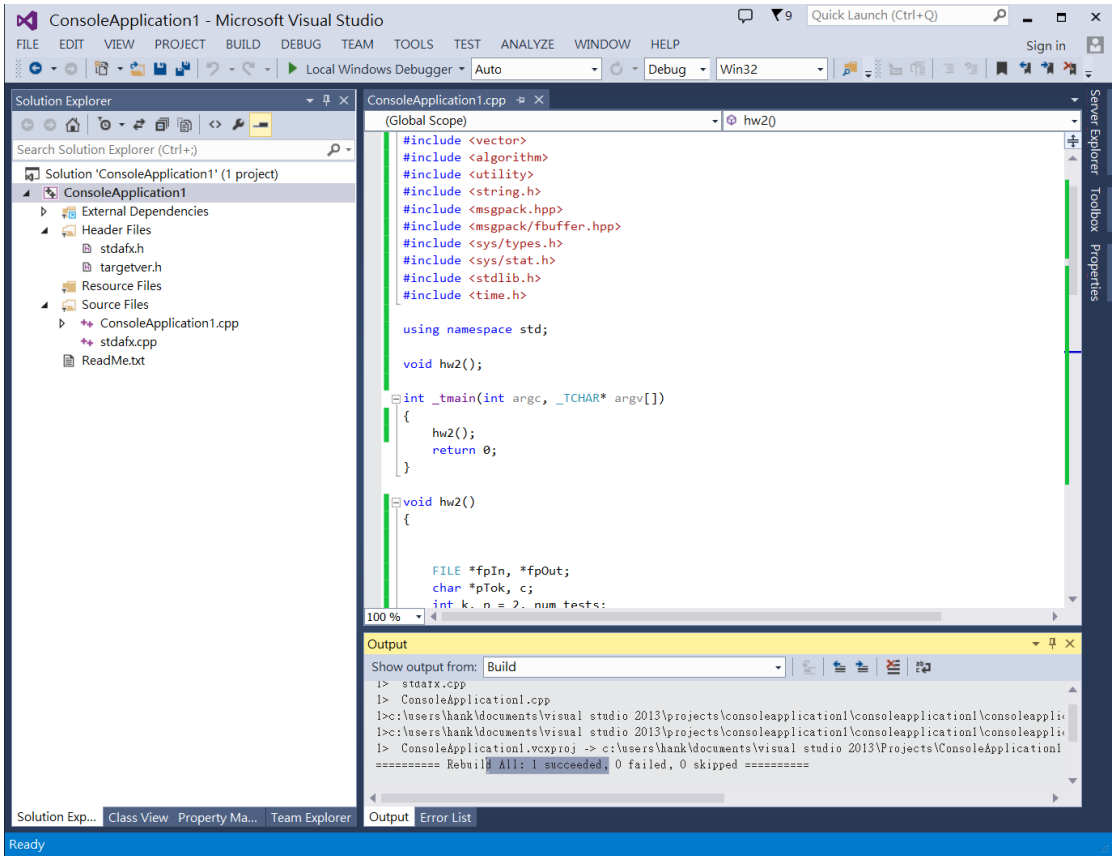


If you use MS Visual Studio on Windows, you need to add msgpack to the include path as follows




ConsoleApplication1 Property Pages
Configuration: Active(Debug)
Platform: Active(Win32)
Configuration Manager...
Common Properties
Configuration Properties
General
Debugging
VC++ Directories
C/C++
General
Optimization
Preprocessor
Code Generation
Language
Precompiled Headers
Output Files
Browse Information
Advanced
All Options
Command Line
Linker
Manifest Tool
XML Document Generator
Browse Information
Build Events
Custom Build Step
Code Analysis
Additional Include Directories
Additional #using Directories
Debug Information Format
Program Database for Edit And Continue (/ZI)
Common Language RunTime Support
Consume Windows Runtime Extension
Suppress Startup Banner
Yes (/nologo)
Warning Level
Level3 (/W3)
Treat Warnings As Errors
No (/WX-)
SDL checks
Yes (/sdl)
Multi-processor Compilation
Additional Include Directories
Specifies one or more directories to add to the include path; separate with semi-colons if more than one. (/I[path])
確定
取消
套用(A)


ConsoleApplication1 Property Pages
Configuration: Active(Debug)
Platform: Active(Win32)
Configuration Manager...
Additional Include Directories
C:\temp\msgpack-c-master\include
Inherited values:
Inherit from parent or project defaults
Macros>>
OK
Cancel
Specifies one or more directories to add to the include path; separate with semi-colons if more than one. (/I[path])
確定
取消
套用(A)


ConsoleApplication1 - Microsoft Visual Studio
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS TEST ANALYZE WINDOW HELP
Sign in
Local Windows Debugger
Auto
Debug
Win32
Quick Launch (Ctrl+Q)
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'ConsoleApplication1' (1 project)
ConsoleApplication1
External Dependencies
Header Files
stdafx.h
targetver.h
Resource Files
Source Files
ConsoleApplication1.cpp
stdafx.cpp
ReadMe.txt
Solution Exp...
Class View
Property Ma...
Team Explorer
ConsoleApplication1.cpp
(Global Scope)
hw2()
#include <vector>
#include <algorithm>
#include <utility>
#include <string.h>
#include <msgpack.hpp>
#include <msgpack/fbuffer.hpp>
#include <sys/types.h>
#include <sys/stat.h>
#include <stdlib.h>
#include <time.h>
using namespace std;
void hw2();
int _tmain(int argc, _TCHAR* argv[])
{
hw2();
return 0;
}
void hw2()
{
FILE *fpIn, *fpOut;
char *pTok, c;
int k, n = 2, num_tests;
100 %
Output
Show output from: Build
1> stdafx.cpp
1> ConsoleApplication1.cpp
1>c:\users\hank\documents\visual studio 2013\projects\consoleapplication1\consoleapplication1\consoleappli
1>c:\users\hank\documents\visual studio 2013\projects\consoleapplication1\consoleapplication1\consoleappli
1> ConsoleApplication1.vcxproj -> c:\users\hank\documents\visual studio 2013\Projects\ConsoleApplication1
========== Rebuild All: 1 succeeded, 0 failed, 0 skipped ==========
Output
Error List
Server Explorer
Toolbox
Properties
Ready

References

https://github.com/msgpack/msgpack-c/wiki/cpp_packer
https://github.com/msgpack/msgpack-c/wiki/cpp_unpacker

```cpp
#include <utility>
#include <string.h>
#include <msgpack.hpp>
#include <msgpack/fbuffer.hpp>
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <stdlib.h>
#include <time.h>

using namespace std;

int main()
{
    FILE *fpIn, *fpOut;
    char *pTok, c;
    int k, p=2, num_tests;
    vector<int> NUMBERS;
    msgpack::sbuffer sbuf;

    msgpack::unpacked result;
    struct stat st;
    size_t off = 0;
    stat("input.txt", &st);

    char* buf = new char[st.st_size];

    fpIn = fopen("input.txt", "rb");
    fread(buf, st.st_size, 1, fpIn );
    fclose(fpIn);

    msgpack::unpack(result, buf, st.st_size, off);
    result.get().convert(&num_tests);

    printf("num_tests = %d\n", num_tests);

    for ( k = 0; k < num_tests; k++) {
        msgpack::unpack(result, buf, st.st_size, off);
        result.get().convert(&NUMBERS);
        sort(NUMBERS.begin(), NUMBERS.end());
        msgpack::pack(&sbuf, NUMBERS);

    }

    assert(off == st.st_size);

    fpOut = fopen("output.txt", "wb");

    fwrite(sbuf.data(), sbuf.size(),1, fpOut);

    if(fpOut)
        fclose(fpOut);
```