C program

```
#include <stdio.h>
#include <windows.h>

void getCPUID(char* cpuid) {
   int cpuInfo[4] = { 0 };
   __cpuid(cpuInfo, 1);
   snprintf(cpuid, 13, "%08X", cpuInfo[3]);
}

void getHostname(char* hostname) {
   DWORD size = sizeof(hostname);
   GetComputerNameA(hostname, &size);
}

int main() {
   char cpuid[13];
   getCPUID(cpuid);

   char hostname[256];
   getHostname(hostname);

   printf("Hello World from %s CPU #%s\n", hostname, cpuid);

   return 0;
}
```

Bash programing

```
#!/bin/bash

# Get the CPU ID
cpuid=$(lscpu | awk '/^Serial/ { print $NF }')

# Get the hostname
hostname=$(hostname)

# Print the message
echo "Hello World from $hostname CPU #$cpuid"
```

Python programing

```
#!/user/bin/python3


import socket

import subprocess


def get_cpu_id():

    try:

        output = subprocess.check_output(['wmic', 'cpu', 'get', 'ProcessorId'],
universal_newlines=True)

        cpu_id = output.strip().split('\n')[-1]

        return cpu_id

    except:

        return "Unknown"


def get_hostname():

    return socket.gethostname()


# Get the CPU ID

cpu_id = get_cpu_id()


# Get the hostname

hostname = get_hostname()


# Print the message

print(f"Hello World from {hostname} CPU #{cpu_id}")
```

C++ programing

```
#include <iostream>
#include <cstring>

#ifdef _WIN32
   #include <intrin.h>
   #include <Windows.h>
#elif defined(__linux__) || defined(__unix__)
   #include <unistd.h>
   #include <sys/utsname.h>
   #include <fstream>
#endif

std::string getCPUID()
{
   std::string cpuid;
#ifdef _WIN32
   int32_t cpuinfo[4] = { 0 };
   __cpuid(cpuinfo, 1);
   char buffer[13];
   std::snprintf(buffer, sizeof(buffer), "%08X", cpuinfo[3]);
   cpuid = buffer;
#elif defined(__linux__) || defined(__unix__)
   std::ifstream cpuinfo("/proc/cpuinfo");
   std::string line;
```

```
    while (std::getline(cpuinfo, line))
    {
        if (line.find("processor") != std::string::npos)
        {
            std::size_t pos = line.find_last_of(":");
            if (pos != std::string::npos)
            {
                cpuid = line.substr(pos + 2);
                break;
            }
        }
    }
#endif

    return cpuid;
}

std::string getHostname()
{
    char buffer[256];
#ifdef _WIN32
    DWORD size = sizeof(buffer);
    if (GetComputerNameA(buffer, &size))
    {
        return buffer;
    }
#elif defined(__linux__) || defined(__unix__)
```

```
    if (gethostname(buffer, sizeof(buffer)) == 0)
    {
        return buffer;
    }
#endif

    return "Unknown";
}

int main()
{
    std::string cpuid = getCPUID();
    std::string hostname = getHostname();

    std::cout << "Hello World from " << hostname << " CPU #" << cpuid << std::endl;

    return 0;
}
```