多线程

WHY

同一个进程中需要同时完成不同的任务

WHAT

在一个进程中同时运行了多个线程，用来完成不同的工作，则称之为“多线程”，多个线程交替占用cpu资源，而非真正的并行执行。

HOW

```
mutex mtx;

void Greet(int id)
{
    lock_guard<mutex> lg(mtx);          // lock_guard : RAII
    cout << "hello there ! My name is " << id << endl;
}

int main()
{
    cout << "Starting threads ..." << endl;

    thread t1(Greet, 1);
    thread t2(Greet, 2);

    // 阻塞等待
    t1.join();
    t2.join();

    cout << "end" << endl;
}
```

```cpp
int main()
{
    cout << "Starting threads ..." << endl;
    vector<thread> v;

    for (int i = 0; i < 10; i++)
    {
        v.push_back(thread(Greet,i));
    }

    // 阻塞等待
    for (auto & t:v)
    {
        t.join();
    }

    cout << "end" << endl;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
end

C:\Users\Nico\OneDrive - officeapp.org\code> cmd /C "c:\Users\Nico\.vscode\extensions\ms-vscode.cpptools-1.12.4-win32-x64\debugAdapte
rs\bin\WindowsDebugLauncher.exe --stdin=Microsoft-MIEngine-In-1cgjfmjm.ram --stdout=Microsoft-MIEngine-Out-cggfbmsj.5sa --stderr=Micr
hello there ! My name is 0
hello there ! My name is 1
hello there ! My name is 2
hello there ! My name is 3
hello there ! My name is 4
hello there ! My name is 5
hello there ! My name is 6
hello there ! My name is 7
hello there ! My name is 8
hello there ! My name is 9
end
```