#include<stdio.h>

#include<stdlib.h>

#include<Windows.h>

#include<process.h>

#define MAX\_DOCUMENT 100

#define TOTAL\_STAFF 2

volatile LONG Document\_Number = 0;

INT Document\_Counter[MAX\_DOCUMENT];

CRITICAL\_SECTION CriticalSection;

UINT \_\_stdcall Staff(PVOID lp) {

INT Staff\_Number = \*(INT\*)lp;

while (Document\_Number < MAX\_DOCUMENT) {

EnterCriticalSection(&CriticalSection);

INT i = Document\_Number;

Sleep(rand() % 2);

Document\_Number = i + 1;

LeaveCriticalSection(&CriticalSection);

printf("Process document %2d by staff %d\n", i, Staff\_Number);

Document\_Counter[i]++;

}

return 0;

}

int main(int argc, TCHAR\* argv[]) {

HANDLE Staff\_Handles[TOTAL\_STAFF];

INT Staff\_Numbers[TOTAL\_STAFF];

srand(GetTickCount());

ZeroMemory(Document\_Counter, sizeof(Document\_Counter));

InitializeCriticalSection(&CriticalSection);

for (int i = 0; i < TOTAL\_STAFF; i++) {

Staff\_Numbers[i] = i;

Staff\_Handles[i] = (HANDLE)\_beginthreadex(NULL, 0, Staff, &Staff\_Numbers[i], CREATE\_SUSPENDED, NULL);

}

printf("Stuffs are ready\n");

for (int i = 0; i < TOTAL\_STAFF; i++) {

ResumeThread(Staff\_Handles[i]);

}

WaitForMultipleObjects(TOTAL\_STAFF, Staff\_Handles, TRUE, INFINITE);

for (int i = 0; i < TOTAL\_STAFF; i++) {

CloseHandle(Staff\_Handles[i]);

}

for (int i = 0; i < MAX\_DOCUMENT; i++) {

printf("Document %2d processed by %d staffs\n", i, Document\_Counter[i]);

}

DeleteCriticalSection(&CriticalSection);

system("pause");

return 0;

}