#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];

HANDLE Process1\_Done, Process2\_Ready, Process2\_GetDoc;

UINT \_\_stdcall Process1(PVOID lp) {

while (Document\_Number < MAX\_DOCUMENT) {

printf("Process document %2d by staff 1\n", Document\_Number);

// process document

printf("Finish1 document %2d by staff 1\n", Document\_Number);

Document\_Counter[Document\_Number]++;

SetEvent(Process1\_Done);

WaitForSingleObject(Process2\_Ready, INFINITE);

WaitForSingleObject(Process2\_GetDoc, INFINITE);

Document\_Number++; // get next document

}

return 0;

}

UINT \_\_stdcall Process2(PVOID lp) {

SetEvent(Process2\_Ready);

while (Document\_Number < MAX\_DOCUMENT) {

WaitForSingleObject(Process1\_Done, INFINITE);

INT i = Document\_Number;

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

SetEvent(Process2\_GetDoc);

// process document

printf("Finish2 document %2d by staff 2\n", i);

Document\_Counter[i]++;

SetEvent(Process2\_Ready);

}

return 0;

}

INT main(INT argc, PCHAR argv[]) {

HANDLE Staff\_Handles[TOTAL\_STAFF];

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

Process1\_Done = CreateEvent(NULL, FALSE, FALSE, NULL);

Process2\_Ready = CreateEvent(NULL, FALSE, FALSE, NULL);

Process2\_GetDoc = CreateEvent(NULL, FALSE, FALSE, NULL);

Staff\_Handles[0] = (HANDLE)\_beginthreadex(NULL, 0, Process1, NULL, 0, NULL);

Staff\_Handles[1] = (HANDLE)\_beginthreadex(NULL, 0, Process2, NULL, 0, NULL);

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

CloseHandle(Process1\_Done);

CloseHandle(Process2\_Ready);

CloseHandle(Process2\_GetDoc);

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

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

}

system("pause");

return 0;

}