

写入共享内存

```
1 #include <iostream>
2 #include <windows.h>
3 #include "share_mem.h"
4 using namespace std;
5 #define BUF_SIZE 2048
6
7 int main()
8 {
9
10     //创建共享文件句柄
11     HANDLE hMapFile = CreateFileMapping(
12         INVALID_HANDLE_VALUE,    //物理文件句柄
13         NULL,
14         PAGE_READWRITE, //可读可写
15         0,
16         BUF_SIZE,    //低位文件大小
17         L"NameOfMappingObject" //共享内存的名字
18     );
19
20     //共享内存映射
21
22     char* pBuf = (char*)MapViewOfFile(
23         hMapFile,
24         FILE_MAP_ALL_ACCESS,
25         0,
26         0,
27         BUF_SIZE
28     );
29
30     struct shm_data* share = (struct shm_data*)pBuf;
31     share->flag = 0;
32
33     while (1)
34     {
35
36         if (share->flag == 0)
37         {
```

```

39         scanf_s("%s", share->c_data, BUF_SIZE);
40         share->flag = 1;
41         if (!strcmp(share->c_data, "end", 3))
42         {
43             break;
44         }
45
46     }
47     else
48     {
49         Sleep(10);
50     }
51
52
53 }
54 UnmapViewOfFile(pBuf);
55 CloseHandle(hMapFile);
56
57 return 0;
58
59 }

```

读取共享内存

```

1  #include <iostream>
2  #include <Windows.h>
3  #include "share_memo.h"
4
5  using namespace std;
6
7
8  #define BUF_SIZE 2048
9
10
11 int main()
12 {
13     HANDLE hMapFile = CreateFileMapping(

```

```
14     INVALID_HANDLE_VALUE,    // 物理文件句柄
15     NULL,    // 默认安全级别
16     PAGE_READWRITE, // 可读可写
17     0,
18     BUF_SIZE,    //低位文件大小
19     L"NameOfMappingObject"
20 );
21
22 char* pBuf = (char*)MapViewOfFile(
23     hMapFile,
24     FILE_MAP_ALL_ACCESS,
25     0,
26     0,
27     BUF_SIZE);
28
29
30 struct shm_data* share = (struct shm_data*)pBuf;
31 while (1)
32 {
33
34     if (share->flag == 1)
35     {
36         printf("%s\n", share->c_data);
37         if (!strncmp("end", share->c_data, 3))
38         {
39             break;
40         }
41         memset(share->c_data, 0, BUFSIZ);
42         share->flag = 0;
43     }
44     else
45     {
46         Sleep(1000);
47         printf("Wait....\n");
48     }
49
50
51
52 }
53
54 UnmapViewOfFile(pBuf);
55 CloseHandle(hMapFile);
56 return 0;
```

57	
58	
59	
60	}