CS 332011 - Operating Systems - Project 1

Generated by Doxygen 1.9.1

Chapter 1

Programming Assignment 1 - Producer & Consumer Problem

1.0.1 Author: Elliott Hager

1.0.2 Fall 2023 Semester

1.0.3 CS-33211: Operating Systems

1.0.4 Dr. Qiang Guan

1.0.5 Kent State University

1.0.5.1 Description (Provided from the assignment instructions)

The producer generates items and puts items onto the table. The consumer will pick up items. The table can only hold two items at the same time. When the table is complete, the producer will wait. When there are no items, the consumer will wait. We use semaphores to synchronize producer and consumer. Mutual exclusion should be considered. We use threads in the producer program and consumer program. Shared memory is used for the “table”.

1.0.6 Implementation

Both the producer and consumer have their own processes, with a thread in each created and pointed to a function of the same name (i.e. consumer and producer respectively). With each thread running their own functions, a shared memory buffer with a capacity to hold two items is shared between the producer and consumer. To help prevent deadlock and other data hazards when it comes to a shared buffer, a semaphore is used to indicate use of the buffer. The number '1' indicates the buffer is currently in use, while the number '0' indicates the buffer is available for use.s

1.0.7 Compilation

Both the producer and consumer are septate files and meant to be septate processes. There are three options within the make file. The first two being consumer and producer respectively. These compiler the producer and consumer. The last option is clean, which cleans any object (.o) files as well as the compiled program files to be executed.

Running the consumer and/or producer separately will not be of benefit as each relies on each other to run concurrently.

1.0.7.1 Consumer:

The following make command will compile the consumer files for execution
```
make consumer
```

1.0.8 Producer:

The following make command will compile the producer files for execution
```
make producer
```

1.0.9 Run:

The following make command will compile the producer, consumer, and run both with the shared memory key of shmfile
```
make run
```

1.0.10 Clean:

The following make command will remove the compiled and executable program files
```
make clean
```

1.0.11 Run Instructions:

To run both of the producer and consumer at the same time, for Linux and Unix systems, there are two options.

1.0.11.1 1.) Manually/Command

```
./out/consumer shmfile & ./out/producer shmfile
```

1.0.12 2.) Using Make (Preferred)

Using the `make run` command will compile and run both the producer and consumer. This is the preferred method of compilation and running the project.

1.0.13 Libraries & Tech Stack

- C++
- Make
- G++
- PThread Library
- SharedMem 3 Library

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 4

Class Documentation

4.1 sharedMem Struct Reference

The shared memory buffer with semaphores.

`#include <membuff.hpp>`

Public Attributes

- int **table** [TABLE_SIZE]
- int **in**
- int **out**
- sem_t **mutex**
- sem_t **empty**
- sem_t **full**

4.1.1 Detailed Description

The shared memory buffer with semaphores.

The documentation for this struct was generated from the following file:

- include/membuff.hpp

Chapter 5

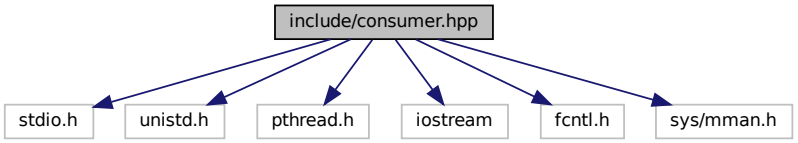
File Documentation

5.1 include/consumer.hpp File Reference

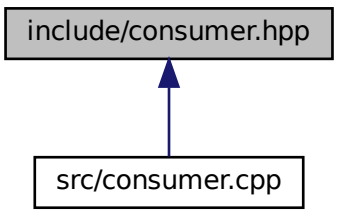
The consumer header for the producer/consumer problem.

```
#include <stdio.h>
#include <unistd.h>
#include <pthread.h>
#include <iostream>
#include <fcntl.h>
#include <sys/mman.h>
```

Include dependency graph for consumer.hpp:



This graph shows which files directly or indirectly include this file:



Functions

- int **main** (int, char ∗)

5.1.1 Detailed Description

The consumer header for the producer/consumer problem.

Author

Elliott Hager

Date

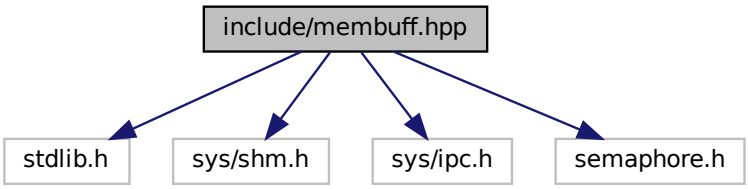
2023-11-05

5.2 include/membuff.hpp File Reference

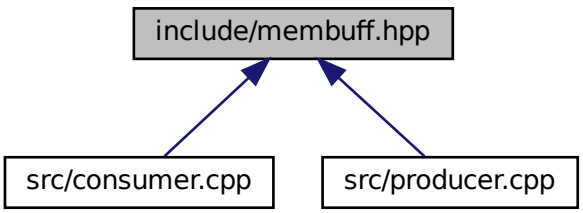
The shared memory buffer structure for the producer/consumer probelm.

```
#include <stdlib.h>
#include <sys/shm.h>
#include <sys/ipc.h>
#include <semaphore.h>
```

Include dependency graph for membuff.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- struct sharedMem

  The shared memory buffer with semaphores.

Macros

- #define TABLE_SIZE 2

  The size of the buffer.

5.2.1 Detailed Description

The shared memory buffer structure for the producer/consumer probelm.

Author

Elliott Hager

Date

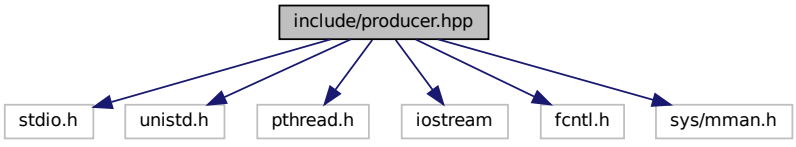
2023-11-05

5.3 include/producer.hpp File Reference

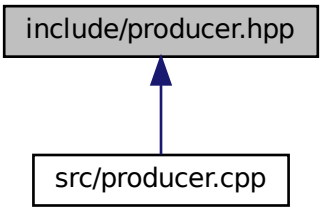
The producer header for the producer/consumer problem.

```
#include <stdio.h>
#include <unistd.h>
#include <pthread.h>
#include <iostream>
#include <fcntl.h>
#include <sys/mman.h>
```

Include dependency graph for producer.hpp:



This graph shows which files directly or indirectly include this file:



Functions

- int **main** (int, char ∗)

5.3.1 Detailed Description

The producer header for the producer/consumer problem.

Author

Elliott Hager

Date

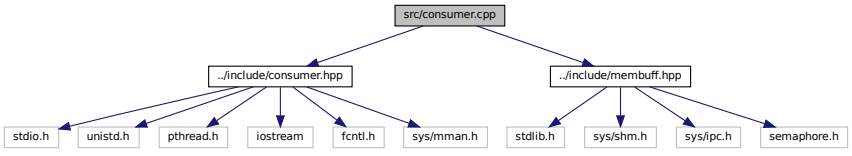
2023-11-05

5.4 src/consumer.cpp File Reference

The consumer in the producer/consumer problem.

```
#include "../include/consumer.hpp"
#include "../include/membuff.hpp"
```

Include dependency graph for consumer.cpp:



Functions

- int main (int argc, char ∗argv[ ])

  The main function for the consumer.

5.4.1 Detailed Description

The consumer in the producer/consumer problem.

Author

Elliott Hager

Date

2023-11-05

5.4.2 Function Documentation

5.4.2.1 main()

```
int main (
            int argc,
            char * argv[] )
```

The main function for the consumer.

Parameters

argc	The number of arguments fed into the program
argv	Array to hold the command line fed arguments, used for the shared memory file name

Returns

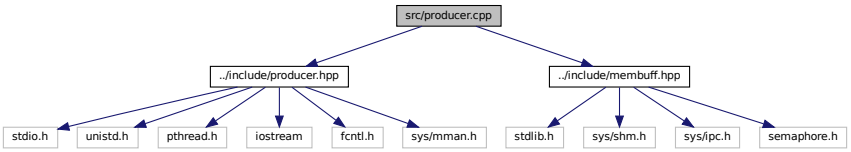
int Exit status of the program

5.5 src/producer.cpp File Reference

The producer in the producer and consumer problem.

```
#include "../include/producer.hpp"
#include "../include/membuff.hpp"
```

Include dependency graph for producer.cpp:



Functions

- int main (int argc, char ∗argv[ ])

  The main function for the producer.

5.5.1 Detailed Description

The producer in the producer and consumer problem.

Author

Elliott Hager

Date

2023-11-05

5.5.2 Function Documentation

5.5.2.1 main()

```
int main (
            int argc,
            char * argv[] )
```

The main function for the producer.

Parameters

argc	The number of arguments fed into the program
argv	Array to hold the command line fed arguments, used for the shared memory file name

Returns

int Exit status of the program

Index