Fatih BAS 150117048

Alperen KOKER 150117035

Süleyman Ahmet SONMEZ 150117010

CSE 3033 – OPERATING SYSTEMS

Programming Assignment #3

In this program we have 3 part.

Our program main component is:

Publisher thread: is publish books according to max publish count and put into buffer according to thread type.

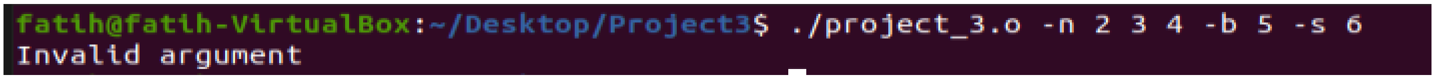
Packager thread: If buffers is not empty, this thread take book randomly and packets.

First Part:

Firstly, our program reads all arguments and checks arguments are complete and arguments' validity. If they are not complete and valid, our program print message on console and exit system.

For example:

Last argument is missing in -s option.



```
fatih@fatih-VirtualBox:~/Desktop/Project3$ ./project_3.o -n 2 3 4 -b 5 -s 6
Invalid argument
```

If they are complete and valid, our program initialize variable which coming argument, mutexes and semaphore.

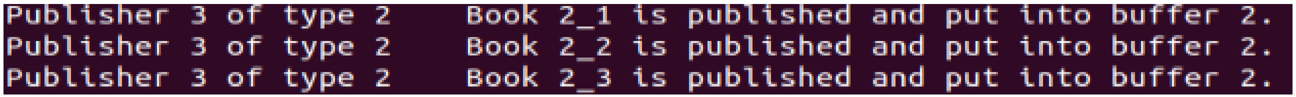
Second Part:

Firstly, our program creates buffers for each publisher type and put into a linked list which is global variable. After, initialize struct variables to send as parameter and prepare empty packages for each packager before creates threads.

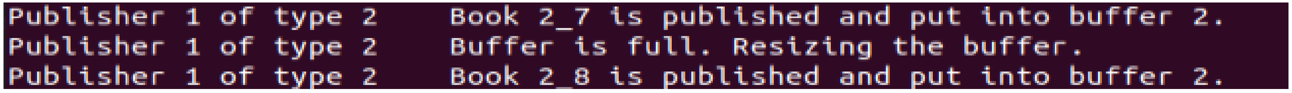
Publisher Threads Working Logic:

Firstly, our threads take parameter. After our program controls count of books published by publisher threads. If count of books is max for each thread, The count of publisher threads decreases in the system and publisher thread's function finishes. If count of books is not max for each thread, one thread enters and locks the system. After this the publisher prepares one book and puts into buffer according to type. If buffer size is full, thread wants to resize buffer and buffer is double size. When publisher publish and put book into a buffer, publisher call sem_post method for packager threads which in the system and unlock the mutex for other publisher threads which in the system.

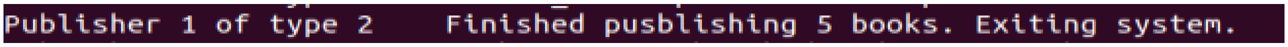
For example:



```
Publisher 3 of type 2    Book 2_1 is published and put into buffer 2.
Publisher 3 of type 2    Book 2_2 is published and put into buffer 2.
Publisher 3 of type 2    Book 2_3 is published and put into buffer 2.
```



```
Publisher 1 of type 2    Book 2_7 is published and put into buffer 2.
Publisher 1 of type 2    Buffer is full. Resizing the buffer.
Publisher 1 of type 2    Book 2_8 is published and put into buffer 2.
```

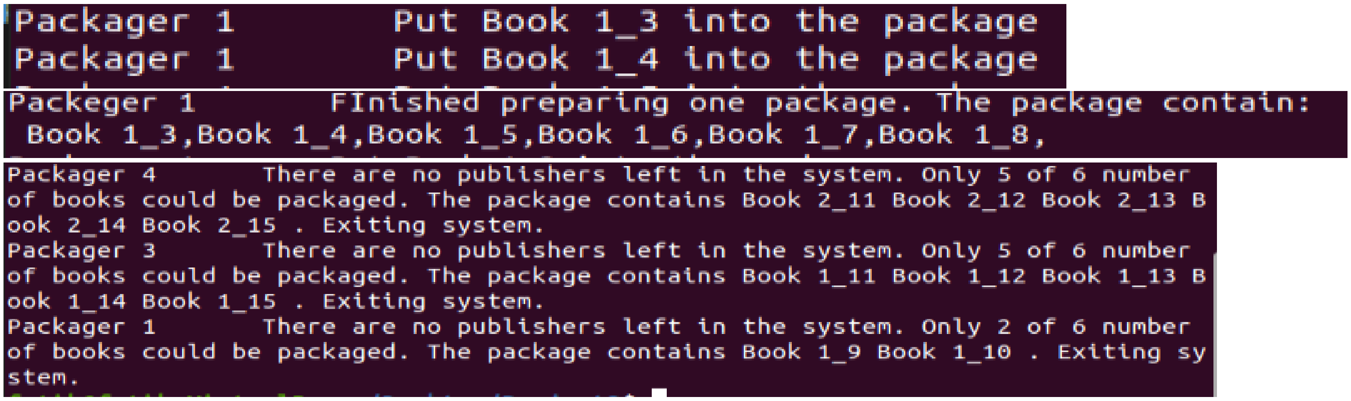


```
Publisher 1 of type 2    Finished pusblishing 5 books. Exiting system.
```

Packager Threads Working Logic:

Firstly, our threads take parameter. After our program controls total number of books to pack. If total number of books to pack is max, all packager threads print information about package's last status which count of books in package more than zero and all packager threads exit the system. If total number of books to pack is not max, one packager thread calls sem_wait method for any book in the system. If any book in the system, packager enters and locks the system. After this packager selects one buffer randomly but if packager's buffer is empty and this buffer's publishers/publisher in the system, this packager can not pack book until publisher publishes book. If packager's buffer is not empty, packager select one book in the buffer and put into package. If packager's package's size is full, packager print info about package and prepare new package for pack book. Lastly packager unlocks the mutexes for other packagers and publishers which in the system.

For example:



```
Packager 1       Put Book 1_3 into the package
Packager 1       Put Book 1_4 into the package
Packeger 1       FInished preparing one package. The package contain:
 Book 1_3,Book 1_4,Book 1_5,Book 1_6,Book 1_7,Book 1_8,
Packager 4       There are no publishers left in the system. Only 5 of 6 number
of books could be packaged. The package contains Book 2_11 Book 2_12 Book 2_13 B
ook 2_14 Book 2_15 . Exiting system.
Packager 3       There are no publishers left in the system. Only 5 of 6 number
of books could be packaged. The package contains Book 1_11 Book 1_12 Book 1_13 B
ook 1_14 Book 1_15 . Exiting system.
Packager 1       There are no publishers left in the system. Only 2 of 6 number
of books could be packaged. The package contains Book 1_9 Book 1_10 . Exiting sy
stem.
```

Third Part:

Our main program waits all of the threads with join methods. If all of the threads exit the system, our main program destroy the all mutex and semaphore and exit.