**Does your program output any garbage? If yes then why**The program shows unexpected behavior when printing values of produced and consumed. The program is designed to make producers increment the global resource and the consumer to read each resource. But as we see on the output below, the two threads created by the main process do not work in harmony. The buffer is shared among the processes so read and write operations occur from some memory source. The producer at times runs for several iterations and is then preempted by the operating system, halfway printing the output at times. The consumer which is supposed to print all values of the shared global resource instead runs several times at a go and prints the same value over and over. The garbage values appear thus on command shell mainly due to no synchronization among the two processes. The shared resource is supposed to be protected and access to it should be given to only one function at a time. Since that is not done here, the operations are not completed at times which can create inconsistency. The random preemption leads to garbage values on the output line.

**Are all the produced values getting consumed? Check your prog for 20 count**

As seen in the output below, all the values are not getting printed or consumed. The consumed function is supposed to read every change in the global n variable but what happens is its given control only when producer is run several times over so consumer ends up missing many of the value changes. Now consumer itself is run several times even though the value is not changing, thus leading to the same value being consumed over and over.

Functions:

• void consumer(int count);

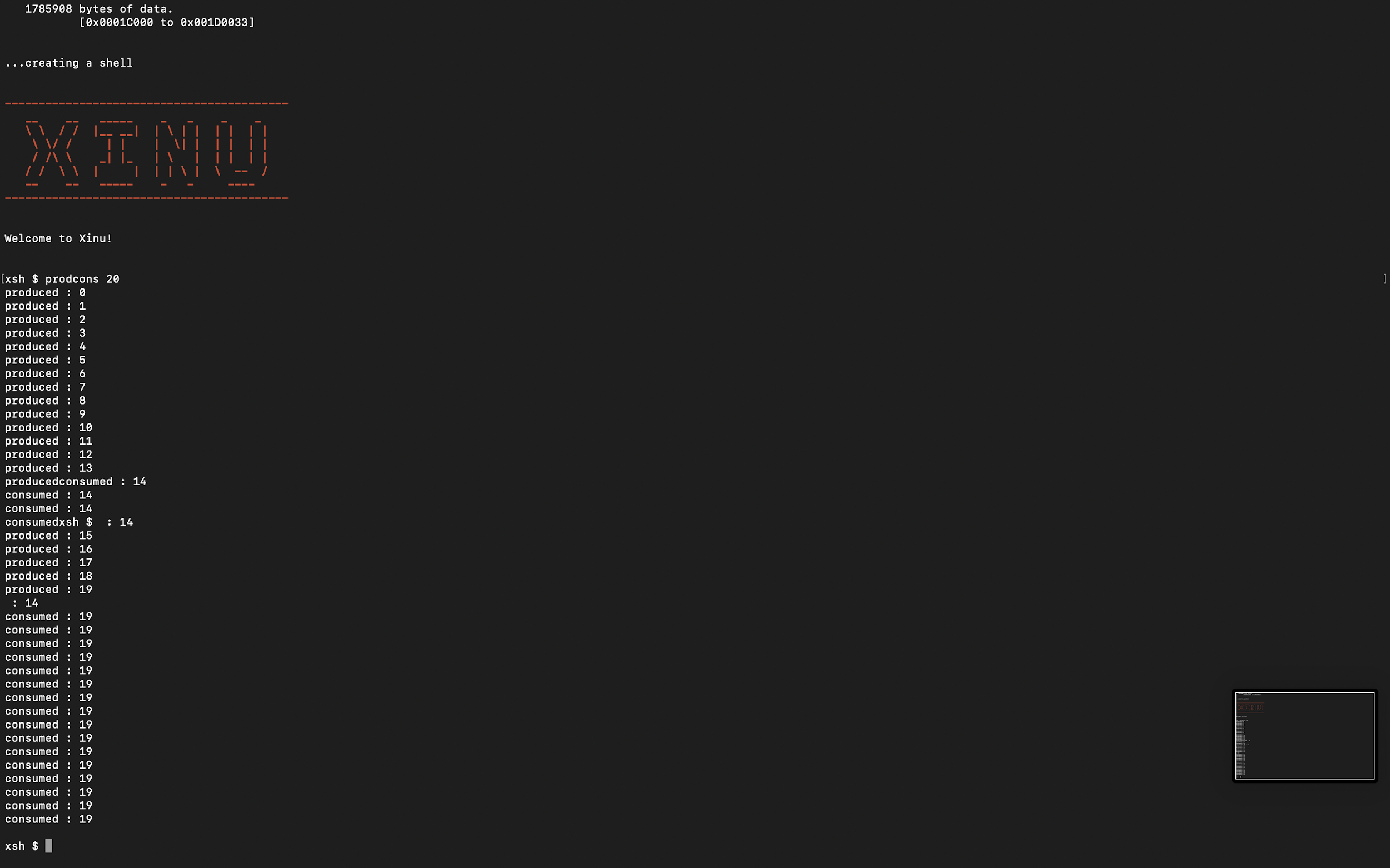
• void producer(int count);

C Standard Library

• int isdigit( int arg ); A check whether arg is a number

• int atoi(const char \*str); converts a string to int

OUTPUT ON BELOW, PLS SCROLL

****