**EXPERIMENT - 7**

**Aim:** Write a program for the implementation of Producer Consumer problem.

**Description:**

Producer consumer problem is a synchronization problem. There is a fixed size buffer where the producer produces items and that is consumed by a consumer process. One solution to the producer consumer problem uses shared memory. To allow producer and consumer processes to run concurrently, there must be available a buffer of items that can be filled by the producer and emptied by the consumer. This buffer will reside in a region of memory that is shared by the producer and consumer processes. The producer and consumer must be synchronized, so that the consumer does not try to consume an item that has not yet been produced.

**Source Code:**

#include<bits/stdc++.h>

using namespace std;

int main() {

    int buffer[10], bufsize, in, out, produce, consume, choice=0;

in = 0;

    out = 0;

    bufsize = 10;

    while(choice != 3) {

        cout<<"\n1.Produce\t2.Consume\t3.Exit";

        cout<<"Enter your choice: ";

        cin>>choice;

        switch(choice) {

            case 1: if((in+1)%bufsize == out)

                        cout<<"\nBuffer is full";

                    else {

                        cout<<"\nEnter the value: ";

                        cin>>produce;

                        buffer[in] = produce;

                        in = (in+1)%bufsize;

                    }

                    break;

            case 2: if(in == out)

                        cout<<"\nBuffer is Empty";

                    else {

                        consume == buffer[out];

                        cout<<"\nThe consumed value is "<<consume;

                        out = (out+1)%bufsize;

                    }

                    break;

        }

    }

    return 0;

}

**Output:**

