Operating Systems
Assignment no.03

Question no.04:

```
Sem_t empty=max, full=0;
class Stack {
private:
int* a; // array for stack
int max; // max size of array
int top; // stack top
public:
Stack(int m) {
a = new int[m]; max = m; top = 0;
}
void push(int x) {
sem_wait(empty);  //we will wait for a slot to get empty is there is no empty slot. Until
then the process of push will be put on hold.
a[top] = x;
++top;
Sem_post(full);   //signaling the full semaphore to indicate that a slot is filled.
}
int pop() {
sem_wait(full);  //wait until a slot gets filled so that it can be popped
int tmp = top;
--top;
Sem_post(empty); //signaling the empty semaphore to tell that a slot is empty so that new
element can be added.
return a[tmp];
}
};
```

Question no.05:

```
Sem_t batter=1, bowler=1;
Sem_t batsemaphore=0, bowlsemaphore=0;
Batsman()
{
Wait(batter);
Sem_post(bowlsemaphore);
Wait(batsemaphore);
Practice();
Signal(batter);

}

bowler()
{
Wait(bowler);
Sem_post(batsemaphore);
Wait(bowlsemaphore);
Practice();
Signal(bowler);


}
```