Instructions:
Please answer all coding questions in C. Clarity will make it easier for us to see your intent.

---

1. Consider a dispatch function that takes an array of function pointers, and an array of arguments of any type and then invokes each function pointer on its corresponding argument. Each referenced function can return NULL, or it can return a non-null value. The dispatch function should cycle indefinitely through one of the invoked routines returns null

Expand on this skeleton code to implement the function (lines with XX's on them are clearly incomplete, as is the blank space.

```
void dispatch(



        )
{









}
```

2. Show the code for a thread-safe stack, where the stack implements a PUSH and POP operation of any arbitrary value (void*). Push should return null if the stack is full. Pop should return NULL on underflow. A push of a null item is a noop. Please use locks.

```
// data structure declarations here



void *pop()
{
```

```
}


void push(void *element)
{




}
```

3. As in 2, but push should wait while the stack is full and pop should wait while the stack is empty. Use mutexes (monitors) and condition variables.

```
// data structure declarations here



void *pop()
{




}
```

4. Prove that SJF is optimal w.r.t. average waiting time.