```
import sys, socket, threading

class ClientHandler(threading.Thread):
    def __init__(self, client, addr):
        theading.Thread.__init__(self)
        self.client = client
        self.addr = addr

    def run(self):
        # Copy code from Lab 3 server, from first line
        # that sends "READY" to the client. Note that
        # you must use "self.client.send()" instead of
        # "client.send()".



class Manager(threading.Thread):
    def __init__(self, max_connections):
        threading.Thread.__init__(self):
        self.max_connections = max_connections
        # Add data members for the running set and
        # FIFO queue.

    def add_client(self, t):
        # add t to the end of the FIFO queue

    def run(self):
        while True:
            # remove any finished threads from the running set

            # if there are threads waiting, and running set is
            # not full:
            #    - dequeue next threading
            #    - start the threading
            #    - add the thread to the running set

            # wait for 1 second


# The main server thread starts here.

# Create and start the manager

# Create the socket, bind() and listen()

while True:
    client, addr = s.accept()
    # Create the ClientHandler thread and add it to the manager
```