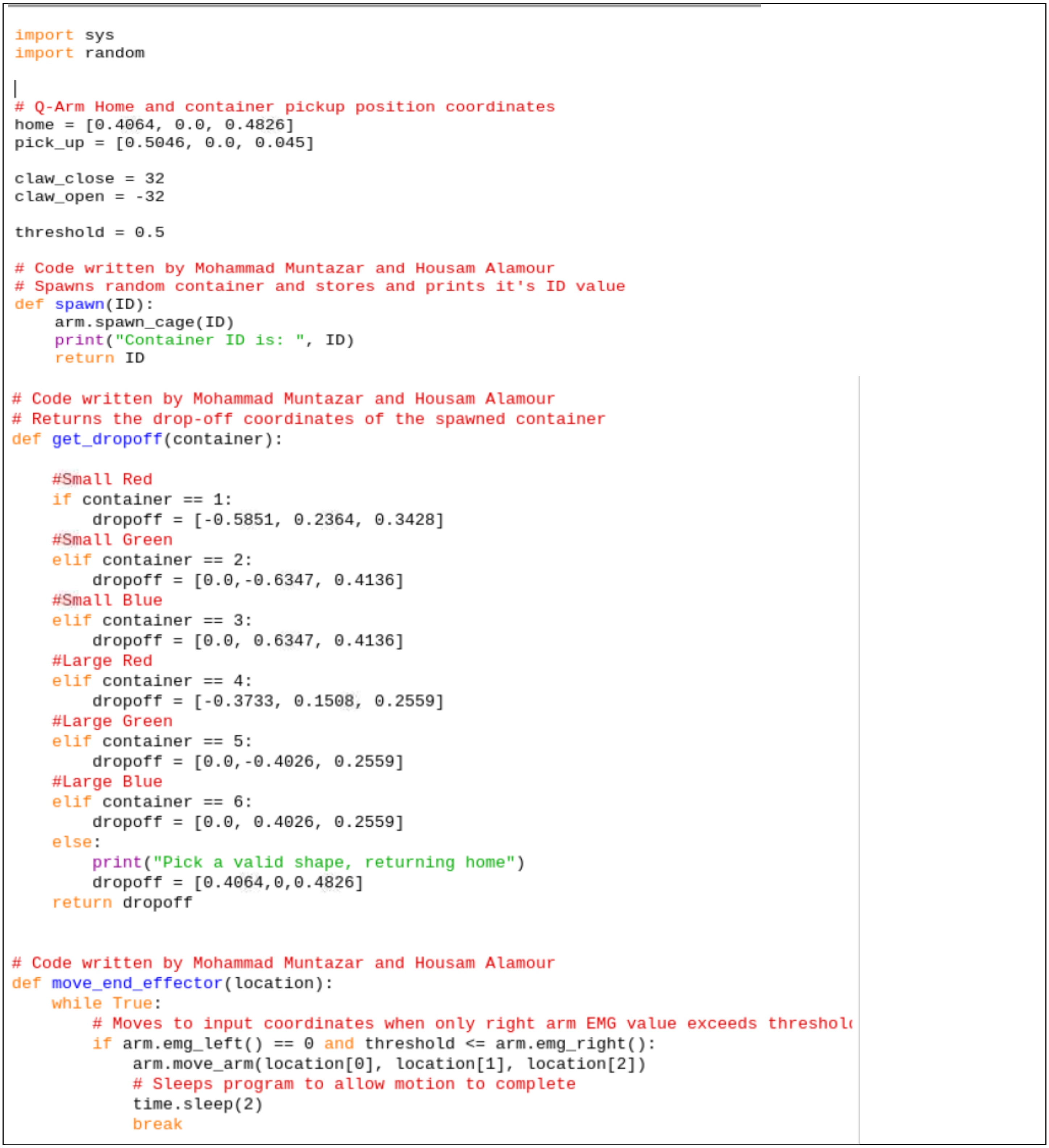
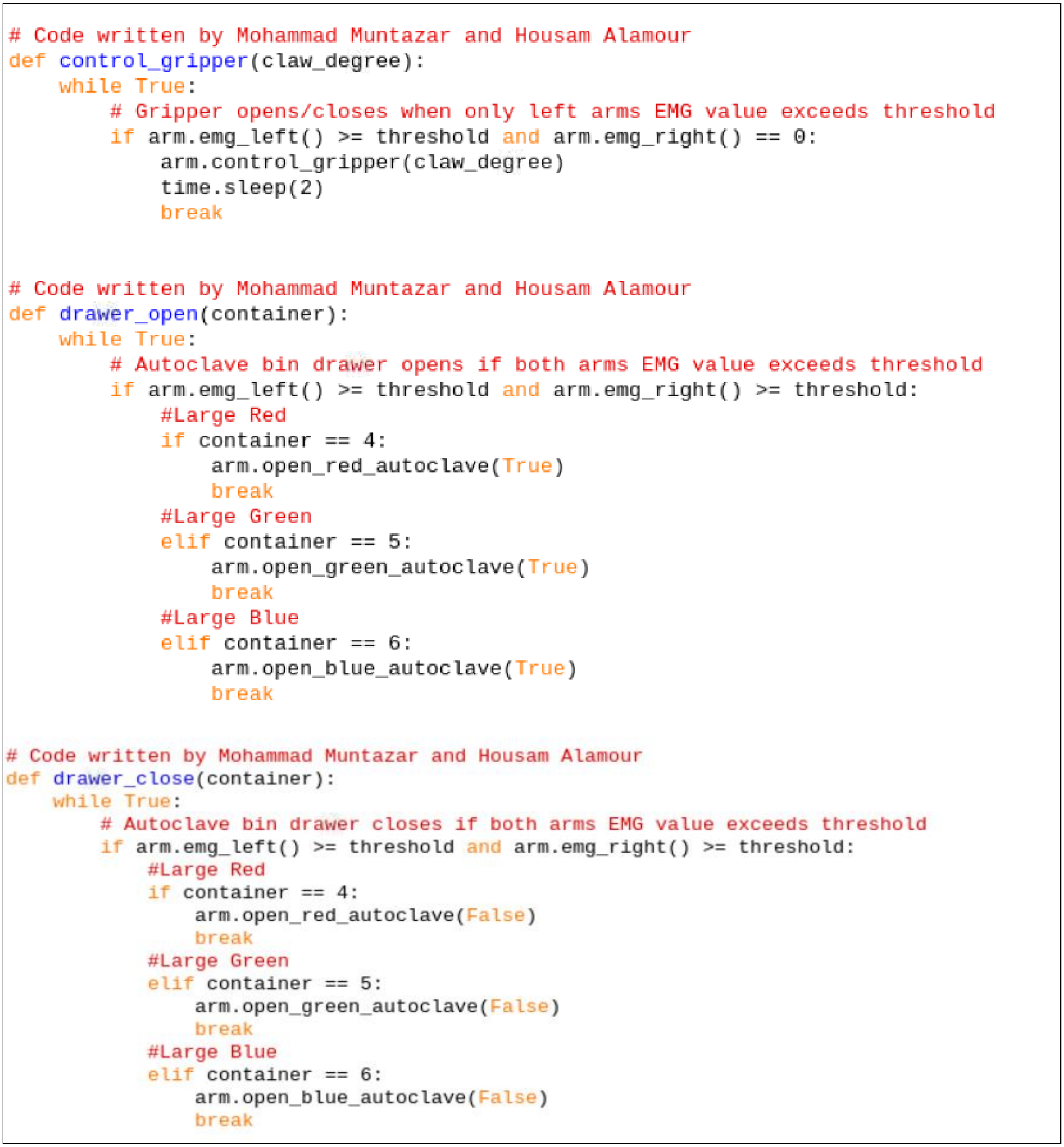
Appendix B - Computation:

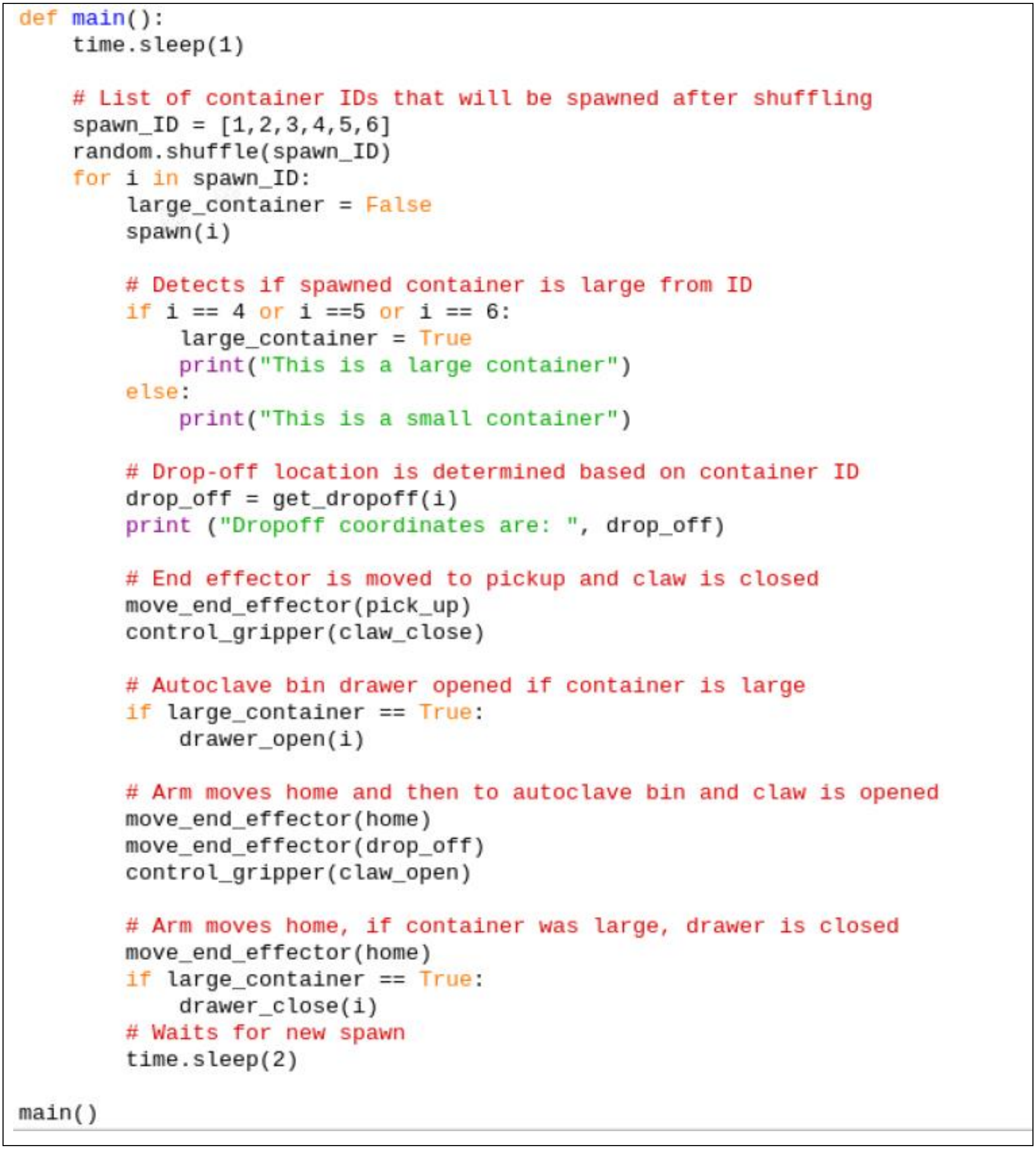
Computer Program:



```
import sys
import random

|
# Q-Arm Home and container pickup position coordinates
home = [0.4064, 0.0, 0.4826]
pick_up = [0.5046, 0.0, 0.045]

claw_close = 32
claw_open = -32

threshold = 0.5

# Code written by Mohammad Muntazar and Housam Alamour
# Spawns random container and stores and prints it's ID value
def spawn(ID):
    arm.spawn_cage(ID)
    print("Container ID is: ", ID)
    return ID

# Code written by Mohammad Muntazar and Housam Alamour
# Returns the drop-off coordinates of the spawned container
def get_dropoff(container):

    #Small Red
    if container == 1:
        dropoff = [-0.5851, 0.2364, 0.3428]
    #Small Green
    elif container == 2:
        dropoff = [0.0,-0.6347, 0.4136]
    #Small Blue
    elif container == 3:
        dropoff = [0.0, 0.6347, 0.4136]
    #Large Red
    elif container == 4:
        dropoff = [-0.3733, 0.1508, 0.2559]
    #Large Green
    elif container == 5:
        dropoff = [0.0,-0.4026, 0.2559]
    #Large Blue
    elif container == 6:
        dropoff = [0.0, 0.4026, 0.2559]
    else:
        print("Pick a valid shape, returning home")
        dropoff = [0.4064,0,0.4826]
    return dropoff


# Code written by Mohammad Muntazar and Housam Alamour
def move_end_effector(location):
    while True:
        # Moves to input coordinates when only right arm EMG value exceeds threshol
        if arm.emg_left() == 0 and threshold <= arm.emg_right():
            arm.move_arm(location[0], location[1], location[2])
            # Sleeps program to allow motion to complete
            time.sleep(2)
            break
```



```
# Code written by Mohammad Muntazar and Housam Alamour
def control_gripper(claw_degree):
    while True:
        # Gripper opens/closes when only left arms EMG value exceeds threshold
        if arm.emg_left() >= threshold and arm.emg_right() == 0:
            arm.control_gripper(claw_degree)
            time.sleep(2)
            break


# Code written by Mohammad Muntazar and Housam Alamour
def drawer_open(container):
    while True:
        # Autoclave bin drawer opens if both arms EMG value exceeds threshold
        if arm.emg_left() >= threshold and arm.emg_right() >= threshold:
            #Large Red
            if container == 4:
                arm.open_red_autoclave(True)
                break
            #Large Green
            elif container == 5:
                arm.open_green_autoclave(True)
                break
            #Large Blue
            elif container == 6:
                arm.open_blue_autoclave(True)
                break

# Code written by Mohammad Muntazar and Housam Alamour
def drawer_close(container):
    while True:
        # Autoclave bin drawer closes if both arms EMG value exceeds threshold
        if arm.emg_left() >= threshold and arm.emg_right() >= threshold:
            #Large Red
            if container == 4:
                arm.open_red_autoclave(False)
                break
            #Large Green
            elif container == 5:
                arm.open_green_autoclave(False)
                break
            #Large Blue
            elif container == 6:
                arm.open_blue_autoclave(False)
                break
```



```python
def main():
    time.sleep(1)

    # List of container IDs that will be spawned after shuffling
    spawn_ID = [1,2,3,4,5,6]
    random.shuffle(spawn_ID)
    for i in spawn_ID:
        large_container = False
        spawn(i)

        # Detects if spawned container is large from ID
        if i == 4 or i ==5 or i == 6:
            large_container = True
            print("This is a large container")
        else:
            print("This is a small container")

        # Drop-off location is determined based on container ID
        drop_off = get_dropoff(i)
        print ("Dropoff coordinates are: ", drop_off)

        # End effector is moved to pickup and claw is closed
        move_end_effector(pick_up)
        control_gripper(claw_close)

        # Autoclave bin drawer opened if container is large
        if large_container == True:
            drawer_open(i)

        # Arm moves home and then to autoclave bin and claw is opened
        move_end_effector(home)
        move_end_effector(drop_off)
        control_gripper(claw_open)

        # Arm moves home, if container was large, drawer is closed
        move_end_effector(home)
        if large_container == True:
            drawer_close(i)
        # Waits for new spawn
        time.sleep(2)

main()
```