

Question 1:

Potential yield: -1, 2, and 1.

Case -1:

Thread one and then thread two will read main.x in memory when it is equal to zero. Thread one then increments and completes, at this point main.x is 2. Then thread two completes, It thinks that main.x is still zero so it will set main.x to -1. Completing the program.

Thread two and then thread one will read main.x in memory when it is equal to zero. Thread one then increments and completes, at this point main.x is 2. Then thread two completes, It thinks that main.x is still zero so it will set main.x to -1. Completing the program.

Case 2:

Thread one and then thread two will read main.x in memory when it is equal to zero. Thread two then increments and completes, at this point main.x is -1. Then thread one completes, It thinks that main.x is still zero so it will set main.x to 2. Completing the program.

Thread two and then thread one will read main.x in memory when it is equal to zero. Thread two then increments and completes, at this point main.x is -1. Then thread one completes, It thinks that main.x is still zero so it will set main.x to 2. Completing the program.

Case:1

Thread one gets the value increments it by 2. So the value is main.x = 2. Then thread two is run that gets the main.x value of 2 and subtracts one from it and then saves it to main.x. The system output would be one.

Thread two gets the value and subtracts one from it, then saves it to main.x. Main.x= -1. Then Thread one runs, gets the main.x=-1 and adds 2 to it. Then it saves main.x=1. The system output should be one.

At first I thought that there would be a higher probability of getting one as an output. But now that I think of it more, there is a higher chance that the answer be 2 or -1 because threads have a higher chance of seeing main.x=0 then taking too long and waiting for the first task to finish. Ultimately it is indeterministic.