CS132: Software Engineering

Answer Sheet

Midterm Exam

15:00-16:40, May 8th, 2024

There are 5 problem sets and the total points are 20 points. Each problem set includes a few questions. For each question, the maximum possible points are stated.

Please write your answers **legibly on the answer booklet** so that we can read and understand your answers. If a problem seems ambiguous, please feel free to state your assumption explicitly and solve the problem. Obviously, your assumption should be reasonable and should not trivialize the problem.

Pledge. Copy the following pledge and sign your name in your answer booklet:

I neither cheated myself nor helped anyone cheat on this exam.

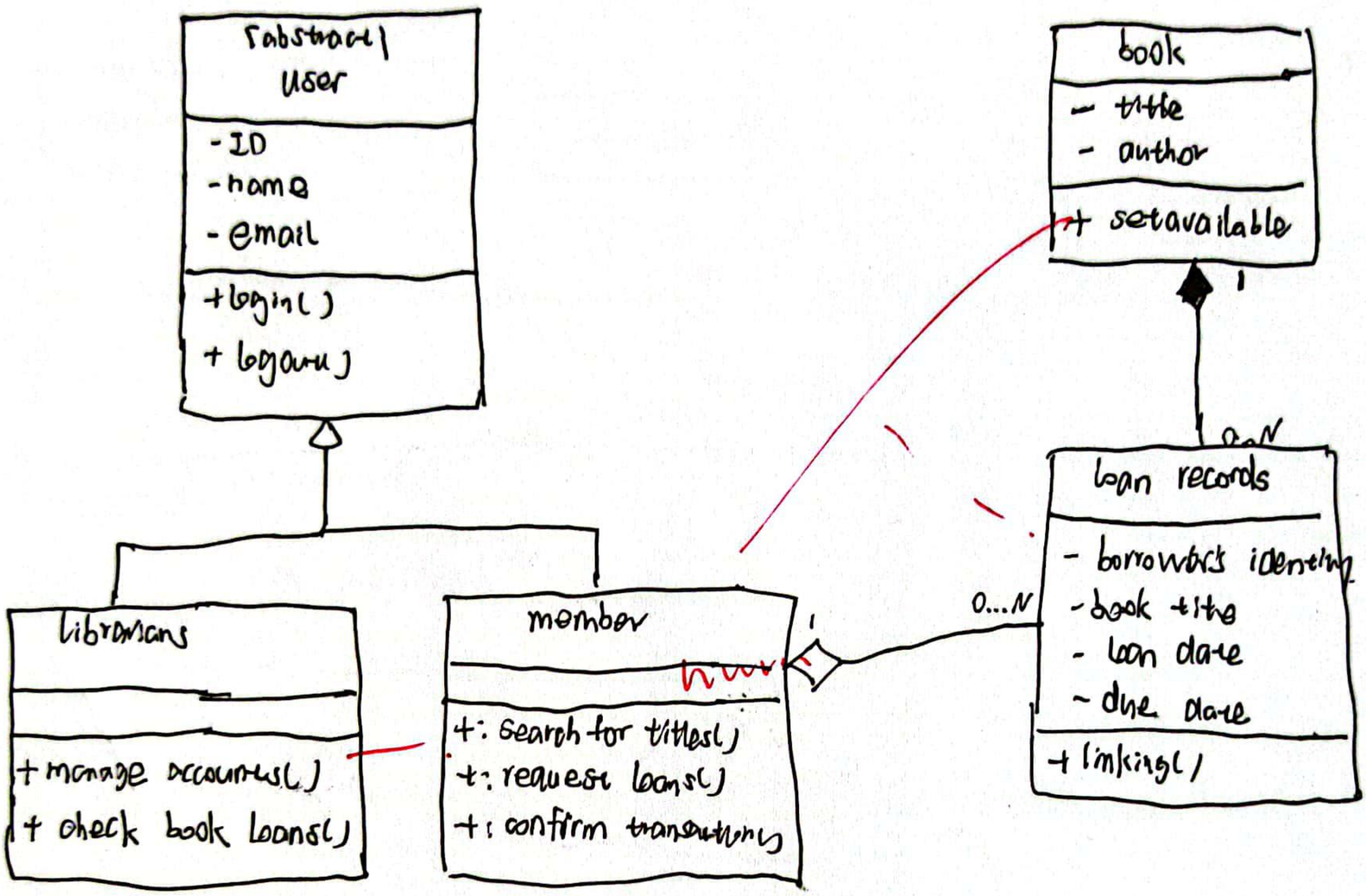
I neither cheated myself nor helped anyone cheat on this exam.

柳润康

2022533080

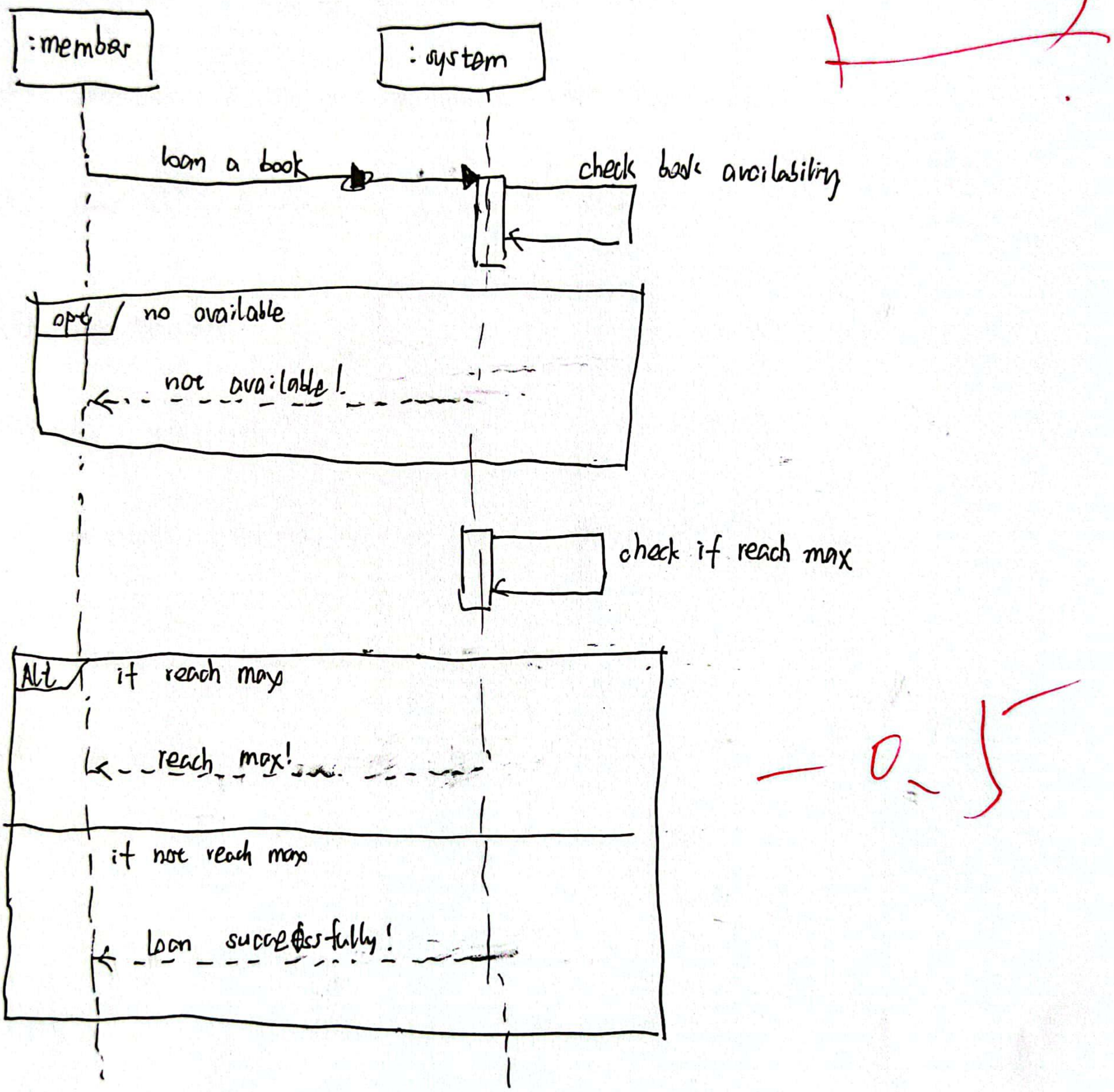
Problem 1. (4 points)

1)



-0.5

Problem 1 Cont. (4 points)



Problem 2. (4 points)

(1) deadlock. B may not go into A when $t \geq 3$

(2) deadlock. A may not go into C when $0 \leq t < 2$

(3) deadlock. A may not go into C when $t > 4$

(4) No deadlock

(5) No deadlock

(6) deadlock. A may not go into C when $0 \leq t < 2$

(7) deadlock. may stay in B forever

(8) No deadlock.

Problem 3. (4 points)

(1) A[] not deadlock

It's False. Since goat may stay in dead_goat forever.

cabbage may stay in eaten forever.

Wolf may stay in stuffed forever.

all have no enabled transitions.

(2) E<> (Wolf.stuffed and Goat.dead_goat and Cabbage.eaten)

It's True. ① Man brings wolf to East, then cabbage goes to eaten

② Man brings wolf back to West

③ Man goes to East by himself then

wolf goes to stuffed, and goat goes to dead_goat

(3) E<> (Man.E and Goat.E and Wolf.E and Cabbage.E)

It's True. ~~① Man brings wolf to East.~~

cabbage

① Man brings goat to East.

② Man goes to West by himself

③ Man brings wolf to East

④ Man brings goat to West

⑤ Man brings cabbage to East

⑥ Man goest to West by himself

⑦ Man brings goat to East.

Then all items and the man go to the E location

Problem 4. (4 points)

(a) ① x=7 z=7

1→2→3→4→5→6→7→8→9→10→3→11

make while (x>1) both T and F

if (y>3) T

if (z>0 and y>1) T

② ~~x=1~~ x=2 z=7

1→2→8→4→5→7→8→10→3→11

make while (x>1) both T and F

if (y>3) F

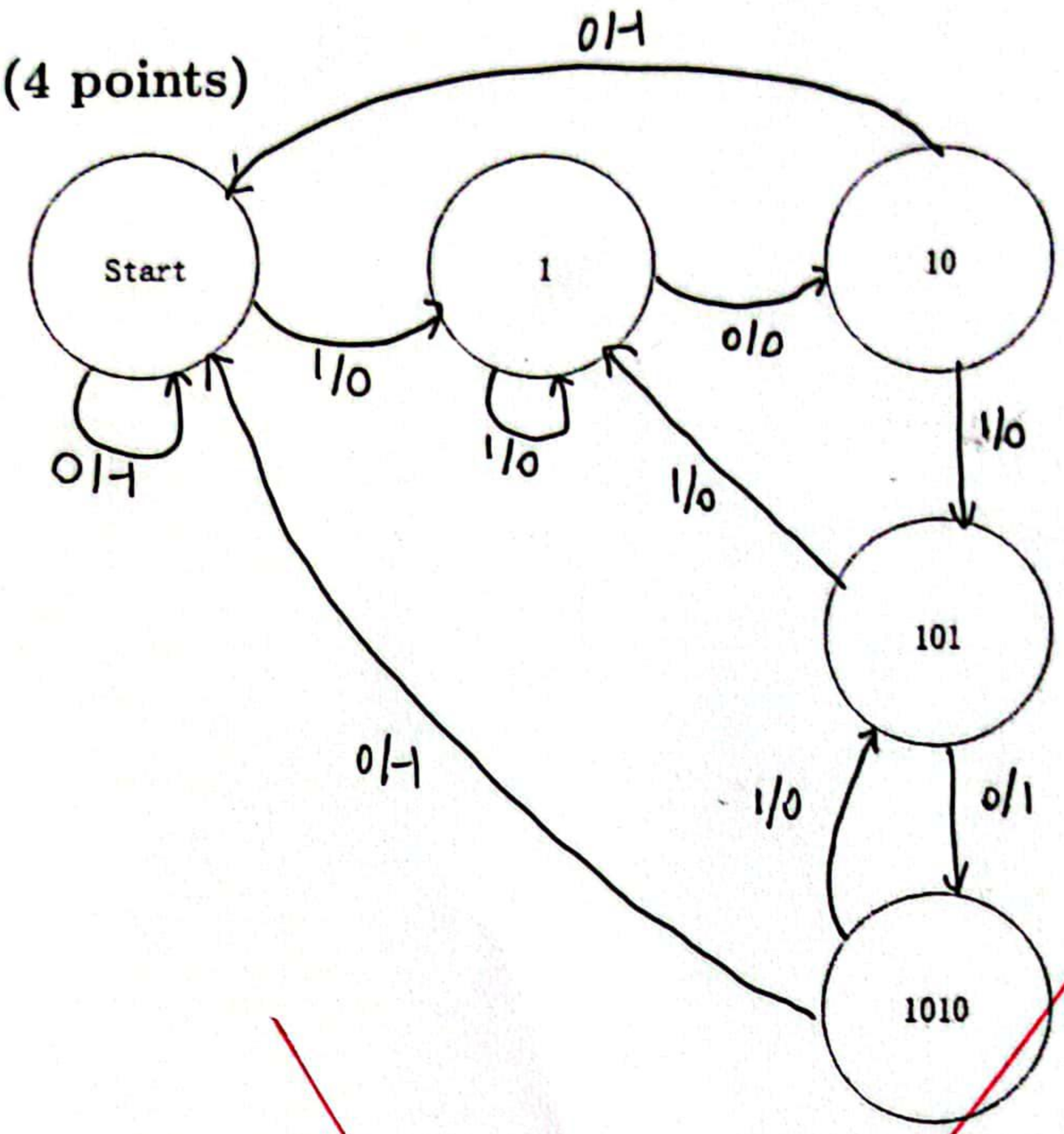
if (z>0 and y>1) F

(b) ① x=1 z=1

② x=7 z=-1

③ x=-1 z=0

Problem 5. (4 points)



Current State/Input	Input 0	Input 1
Start	start/-1	1/0
1	10/0	1/0
10	start/-1	101/0
101	1010/1	1/0
1010	start/-1	101/0

Test Case	TC1	TC2	TC3	TC4	TC5	TC6	TC7	TC8	TC9	TC10
Start State	start	start	1	1	10	10	101	101	1010	1010
Input	0	1	0	1	0	1	0	1	0	1
Final State	start	1	10	1	start	101	1010	1	start	101
Expected Output	-1	0	0	0	-1	0	1	0	-1	0
Test Coverage Item	1	2	3	4	5	6	7	8	9	10