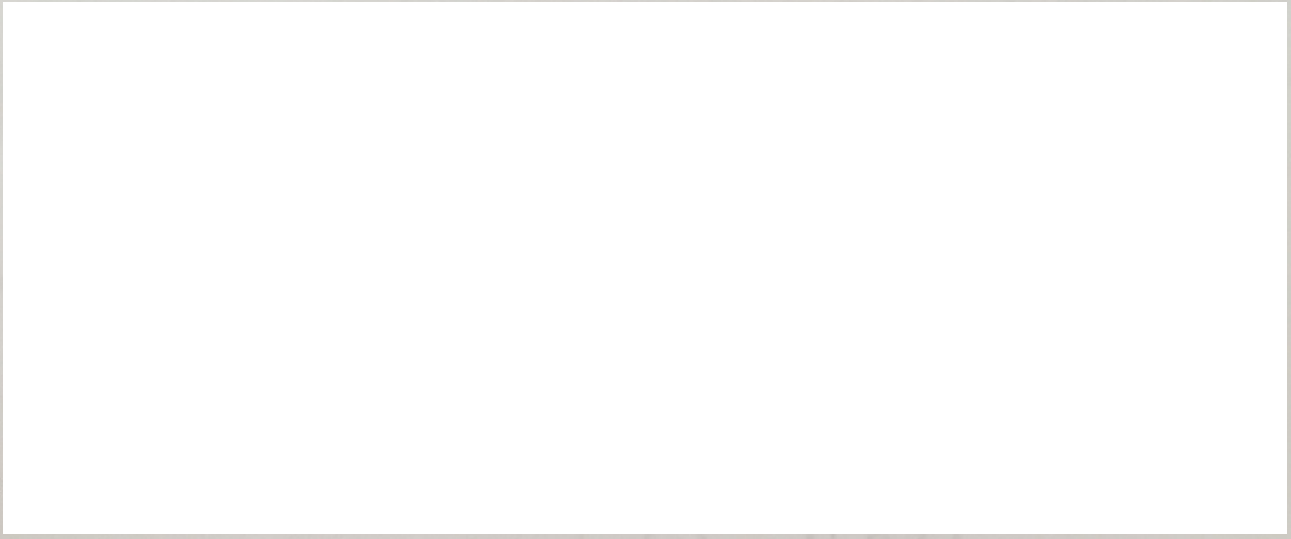


Q.1. (15 points) Consider a system consisting of 5 processes and 3 resources. The current **Allocation** and **Request** matrices are,

$$\text{Allocation} = \begin{bmatrix} 0 & 1 & 1 \\ 2 & 0 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 1 \\ 0 & 0 & 2 \end{bmatrix} \qquad \text{Request} = \begin{bmatrix} a & 0 & 0 \\ 2 & 0 & 3 \\ 0 & b & 1 \\ 1 & 0 & 0 \\ c & 0 & 2 \end{bmatrix}$$

and **Available** = [0 0 0]. What are the maximum possible values of **a**, **b** and **c** so that there will be no deadlock in the system? Show your work clearly!



Handwritten notes:

work = work + Allocation (1) = [0 0 0]
 Request (1) = true
 Request (2) = false
 Request (3) = false
 Request (4) = false
 Request (5) = false
 Request (6) = false
 Request (7) = false
 Request (8) = false
 Request (9) = false
 Request (10) = false
 Request (11) = false
 Request (12) = false
 Request (13) = false
 Request (14) = false
 Request (15) = false
 Request (16) = false
 Request (17) = false
 Request (18) = false
 Request (19) = false
 Request (20) = false
 Request (21) = false
 Request (22) = false
 Request (23) = false
 Request (24) = false
 Request (25) = false
 Request (26) = false
 Request (27) = false
 Request (28) = false
 Request (29) = false
 Request (30) = false
 Request (31) = false
 Request (32) = false
 Request (33) = false
 Request (34) = false
 Request (35) = false
 Request (36) = false
 Request (37) = false
 Request (38) = false
 Request (39) = false
 Request (40) = false
 Request (41) = false
 Request (42) = false
 Request (43) = false
 Request (44) = false
 Request (45) = false
 Request (46) = false
 Request (47) = false
 Request (48) = false
 Request (49) = false
 Request (50) = false
 Request (51) = false
 Request (52) = false
 Request (53) = false
 Request (54) = false
 Request (55) = false
 Request (56) = false
 Request (57) = false
 Request (58) = false
 Request (59) = false
 Request (60) = false
 Request (61) = false
 Request (62) = false
 Request (63) = false
 Request (64) = false
 Request (65) = false
 Request (66) = false
 Request (67) = false
 Request (68) = false
 Request (69) = false
 Request (70) = false
 Request (71) = false
 Request (72) = false
 Request (73) = false
 Request (74) = false
 Request (75) = false
 Request (76) = false
 Request (77) = false
 Request (78) = false
 Request (79) = false
 Request (80) = false
 Request (81) = false
 Request (82) = false
 Request (83) = false
 Request (84) = false
 Request (85) = false
 Request (86) = false
 Request (87) = false
 Request (88) = false
 Request (89) = false
 Request (90) = false
 Request (91) = false
 Request (92) = false
 Request (93) = false
 Request (94) = false
 Request (95) = false
 Request (96) = false
 Request (97) = false
 Request (98) = false
 Request (99) = false
 Request (100) = false

Q.1. (25 points) A system has three processes and three different types of resources. The current allocation and requests are as follows:

	Allocated	Request	Available
Process A	1 0 2	a b c	0 0 0
Process B	2 0 1	d e f	
Process C	1 1 0	g h j	

Apply the deadlock detection algorithm to compute the **upper limits** of **a, b, c, d, e, f, g** and **h** so that the sequence <ABC> will not lead to a deadlock (i.e. it is a safe process allocation sequence)?

requests	a	b	c	d	e	f	g	h	j
Upper limits	0	0	0	1	0	2	3	0	3

Q.5. (25 points) A system has four processes and five different types of resources. The current allocation and maximum needs are as follows:

	Allocated	Maximum	Available
Process A	1 0 2 1 1	1 1 2 1 3	0 0 2 1 1
Process B	2 0 1 1 1	2 2 X 1 1	
Process C	1 1 0 1 0	2 1 3 1 0	
Process D	1 1 1 1 0	1 1 2 2 1	

- a) What is the **largest possible value of X** so that DCBA is a safe sequence? In order to get credits, *you should show your work clearly!*

