Question 1

a)

Nkem	H	R	W
Elena	W	R	H
Fatima	W	H	R

Whittingdon (W)	N	F & E
Royal Free (R)	F	N / E
Highgate (H)	F	E / N

Step 1 - Find the Pairs

(N) Nkem	(H)	R	W
(E) Elena	~~H~~	R	H
(F) Fatima	(W)	H	R

Whittingdon (W)	N	(F)	~~E~~
Royal Free (R)	F	N	(E)
Highgate (H)	F	E	(N)

Step 2 - Find Next Pairs

N	(H)	R	W
E	W	(R)	H
F	(W)	H	R

W	N	(F)	E
R	F	N	(E)
H	F	E	(N)

Answer: Final Pair is:

NH, ER, FW

Question 1

b) i) Epsilon

ii) aa

iii) b

iv) a

c) i) String 1 : G -> X -> 0X -> 00V -> 00

String 2 : G -> V -> V01 -> 1V00 -> 100

ii) a) 110 (More 1's then 0's)

b) 00111 (More 1's then 0's)

iii) This grammar generates more 0's then 1's therefore the string "01" cannot be generated as it has an equal number of 0's and 1's

iv) This grammar generates two 0's for every 1

D) n = 9 k = 0, 2, 4, 6, 8

n x k = ?

Answer = 5x9 = 45

Question 2

a) 6 characters long

1 Lowercase letter

3 Digits

Total Lowercase Possibilities = 26^3

Total Digit Possibilities = 10^3

Total Combinations = $10^3 \times (26^3) = 17,576,000$

Digits can come in any order therefore = 6C3

$6C_3 = 20$ combinations

Total Passwords = $15 \times (10^3) \times (26^3) = 263,640,000$

B) i) $T(n) = 8T(n/2) + O(n^2)$

Where $a = 8$, $b = 2$, $f(n) = O(n^2)$ and then $C = 2$

$\text{Log}_b(a)$

$= \text{Log}_2(8)$

$= 3$ (Greater than C)

$T(n) = O(n^{\text{Log}_b(a)})$

$= O(n^3)$

ii) $T(n) = 3T(n/3) + O(n)$

Where $a = 3$, $b = 3$, $f(n) = O(n)$, and $C = 0$

$\text{Log}_b(a)$

$= \text{Log}_3(9)$

$= 3$ (Greater then C)

$\longrightarrow T(n) = O(n^{\log_b(a)})$

$= O(n^3)$

Question 2

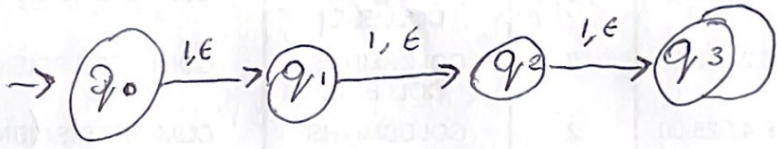
c) Context-free grammar that accepts the language of all binary settings with at most one occurence of 11

$S \rightarrow \epsilon / AB$

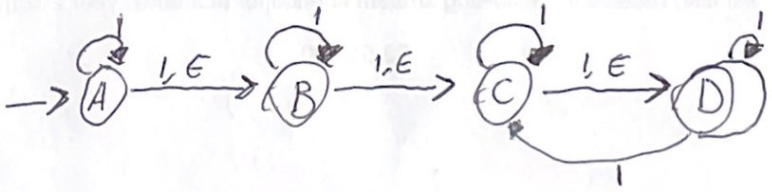
$A \rightarrow 11 / \epsilon$

$B \rightarrow 0/1/0B/1B/\epsilon$

D) Finite-automation that accepts all binary strings with at most one occurence of 111



- This FA can accept the empty string or 111
- Include 1's to get all binary settings
- At least one occurence of 111 should be present



e) Regular expression that accepts the language of all binary strings with no occurences of 111

$(0 + 00^*1 + 00^*10^*10^*)^*$