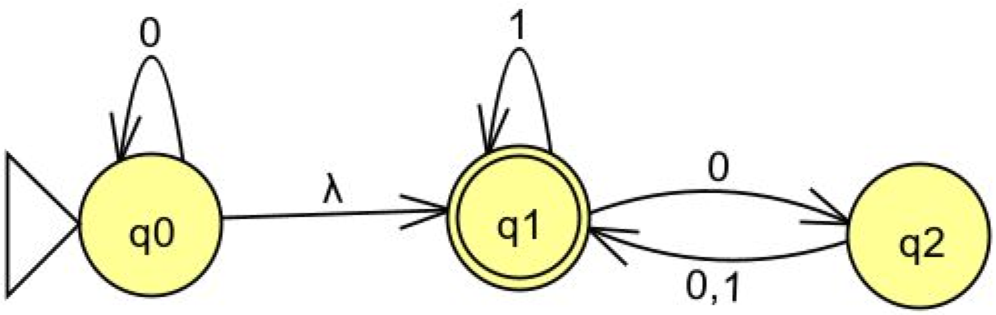
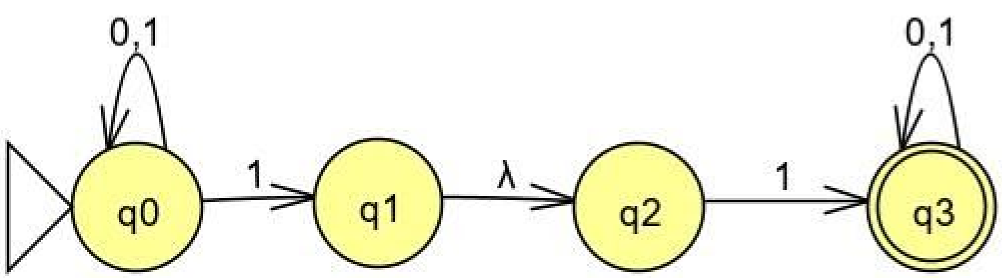
Question #1

Deterministic Finite Automata



Non Deterministic Finite Automata



Question #2

This can be proved through induction. The summation can be written as

$1^r + 2^r ... n^r$ with r > 0

if r=0 $\Rightarrow 1$

if r=1 $\Rightarrow \dfrac{n(n+1)}{2}$

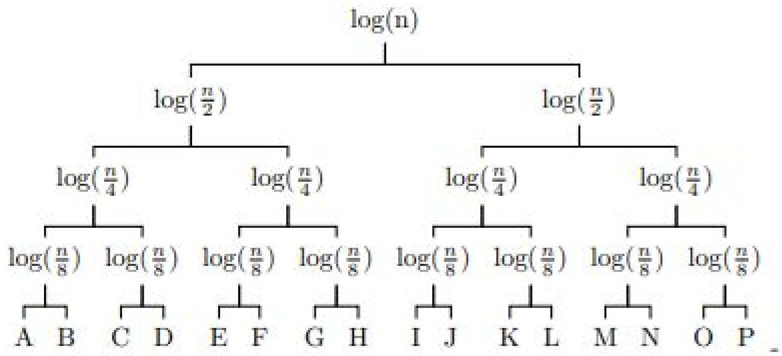
if r=2 $\Rightarrow \dfrac{n(n+1)(2n+1)}{6}$

$$\sum_{k=1}^{n} k^r = \frac{1}{r+1}\left[(n+1)^{r+1} - 1 - \sum_{j=1}^{n}(j+1)^{r+1} - k^{r+1} - (r+1)\cdot k^r\right]$$

$f(n) = \Theta\left[g(n)\right]$

Therefore $\sum_{k=1}^{n} k^r = \Theta\left[n^r\right]$

Question #3:



There are 2^1 nodes at level i. Therefore the cost is

$$log\frac{n}{2^i} = log(n-i)$$

Therefore cost of all operations is

$$\sum_{i=0}^{log(n)-1} logn - i$$

$$= \sum_{i=1}^{log(n)} i$$

$$= \Theta(n)$$

It can also be shown by the first condition of the masters theorem $c = \log_2 2 = 1$ therefore $n^c = n^1 = \Theta(n)$

Question #4

We need to show that one can simulate the other. It is less powerful than a regular Turing machine. Some DFA's can simulate the machine meaning it only accepts regular language. This machine is no more powerful than a DFA meaning they are equivalent in power