演算法方面我們將會使用Q\_learning 及SARSA來實作

首先兩演算法內部基礎概念來源於TD(Temporal Difference)-update rule，由下圖所示

C:\Users\User\AppData\Local\Temp\msohtmlclip1\02\clip_image001.png

Alpha、Gamma

藉由應用該公式，我們能透過程式在每一次的state-transition中進行Q value的更新，使得Q value能夠被逐漸訓練成在pacman遊戲中能表現出最好(optimal)成效的Q value。而實作過程中，我們也會應用到Epsilon Greedy以機率決定agent在state上擁有隨機選擇action的能力能夠來探索地圖世界。

對個別agent(怪獸及小精靈的)演算法差異說明

機器產生的替代文字:
Q-Learnin
0POy
一Sarsa
On-policy
InitializeQ(s,a)b|y
Repeat(foreachepisode):
Initialize
Repeahstepofepisode):
ChooseafromsusingpolicyderivedfromQ(e.g.,E-greedy)
Takeactiona,observer,,'
un8isterminal
|Q(),)arbitrarily
Repeat(e丨叩d:
Ch“罒,ug|deri、ulfromQ(eg,E-greedy)
R(foreach叩。fel,面):
Takeactionobseru'r,s
Chooseu'from'usingpolicyderi、edfromQ(e.g.,E-greedy)
|川Ⅲterminal