Francesco Michael Kusuma

13522038

1. **Jelaskan cara kerja dari algoritma Q-Learning dan SARSA!**

- **Algoritma Q-Learning** adalah algoritma *reinforcement learning* yang menggunakan prinsip *off-policy*, artinya algoritma ini mempelajari nilai optimal pada tahap selanjutnya, bukan tahapan yang benar-benar diambil. Algoritma ini memiliki cara kerja:

a. Inisiasi Q-table dengan nilai 0 semua.

b. Pada setiap langkah, agen berada dalam *state* s dan memiliki tindakan a berdasarkan eksplorasi atau eksploitasi(epsilon-greedy).

c. Agen melakukan tindakan a dan menerima *reward* r serta berpindah ke *state* s'.

d. Nilai pada Q-table diperbarui dengan persamaan

$$Q(s,a) \leftarrow Q(s,a) + \alpha(r + \gamma \max\_a' Q(s',a') - Q(s,a))$$

dengan α adalah *learning rate* ($0 < \alpha \leq 1$) menentukan seberapa cepat algoritma belajar, γ adalah *discount factor* ($0 \leq \gamma < 1$) menentukan seberapa penting nilai tahap selanjutnya dibanding nilai saat ini, dan $\max\_a' Q(s',a')$ nilai Q maksimum dari keadaan berikutnya s'.

e. Proses b-d diulang hingga mencapai konvergensi atau jumlah *epoch* telah terpenuhi.

- **Algoritma SARSA** adalah algoritma *reinforcement learning* yang menggunakan *prinsip on-policy*, artinya algoritma ini mempelajari nilai yang benar-benar diambil pada tahapan selanjutnya. Algoritma ini memiliki cara kerja:

a. Inisiasi Q-table dengan nilai 0 semua.

b. Pada setiap langkah, agen berada dalam *state* s dan memiliki tindakan a berdasarkan eksplorasi atau eksploitasi(epsilon-greedy).

c. Agen melakukan tindakan a dan menerima *reward* r serta berpindah ke *state* s'.

d. Agen memiliki tindakan berikutnya a′ di keadaan baru s′.

e. Nilai pada Q-table diperbarui dengan persamaan

$$Q(s,a) \leftarrow Q(s,a) + \alpha(r + \gamma Q(s',a') - Q(s,a))$$

dengan α adalah *learning rate* ($0 < \alpha \leq 1$) menentukan seberapa cepat algoritma belajar, γ adalah *discount factor* ($0 \leq \gamma < 1$) menentukan seberapa penting nilai tahap selanjutnya dibanding nilai saat ini, dan $Q(s',a')$ nilai Q dari keadaan berikutnya s'.

f. Proses b-e diulang hingga mencapai konvergensi atau jumlah *epoch* telah terpenuhi.

2. **Bandingkan hasil dari kedua algoritma tersebut, bagaimana hasil perbandingannya? Jika ada perbedaan, jelaskan alasannya!**

   Dengan menggunakan:

   - *learning rate* sebesar 0,8
   - *discount factor* 0,8
   - *exploration* sebesar 0,5

   diperoleh hasil di bawah ini

```
Q-Learning:
[[ 282.29469389  424.7718821 ]
 [ 282.29469389  473.079869  ]
 [ 424.7718821   526.75541   ]
 [ 473.079869    586.3949    ]
 [ 526.75541     652.661     ]
 [ 586.3949      726.29      ]
 [ 652.661       808.1       ]
 [ 726.29        899.        ]
 [ 808.1        1000.        ]
 [ 899.         1000.       ]]
Path: [2, 3, 4, 5, 6, 7, 8, 9, 3, 4, 5, 6, 7, 8, 9, 3, 4, 5, 6, 7, 8, 9, 3, 4, 5, 6, 7, 8, 9, 3, 4, 5, 6, 7, 8, 9, 3]
Total scores: 569

SARSA:
[[-156.49543736  -89.63422038]
 [-148.05963164  134.60995516]
 [  47.34973671  245.83196938]
 [  29.91874442  134.27782392]
 [ 227.19609428  344.91194639]
 [ 113.36553907  483.07625182]
 [ 209.63716615  565.9189943 ]
 [ 231.7518668   666.42677232]
 [ 228.64389566  706.90160219]
 [ 638.03619655  671.76787773]]
Path: [2, 3, 4, 5, 6, 7, 8, 9, 3, 4, 5, 6, 7, 8, 9, 3, 4, 5, 6, 7, 8, 9, 3, 4, 5, 6, 7, 8, 9, 3, 4, 5, 6, 7, 8, 9, 3]
Total scores: 569
```

Dari hasil ini, terlihat bahwa algoritma Q-Learning memiliki nilai Q-table yang relatif lebih besar dibanding dengan algoritma SARSA. Hal ini disebabkan karena algoritma Q-Learning menggunakan nilai maksimum dari langkah berikutnya untuk memperbarui nilai di Q-table sehingga dapat memperoleh nilai yang lebih optimal. Algoritma SARSA menggunakan aksi yang akan diambil selanjutnya sehingga belum tentu memiliki nilai yang optimal.

Dari jalur dan total skor yang diperoleh, algoritma Q-Learning dan SARSA memiliki jalur dan total skor yang sama.