سبحان کوکبی ۸۱۰۱۰۱۴۹۲

۱)

۲			+۵
۱	S		-۵
	۱	۲	۳

	(۱،۱)	(۱،۲)	(۱،۳)	(۲،۱)	(۲،۲)	(۲،۳)
V_0	۰	۰	-۵	۰	۰	+۵
V_1	۰	۰	-۵	۰	۳٫۶	+۵
V_2	۰	۳٫۱۴۲	-۵	۲٫۵۴۲	۳٫۹۲۴	+۵

(۱،۱) و (۲،۱) → → ↑ ← ↓

۰ ۰ ۰ ۰ → Max = ۰

action(→) $\frac{۸}{۱۰}(۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) = ۰$

action(↑) $\frac{۸}{۱۰}(۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) = ۰$

action(←) $\frac{۸}{۱۰}(۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) = ۰$

action(↓) " " " $= ۰$

(۱،۲) → ↑ ← ↓

-۳٫۲ -۵٫۰ ۰ -۵٫۴ → Max = ۰

action(→) $\frac{۸}{۱۰}(۰ + \frac{۹}{۱۰} \times -۵) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) = -۳٫۲$

action(↑) $\frac{۸}{۱۰}(۰ + \frac{۹}{۱۰} \times ۰) + \frac{۱}{۱۰}(۰ + \frac{۹}{۱۰} \times -۵) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) = -۵٫۰$

action(←) $\frac{۸}{۱۰}(۰ + \frac{۹}{۱۰} \times ۰) + \frac{۱}{۱۰}(۰ + \frac{۹}{۱۰} \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) = ۰$

action(↓) $\frac{۸}{۱۰}(۰ + \frac{۹}{۱۰} \times ۰) + \frac{۱}{۱۰}(۰ + \frac{۹}{۱۰} \times -۵) + \frac{۱}{۱۰} \times (۰ + ۰٫۹ \times ۰) = -۵٫۰۴۵$

(۲،۲) → ↑ ← ↓

۳٫۶ ۰٫۰۵ ۰ ۰٫۰۵ → Max = ۳٫۶

action(→) $\frac{۸}{۱۰} \times (۰ + ۰٫۹ \times ۵) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰} \times (۰ + ۰٫۹ \times ۰) = ۳٫۶$

action(↑) $\frac{۸}{۱۰} \times (۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۵) + \frac{۱}{۱۰} \times (۰ + ۰٫۹ \times ۰) = ۰٫۰۵$

action(←) $\frac{۸}{۱۰} \times (۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰} \times (۰ + ۰٫۹ \times -۵) = ۰$

action(↓) $\frac{۸}{۱۰} \times (۰ + ۰٫۹ \times ۰) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۵) + \frac{۱}{۱۰}(۰ + ۰٫۹ \times ۰) = ۰٫۰۵$

۰ →	۳٫۶ →	+۵
↑ ۰	● ← ۰	-۵

P_1

۳٫۲۴۸ →	۳٫۵۰۴۲ →	+۵
↑ ۰	↑ ۱٫۹۲۸۸	-۵

(۲، ۱) ۳٫۵۰۴۲ → ↑ ← ↓

(→): $\frac{۸}{۱۰} \times (۰ + \frac{۱۷}{۱۸} \times ۵) + \frac{۱}{۱۰} \times (۰ + \frac{۱۷}{۱۸} \times ۳٫۶) + \frac{۱}{۱۰}(۰ + \frac{۱۷}{۱۸} \times ۰) = ۳٫۵۰۴۲$

(د)

	۱	۲	۳
۲	۰	۳,۶	+۵
۱	۰	۰	-۵

۲,۵۹۲ →	۳,۹۲۴ →	۵
↑ ۰	↑ ۳,۱۴۲	-۵

s	۱,۱	۱,۲	۱,۳	۲,۱	۲,۲	۲,۳	
π*(s)	↑	↑	−	→	→	−	

(ه) (۱,۱) → ↑ ← ↓
۰ ۰ ۰ ۰ max = ۰

(۲,۱) → ↑ ← ↓
۲,۵۹۲ ۰,۳۲۴ ۰ ۰,۳۲۴ max = ۲,۵۹۲

→ : ۸/۱۰ × ۹/۱۰ × ۳,۶ + ۱/۱۰ × ۰ + ۱/۱۰ × ۰ = ۲,۵۹۲

↑ : ۱/۱۰ × ۹/۱۰ × ۳,۶ = ۰,۳۲۴

← : ۰

↓ : as same as (↑) = ۰,۳۲۴

(۱,۲) : → ↑ ← ↓
-۳,۲۷۶ ۲,۱۴۲ ۰,۳۲۴ -۰,۴۵ max → ۲,۱۴۲

→ : ۸/۱۰ × ۹/۱۰ × -۵ + ۱/۱۰ × ۹/۱۰ × ۳,۶ + ۰ = -۳,۲۷۶

↑ : ۸/۱۰ × ۹/۱۰ × ۳,۶ + ۱/۱۰ × ۹/۱۰ × -۵ + ۰ = ۲,۱۴۲

← : ۱/۱۰ × ۹/۱۰ × ۳,۶ = ۰,۳۲۴

↓ : ۱/۱۰ × ۹/۱۰ × -۵ = -۰,۴۵

(۲,۲) : → ↑ ← ↓
۳,۹۲۴ ۳,۰۴۲ ۰,۳۲۴ ۰,۴۵ max = ۳,۹۲۴

→ : ۸/۱۰ × ۹/۱۰ × ۵ + ۱/۱۰ × ۹/۱۰ × ۳,۶ = ۳,۹۲۴

↑ : ۸/۱۰ × ۹/۱۰ × ۳,۶ + ۱/۱۰ × ۹/۱۰ × ۵ = ۳,۰۴۲

← : ۱/۱۰ × ۱/۱۰ × ۳,۶ = ۰,۳۲۴

↓ : ۰,۴۵

Monte Carlo

[redacted]

۱) (۱,۱), (۱,۲), (۱,۳)

reward (۱,۱) → (۱,۲) = ۰

" (۱,۲) → (۱,۳) = -۵

۲) (۱,۱), (۱,۲), (۲,۲), (۲,۳)

" (۱,۱) → (۱,۲) = ۰

" (۱,۲) → (۲,۲) = ۰

" (۲,۲) → (۲,۳) = ۵

۳) (۱,۱), (۲,۱), (۲,۲), (۲,۳)

reward (۱,۱) → (۲,۱) = ۰

" (۲,۱) → (۲,۲) = ۰

" (۲,۲) → (۲,۳) = ۵

V* ((۱,۱)) = ۱/۳ [(۱×۰ + ۰,۹×-۵) + (۱×۰ + ۰,۹ × ۰ + (۰,۹)^۲×۵) + (۱×۰ + ۰,۹×۰ + (۰,۹)^۲× ۵)] = ۱,۲

V* (۲,۲) = ۱/۲ [(۱×۵) + (۱×۵)] = ۵

الگوریتم (DQN) یکی از الگوریتم‌های معروف در حوزه یادگیری تقویتی عمیق است. این الگوریتم با ترکیب یادگیری تقویتی و شبکه عصبی عمیق توانست موفقیت‌های قابل توجهی داشته باشد.

DQN ترکیبی از Q-learning و شبکه عصبی عمیق است. هدف Q-learning یافتن تابع Q است که حداکثر مقدار پاداش مورد انتظار را برای هر حالت و اقدام ممکن محاسبه کند. این تابع با استفاده از معادله بلمن به‌روزرسانی می‌شود. در DQN از شبکه عصبی عمیق برای تقریب این تابع Q استفاده می‌شود

کاربرد DQN ← بازی‌های ویدئویی، رباتیک، مدیریت منابع: در آینده مانند مدیریت ترافیک شبکه‌های کامپیوتری یا تخصیص منابع در سیستم‌های بزرگ، DQN می‌تواند کمک کند.

مزایا: توانایی یادگیری از تجارب خام و داده‌های پیچیده.
قابلیت استفاده در محیط‌های متنوع و پویا.

معایب: نیاز به حافظه زیاد برای ذخیره Replay Buffer
مشکلات پایداری و همگرایی به دلیل وابستگی زمانی داده‌ها.