**نازنین درستکار**

**شماره دانشجویی: 39916341054122**

**Homework1:**

**ربات های فوتبالیست**

|  |  |  |  |
| --- | --- | --- | --- |
| **کارآیی capabitities** | **محیطEnviroment** | **عملگر actuators** | **سنسور sensor** |
| * **حرکت در جهات مختلف( جلو،عقب،چپ، راست)** * **تشخیص تیم خود و تیم حریف** * **تعامل با توپ(گرفتن توپ، کنترل توپ)** | * **سطح زمین(چمنی، سخت)** * **حضور توپ** * **حضور دیگر بازیکنان** * **شرایط آب و هوایی** | **چهارچوب حرکت**  **سیستم کنترل** | * **سنسور دید(تشخیص توپ)** * **سنسور لمسی( اندازه گیری فشار)** * **سنسور شتاب سنج** |

**Homework2 :**

**کد هشت وزیر شطرنج**

def is\_safe(board, row, col):

for i in range(row):

if board[i][col] == 1:

return False

for i, j in zip(range(row, -1, -1), range(col, -1, -1)):

if board[i][j] == 1:

return False

for i, j in zip(range(row, -1, -1), range(col, len(board))):

if board[i][j] == 1:

return False

return True

def solve\_n\_queens(board, row):

if row >= len(board):

return True

for col in range(len(board)):

if is\_safe(board, row, col):

board[row][col] = 1

if solve\_n\_queens(board, row + 1):

return True

board[row][col] = 0

return False

def print\_board(board):

for row in board:

print(" ".join(str(x) for x in row))

def main():

n = 8

board = [[0 for \_ in range(n)] for \_ in range(n)]

if solve\_n\_queens(board, 0):

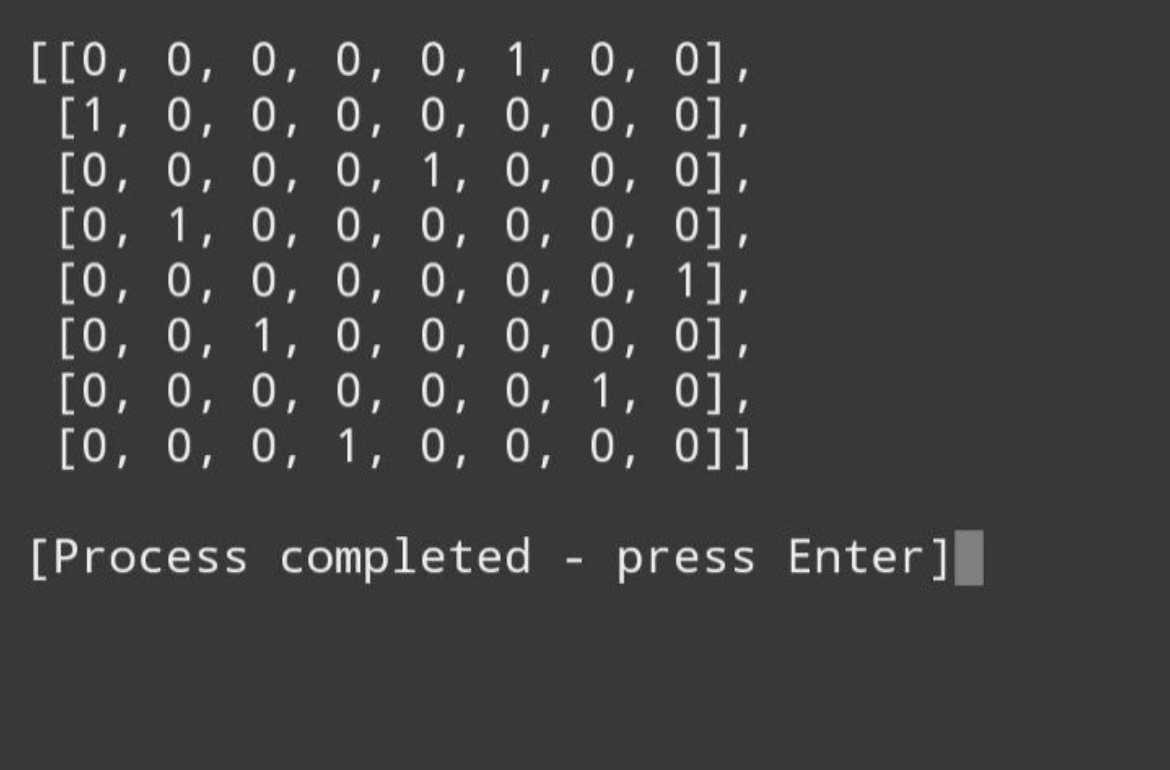
print\_board(board)

else:

print("هیچ راه حلی وجود ندارد.")

if \_\_name\_\_ == "\_\_main\_\_":

main()



**Homework3**

**چگونه می توان با مسائل غیر قطعی رفتار کرد؟**

**1. روش‌های احتمالی**

استفاده از روش‌های احتمالی می‌تواند در برخورد با مسائل غیر قطعی موثر باشد. این روش‌ها شامل مدل‌های آماری، شبکه‌های بیزین و روش‌های مونت کارلو هستند.

**مثال:**

* **شبکه‌های بیزین**: این شبکه‌ها می‌توانند برای مدل‌سازی روابط علّی و احتمالاتی بین متغیرها استفاده شوند.
* **روش‌های مونت کارلو**: این روش‌ها برای تقریب توزیع‌های احتمالاتی و انجام شبیه‌سازی‌های متعدد به کار می‌روند.

**2. روش‌های یادگیری ماشین**

الگوریتم‌های یادگیری ماشین، به ویژه روش‌های یادگیری تقویتی و یادگیری عمیق، می‌توانند در شرایط غیر قطعی موثر باشند.

**مثال:**

* **یادگیری تقویتی**: این روش به عامل اجازه می‌دهد از طریق تعامل با محیط و دریافت پاداش‌ها و مجازات‌ها، استراتژی‌های بهینه را یاد بگیرد.
* **یادگیری عمیق**: شبکه‌های عصبی عمیق می‌توانند برای پیش‌بینی نتایج در شرایط غیر قطعی استفاده شوند.

**3. الگوریتم‌های ژنتیک و تکاملی**

این الگوریتم‌ها از اصول تکامل زیستی برای جستجو و یافتن راه‌حل‌های بهینه در مسائل غیر قطعی استفاده می‌کنند.

**مثال:**

* **الگوریتم‌های ژنتیک**: این الگوریتم‌ها با استفاده از عملیات تقاطع و جهش، جمعیت‌های جدیدی از راه‌حل‌ها تولید می‌کنند.
* **الگوریتم‌های کلونی مورچه**: این الگوریتم‌ها از رفتار جمعی مورچه‌ها برای پیدا کردن مسیرهای بهینه استفاده می‌کنند.

**4. استفاده از سیستم‌های فازی**

سیستم‌های فازی می‌توانند با عدم قطعیت و اطلاعات مبهم به خوبی کار کنند.

**مثال:**

* **کنترل فازی**: در این سیستم‌ها، قوانین فازی برای مدل‌سازی و کنترل سیستم‌های پیچیده استفاده می‌شوند.
* **استدلال فازی**: این روش برای تصمیم‌گیری در شرایطی که اطلاعات کامل نیست، به کار می‌رود.

**5. روش‌های ترکیبی**

ترکیب چندین روش می‌تواند به بهبود عملکرد در مواجهه با مسائل غیر قطعی کمک کند.

**مثال:**

* **ترکیب یادگیری تقویتی و شبکه‌های بیزین**: استفاده از شبکه‌های بیزین برای مدل‌سازی محیط و یادگیری تقویتی برای بهینه‌سازی استراتژی‌ها.
* **ترکیب الگوریتم‌های ژنتیک و یادگیری ماشین**: استفاده از الگوریتم‌های ژنتیک برای جستجوی فضای راه‌حل‌ها و یادگیری ماشین برای بهبود دقت پیش‌بینی.

**6. شبیه‌سازی و آزمایش‌های متعدد**

اجرای شبیه‌سازی‌های متعدد و تحلیل نتایج می‌تواند به درک بهتر مسائل غیر قطعی و یافتن راه‌حل‌های بهینه کمک کند.

**مثال:**

* **شبیه‌سازی مونت کارلو**: اجرای شبیه‌سازی‌های متعدد با استفاده از توزیع‌های احتمالی برای بررسی نتایج مختلف.

**نتیجه‌گیری**

هر یک از این روش‌ها می‌تواند بسته به نوع مسئله و شرایط خاص آن، موثر باشد. ترکیب این روش‌ها و تنظیم آن‌ها بر اساس نیازهای خاص مسئله می‌تواند به دستیابی به بهترین نتایج در شرایط غیر قطعی کمک کند.

**Homework4**

