**인공지능 Assignment 2**

2019062833 김유진

1. 코드 설명

1) 프로그램 시작 시 Q 테이블을 한 번 초기화 시켜주어서 각 지점마다 딕셔너리를 생성시켜준다. 이후상하 좌우 4방향 중 이동가능한 방향에 대해 키-값 쌍을 생성한다. (아래 init table 함수 설명 참고) 예를 들면 시작점이 1인 경우 오른쪽으로 2, 아래쪽으로 6으로 갈 수 있으므로 q\_table[1] 에는 key가 2이고 값이 0, key가 6이고 값이 0인 두 쌍을 가진 딕셔너리가 들어가게 된다.

2) 파일에서 게임의 map을 읽어와서 list에 저장한 후 시작 지점을 파악한다.

3) Q 학습이 수렴할 수 있도록 충분히 큰 횟수(10000번)로 게임을 실행한다.

4) 게임을 실행하게 되면 시작 지점부터 랜덤하게 다음 이동을 선택하게 된다. 이후 큐테이블을 업데이트 하게 되는데 (q\_table[now\_point][next\_point] = new\_q), 이때 new\_q는 next\_point에 도착했을 때 얻게 되는 즉각 보상(next point가 어떤 타입이었는지에 따라 다르게 부여됨)과 delayed 보상의 최댓값에 감마 값을 곱한 것을 더해서 생성된 값이다. 이와 같이 임의의 이동을 반복하다가 골에 도착하거나 폭탄을 만나게 되면 게임을 종료하게 되며 다음 게임을 재시작하게 된다.

4) 모든 테스트 게임이 끝난 후 시작 지점에서 목표 지점까지의 경로와 시작 지점에서의 Q 최댓값을 output.txt 파일에 출력한다. 이 때 경로를 찾는 방식은 현재의 지점에서 선택할 수 있는 action들 중 Q 값이 가장 큰 action을 선택하는 것이다.

2. 함수 설명

1) init\_table :

Q\_table을 만들어주는 함수로 테이블은 각 지점(25개의 point)에 대해 딕셔너리를 보유한다. 각 딕셔너리는 해당 지점에서 갈 수 있는 모든 위치(상하좌우)에 대해 key-value 쌍을 가지며, 이때 key는 point number, value는 Q 값이다.

텍스트이(가) 표시된 사진

자동 생성된 설명

2) read\_file/write\_file :

파일에서 입력을 받아 map의 상태를 받아오고 시작 지점을 파악한다. 모든 게임이 끝난 이 후 파일에 경로와 시작지점의 Q 최댓값을 출력한다. 경로를 찾을 때는 큐 값이 최대가 되는 방향으로 이동하게 된다.

텍스트이(가) 표시된 사진

자동 생성된 설명 텍스트이(가) 표시된 사진

자동 생성된 설명

3) start\_game : 게임을 시작

현재 지점에서 이동이 가능한 지점들 중 하나를 랜덤으로 고른다. 그 지점에 도달했을 때 받게 되는 점수 값 + 추후에 얻게 될 delayed Q 값의 최댓값에 감마값이 0.9를 곱한 값을 더해서 해당 지점의 큐 테이블을 업데이트 한다. 골이나 폭탄에 도착하게 되면 게임이 끝이 나게 되며 게임이 끝날 때까지 위의 과정을 반복한다.

텍스트이(가) 표시된 사진

자동 생성된 설명 텍스트이(가) 표시된 사진

자동 생성된 설명

3. 실험 결과 및 분석 – 보너스 지점에 관해

1) reward = 1

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트, 화면, 스크린샷이(가) 표시된 사진

자동 생성된 설명

2) reward = 10

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

3) reward = 20

-경로를 찾을 수 없음

-보너스 지점 근처에서 빠져나오지 못함 (7<->8을 반복하게 됨)

-텍스트이(가) 표시된 사진

자동 생성된 설명

4) 분석

- 보너스가 커짐에 따라 시작지점의 Q 값의 최댓값이 커진다.

- 보너스가 일정수준 이상으로 커지게 되면 goal로 가기 전에 보너스 근처에서 잡혀서 다른 지점으로 이동할 수 없게 된다.

- 현재의 지점에서 최선의 선택을 선택하는 local 탐색이기 때문에 최적의 해를 찾는 것이 항상 보장되는 것은 아닌 것으로 보인다.