

## 2025-02 KNU RL Competition Round 3

***Avoid Blurp!***



**Time elapsed: 3.80 seconds**

세 번째 RL Competition은 하늘에서 떨어지는 보글보글Blurp을 피해 최대한 오랫동안 살아남는 **Avoid Blurp** 입니다!

## Purpose

플레이어는 마리오를 좌우로 조작해서 하늘에서 떨어지는 보글보글을 피해야 합니다. 단순해보이지만, 보글보글은 시간이 갈수록 더 많이 등장하고, 그 위치와 속도 또한 무작위로 변합니다. 여러분은 최대 2분 동안 하늘에서 떨어지는 보글보글의 폭격을 피하는 에이전트를 훈련해야 합니다.

## Prerequisites

### Python 3.12.x

Grid Crossing은 Python 3.12 버전에서 동작하도록 구현되었습니다 (그 외의 버전에서는 동작을 보장할 수 없습니다)

### Installation

Grid Crossing과 그 외 과제에서 수행할 강화 학습 환경은 다음과 같이 설치할 수 있습니다:

```
pip install -U kymnasium
```

그 외 실행에 필요한 `gymnasium`, `pygame`, `numpy` 등의 라이브러리는 의존성으로 함께 설치되므로 별도 설치할 필요는 없습니다.

※ 이전에 이미 설치했다라도, **-U 옵션을 통해 최신 버전 (1.0.9)으로 업데이트 하셔야 합니다!**

## Methods

`kymnasium.avoid_blurp.ManualPlayWrapper(env_id: str, debug: bool, **kwargs).play()`

- `env_id`: `gymnasium.make()` 등을 통해 환경을 불러올 때 사용하는 환경의 id를 의미합니다.
- `debug`: `True`일 경우 각종 로그를 출력합니다

- **kwargs:** `gymnasium.make()`에서 `id` 이외에 필요한 인자들을 정의합니다 (예.  
`render_mode = "human"`)

조작 방법은 다음과 같습니다:

- 방향키  $\leftarrow$ : 마리오가 좌측으로 조금 이동합니다.
- 방향키  $\rightarrow$ : 마리오가 우측으로 조금 이동합니다.

## Environment

### Environment Generation

Grid Adventure 환경은 일반적인 `gymnasium` 환경처럼 다음과 같이 생성할 수 있습니다:

```
import gymnasium as gym

env = gym.make(
    # Avoid Blurp의 ID
    id='kymnasium/AvoidBlurp-Normal-v0',
    # "human" 일 시 메인 스레드에서 게임 화면 렌더링,
    # "rgb_array" 일 시 백그라운드에서 게임 화면 렌더링
    render_mode='human',
    # "True" 일 시 BGM 재생; 오직 render_mode = "human"일 때만 동작
    bgm=True,
    # "custom" 일 시 마리오와 보글보글의 속성들을 observation으로 반환
    # "image" 일 시 화면에 출력되는 이미지를 그대로 observation으로 반환
    obs_type='custom'
)
```

### Observation Space

`obs_type = 'custom'`

에이전트는 마리오와 보글보글의 속성들을 각각 **player**와 **enemies** 라는 키를 갖고 Numpy 행렬을 값으로 가지는 딕셔너리로 알 수가 있습니다.

```
{
    "player": [x 좌표, y좌표, 너비, 높이, 속도],
```

```

    "enemies": [
        [x좌표, y좌표, 너비, 높이, 속도, 가속도],
        [x좌표, y좌표, 너비, 높이, 속도, 가속도],
        ...
    ]
}

```

**player**는 플레이어가 조작하는 마리오의 속성을 5개의 원소를 가지는 Numpy 행렬로 표시합니다. **enemies**의 속성은 [30, 6] Numpy 행렬로 제공되며, 각 행마다 화면에 표시된 보글보글의 속성을 6개의 원소로 표현합니다.

화면에 표시되지 않은 보글보글의 경우 모든 원소의 값이 0이며, x, y 좌표는 화면 좌측 최상단이 각각 0, 0입니다.

`obs_type = 'image'`

이번 RL Competition 부터는 표시된 화면의 이미지를 observation으로 활용할 수 있습니다! 화면의 이미지는 [600, 750, 3] Numpy 행렬로 반환되며, 각각 화면 너비, 높이, RGB를 의미합니다. 원소의 값은 [0, 255] 범위 내에 존재합니다.

## Action Space

에이전트가 움직이도록 하기 위해서는 **Env.step()** 호출 시 다음 숫자 중 하나를 전달해야 합니다:

- 0: 움직이지 않습니다.
- 1: 마리오가 좌측으로 조금 이동합니다.
- 2: 마리오가 우측으로 조금 이동합니다.

## Reward Space

어떤 행동이든 보상의 양을 0으로 설정했습니다. 여러분들이 자유롭게 보상을 설계해보세요!

## Info

현재까지 경과한 시간을 다음과 같이 반환합니다:

```

{
    "time_elapsed": XXXX
}

```

```
}
```

## Episode End

보글보글과 마리오가 충돌하면 *terminated = False, truncated = True*로 종료됩니다.

2분 동안 보글보글을 피했다면 *terminated = True, truncated = False*로 종료됩니다.

## Submission

이루리에 제출해야 하는 파일은 두 개입니다.

### Python for Training Your Own Agent (\*.py)

아래 작성 예시를 복사하여, 자신만의 에이전트를 정의하고 이 에이전트를 훈련하는 코드를 구현한 후 제출하세요.

```
import gymnasium as gym
import kymnasium as kym
from typing import Any, Dict

'''
kymnasium.Agent를 상속하여
자신만의 에이전트를 구현
'''
class YourAgent(kym.Agent):
    def act(self, observation: Any, info: Dict):
        pass

    @classmethod
    def load(cls, path: str) -> 'kym.Agent':
        pass

    def save(self, path: str):
        pass

def train():
    '''
```

```
Avoid Blurp 환경은 다음과 같이 생성
'''
```

```
env = gym.make(
    id='kymnasium/AvoidBlurp-Normal-v0',
    render_mode='human', # or "rgb_array"
    bgm=True, # or False
    obs_type='custom' # or "image"
)
'''
```

```
여기서부터는 이 환경에 대해서 에이전트를 훈련시키는 코드를
자유롭게 작성
'''
```

## Dependencies

에이전트 구현 시에 사용한 의존성 라이브러리의 목록을 제출해야 합니다.

가상 환경으로 **venv**를 사용하고 있다면, 다음 명령어를 실행하여 생성된 **requirements.txt** 파일로 제출해야 합니다.

```
pip list -format=freeze > requirements.txt
```

가상 환경으로 **conda**를 사용하고 있다면, 다음 명령어를 실행하여 생성된 **environment.yml**을 제출하세요.

```
conda env export > environment.yml
```

## Due Date

- - 2025년 11월 20일 08:59

## Evaluation

### Code

RL Competition은 수업 시간에 다음의 코드를 실행하여 플레이한 결과로 결정됩니다 (별도의 실행 코드는 인정하지 않습니다):

```
import kymnasium as kym
```



```
'''
여러분이 정의하고 학습시킨 에이전트를 불러오는 코드를 넣으세요
예. agent = YourAgent.load('some-path.pkl')
'''

agent = ...

kym.evaluate(
    env_id='kymnasium/AvoidBlurp-Normal-v0',
    agent=agent,
    render_mode='human',
    bgm=True,
    obs_type='custom' # or "image"
)
```

## Grading Criteria



**Time elapsed: 3.80 seconds**

보글보글을 피해 최대한 오래 버티는 게 목표이며, 총 2번 시도하여 경과한 시간의 합이 높은 순으로 순위를 매깁니다. 순위에 따른 점수는 다음과 같습니다:

- 1위 - 2위: 20%
- 3위 - 4위: 17.5%
- 5위 - 6위: 15%
- 7위 - 8위: 12.5%
- 9위 - 10위: 10%
- 11위 - 12위: 7.5%
- 13위: 5%

## 0 Points Policy

단, 다음 중 하나에 해당하는 경우에는 무조건 0% 처리가 됩니다.

- 제출 마감 내에 제출물 (에이전트 구현체 및 requirements.txt)을 이 루리에 제출하지 않은 경우(지연 제출 또한 허용 안됨)

- requirements.txt에서 정의된 모든 의존성 라이브러리를 설치했음에도 불구하고 제출한 코드의 실행이 되지 않는 경우
- 제출한 코드의 실행 결과로 학습된 에이전트가 실제 시연 결과와 크게 다른 경우
- 시연을 하지 않거나 시연이 불가능한 경우 (자신의 팀이 시연을 하는 일자에 출석하지 않는 경우, 해당 학생만 0% 처리)
- 정책 또는 가치 함수를 학습하지 않고, 직접 임의로 지정한 정책 또는 가치 함수를 사용하는 경우
- 환경을 바꾸는 경우

## Leaderboard

| Index | Team Name   | 1st Trial | 2nd Trial | Total | Rank | Remark |
|-------|-------------|-----------|-----------|-------|------|--------|
| 1     | 123         |           |           |       |      |        |
| 2     | Ost         |           |           |       |      |        |
| 3     | Cherry      |           |           |       |      |        |
| 4     | Monte Carlo |           |           |       |      |        |
| 5     | JSL게임개발원    |           |           |       |      |        |
| 6     | wqdsdsf     |           |           |       |      |        |
| 7     | 봉구스밥버거      |           |           |       |      |        |
| 8     | 우주선         |           |           |       |      |        |
| 9     | 유재형         |           |           |       |      |        |
| 10    | 전형규         |           |           |       |      |        |
| 11    | 조우석과 박경범    |           |           |       |      |        |
| 12    | 카페모카        |           |           |       |      |        |
| 13    | 김새연         |           |           |       |      |        |

|    |          |  |  |  |  |  |
|----|----------|--|--|--|--|--|
| 특별 | 청강생들의 반란 |  |  |  |  |  |
|----|----------|--|--|--|--|--|