

DMSC Pybot Tournament

5th edition





Program

Schedule:

- 13:00 - 13:30: Game presentation
- 13:30 - 18:00: Work individually or in teams on your Pybot
- 18:00: Food!
- 18:00 - 19:30: Tournament!

How we run this event:

- Have fun with the challenge
- It is (most probably) possible to cheat, please show good sportsmanship



Disclaimer!

- I made this game because it was January and I was bored ;-)
- I wanted to do something social, and tried to find the quickest way I could have a game ready.
- It is not as polished/tested as the previous games.

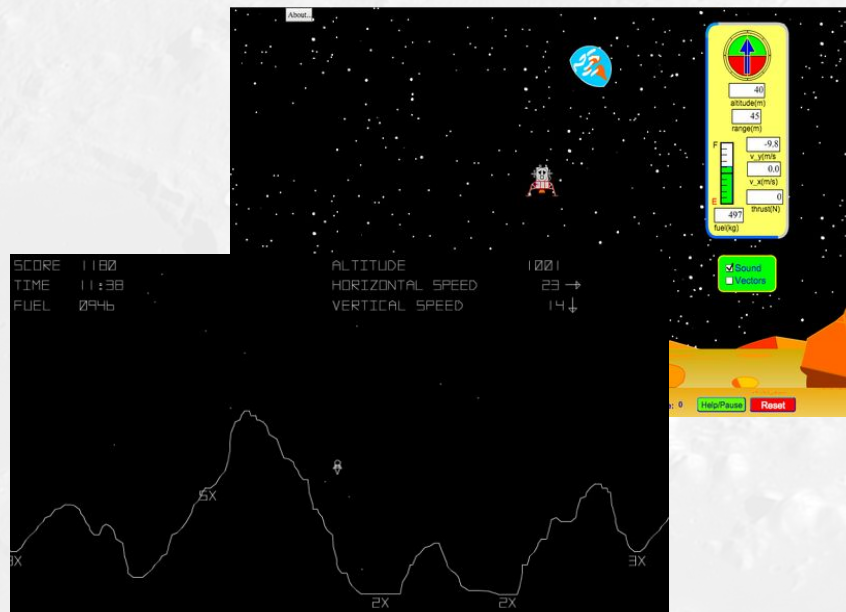
About the game

Inspired by:

- [Lunar Lander](#)

Implementation:

- Python using [Pyglet](#) for the graphics
- ~1000 lines of code



Lunar Lander



Time left: 04:14

Team Apollo 11
x=1489.9, y=800.8
v=[8.5, -0.1]
θ=0.5, fuel=806.9





Game rules (1/4)

Goal:

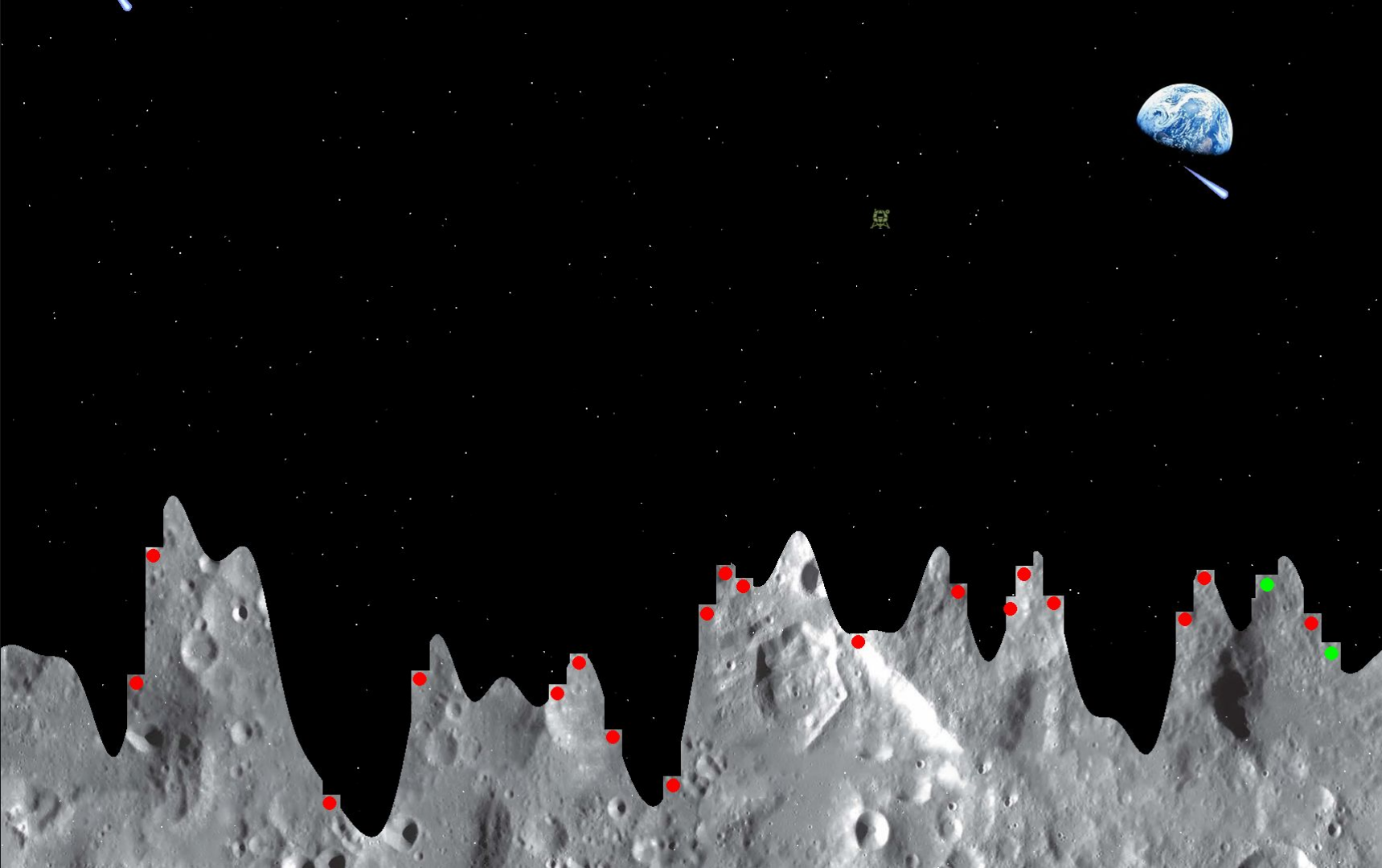
- Land safely on the Moon in less than 5 minutes.
- Start at an altitude of 980m, with a horizontal velocity of 40m/s.

Landing site:

- Initially, there is nowhere you can land!
- Asteroids come down and create craters, which are flat areas in the terrain where you can land.
- Note that the crater made by a single asteroid is not large enough for your ship to land!

Time left: 03:02

Team Apollo 11
x=1086.1, y=806.9
v=[11.6, -0.1]
θ=-0.3, fuel=566.4





Game rules (2/4)

Landing rules:

1. Terrain must be even: LEM size = **25 pixels** (crater size = 24px)
2. Velocity (norm of velocity vector) **< 5 m/s**
3. Landing angle **< 5 degrees**

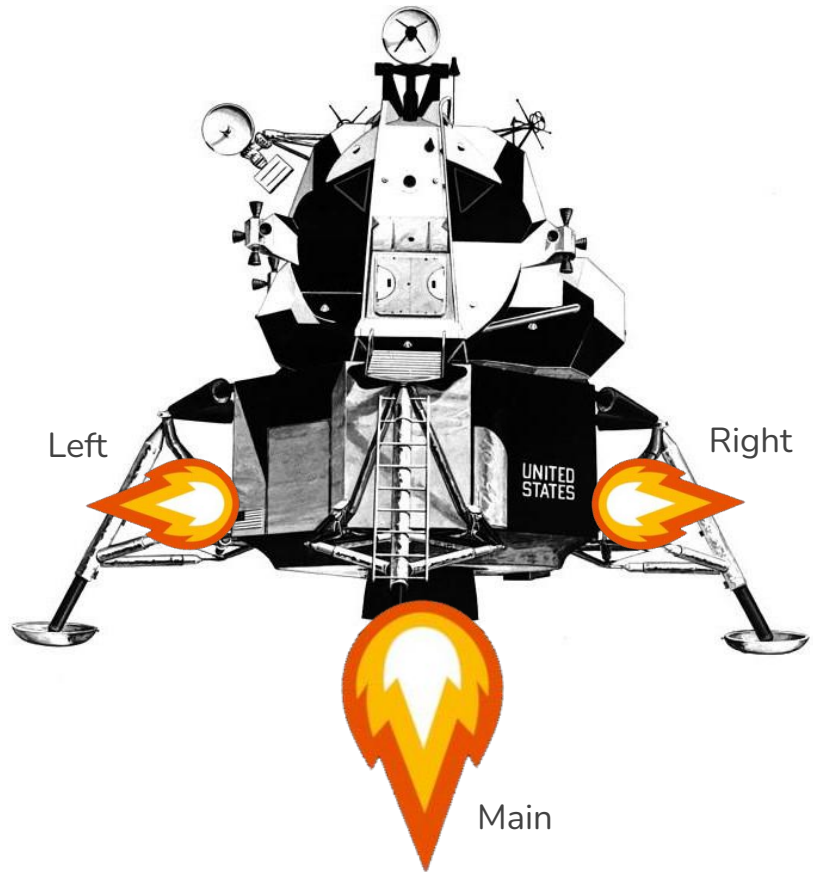
Collisions:

- Asteroids are dangerous: if you hit one you will die!
- You can also collide (elastically) with other players (no damage)

Game rules (3/4)

Thrust:

- The ship has 3 thrusters
- The **Main** thruster provides linear acceleration
- The **Left** and **Right** thrusters provide rotation
- Angular momentum is **not** conserved!





Game rules (4/4)

Scoring:

- 5 points for landing safely
- Time bonus = $5 * \text{time_left} / \text{time_limit}$
- Fuel bonus = $5 * \text{fuel_left} / \text{max_fuel}$
- Landing site bonus = $8 * \text{LEM_size} / \text{landing_site_width}$

Time:

- Time limit is 5 minutes
- There are more and more asteroids as time passes



Demo!

Your bot (1/2)

Information provided every time step:

- `t`: float: current time (in s)
- `dt`: float: the time step (in s)
- `terrain`: np.array: height (y) of terrain
- `players`: dict: listing of all players in the game
- `asteroids`: list: list of all asteroids currently flying

Player info:

- `.team`: team name
- `.position`: player (x, y)
- `.velocity`: player (vx, vy)
- `.heading`: angle (0deg=up)
- `.fuel`: fuel remaining
- `.dead`, `.landed`: status

Asteroid info:

- `.id`: asteroid uuid
- `.position`: (x, y)
- `.velocity`: (vx, vy)
- `.heading`: angle (0deg=up)
- `.size`: size of the tip



Your bot (2/2)

Instructions to return:

Set the thrusters to True or False

- `instructions.main`
- `instructions.left`
- `instructions.right`





Template bot



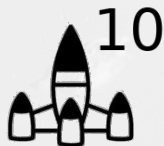


Pick your avatar!

Number from 0 to 17

Or

Path to a png file





Pick your flag!

If your ship successfully lands on the moon, it can deploy a flag.

Use your favourite country code
(e.g. `dk` for Denmark)

Or

Path to a png file



Optimizing development

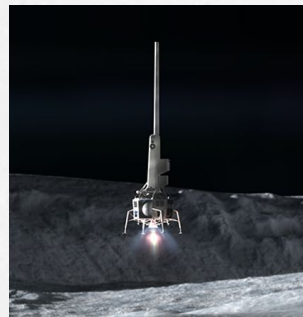
To increase your productivity, you can

1. Play manually with the arrow keys using `manual=True`
2. Increase crater size so you can land earlier with `crater_scaling`
3. Disable player collisions with `player_collisions=False`
4. Disable asteroid collisions with `asteroid_collisions=False`
5. Speed up time with `speedup=2` (does not work that well!)





Get started!



- **Clone the game** from <https://github.com/nvaytet/lunarlander>
- **Create a template repo** from https://github.com/nvaytet/lunarlander_bot
- Start coding (alone or in pairs)
- Tips:
 - Start by just trying to land
 - Worry about landing quickly, other players or asteroids later!
- Tournament will be 10 rounds of 5 minutes (15 min half-time tinkering)

Source: github.com/nvaytet/lunarlander

Template: github.com/nvaytet/lunarlander_bot

```
conda create -n <NAME> -c conda-forge python=3.10.*
```

```
conda activate <NAME>
```

```
git clone https://github.com/nvaytet/lunarlander
```

```
git clone https://github.com/<USERNAME>/<MYBOTNAME>_bot.git
```

```
cd lunarlander/
```

```
python -m pip install -e .
```

```
cd run/
```

```
ln -s ../../<MYBOTNAME>_bot .
```

```
python test.py
```