



# Kötelező Program

## Ütemezés

Okt. hét	Dátum	Számonkérés
6.	okt. 17	Kötelező programok ismertetése. Projekt labor I.
9.	nov. 7	Projekt labor II.
14.	dec. 12	Kötelező programok bemutatása.

## Nehézségi fokozatok és érdemjegyek

A kötelező programok három *nehézségi fokozatban* teljesíthetők. A *nehézségi fok* meghatározza a **legjobb** érdemjegyet, amely a teljesítéséért kapható!

Nehézségi fok	Legjobb megszerezhető érdemjegy
Basic	3
Advanced	4
Epic	5

### Tip

A feladatok úgy vannak megadva, hogy érdemes a **Basic** szinttel kezdeni, és onnan fokozatosan építkezni az **Epic** szintig.

A kötelező programok a következő szempontok szerint kerülnek értékelésre:

- Bizonyítottan saját munka
- Értékelhető eredményeket produkáljon
- Verziókövetés használata, feltöltés GitHub/GitLab/egyéb repoba
- Launch fájlok
- Megoldás teljessége
- Megfelelő ROS kommunikáció alkalmazása
- Program célszerű ROS struktúrája
- Implementáció minősége
- Kód dokumentálása

#### Tip

ChatGPT és egyéb MI eszközök használata megengedett.

## Évközi jegy

A félév elfogadásának feltétele, hogy mind a két ZH, mind a kötelező program értékelése legalább elégséges. A két ZH közül az egyik az utolsó óra alkalmával pótolható.

#### Félév végi jegy

$$\backslash(Jegy = (ZH1 + ZH2 + 2 \backslash times K\o tProg) / 4\backslash)$$

## Kötelező program témák

### 1. Mobil robot

#### A. Playground Robot

- [Gazebo install](#)

- [Setting up a robot simulation \(Gazebo\)](#)



## **B. TurtleBot4**

- [TurtleBot4 Simulator Tutorial](#)
- [TurtleBot4 GUI Docs](#)



### C. PlatypOUs (ROS 1)

- [PlatypOUs GitHub](#)



### D. Bármilyen mobil robot

### 1.1. Mobil robot akadály elkerülés

- **Basic:** Szimulátor élesztése. ROS node/node-ok implementálása szenzorok adatainak beolvasására és a robot mozgására.
- **Advanced:** ROS rendszer implementálása akadály felismerésére és az akadályt kikerülő trajektória tervezésére és megvalósítására szimulált környezetben bármely szenzor felhasználásával.
- **Epic:** Nyűgözz le!

### 1.2. Mobil robot pályakövetés

- **Basic:** Szimulátor élesztése. ROS node/node-ok implementálása szenzorok adatainak beolvasására és a robot mozgására.
- **Advanced:** ROS rendszer implementálása pályakövetésre szimulált környezetben bármely szenzor felhasználásával(pl. fal mellett haladás adott távolságra LIDAR segítségével).
- **Epic:** Nyűgözz le!

### 1.3. Mobil robot objektum követés/visual servoing

- **Basic:** Szimulátor élesztése. ROS node/node-ok implementálása szenzorok adatainak beolvasására és a robot mozgására.
- **Advanced:** ROS rendszer implementálása objektum megkeresésére/felismerésére és követésére/megközelítésére szimulált környezetben bármely szenzor felhasználásával (pl. visual servoing).
- **Epic:** Nyűgözz le!

### 1.4. Mobil robot action library

- **Basic:** Szimulátor élesztése. ROS node/node-ok implementálása szenzorok adatainak beolvasására és a robot mozgására.
- **Advanced:** Egyszerű műveleteket tartalmazó, ROS action alapú könyvtár és ezeket végrehajtó rendszer implementálása (pl. push object, move to object, turn around).
- **Epic:** Nyűgözz le!

## 2. Quadcopter

- [Gazebo install](#)
- [Setting up a robot simulation \(Gazebo\)](#)

```
ign gazebo -v 4 -r quadcopter.sdf
```



- **Basic:** Szimulátor élesztése. ROS node/node-ok implementálása szenzorok adatainak beolvasására és a robot mozgatására.
- **Advanced:** ROS rendszer implementálása magasság/sebesség szabályozására.
- **Epic:** Nyűgözz le!

## 3. Szabadon választott Gazebo szimuláció

- [Gazebo install](#)
- [Setting up a robot simulation \(Gazebo\)](#)
- [Gazebo World Examples](#)

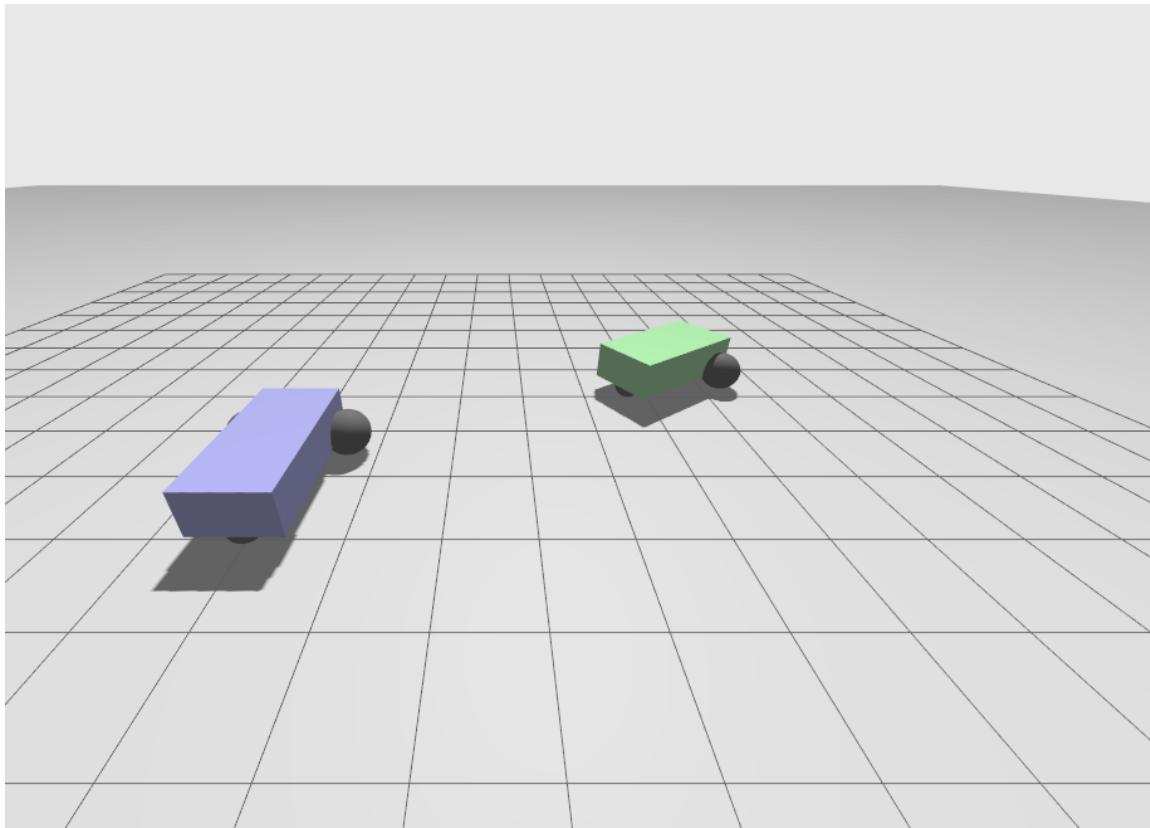


Megegyezés alapján.

#### 4. Gazebo szimuláció összeállítása

- [Gazebo install](#)
- [Setting up a robot simulation \(Gazebo\)](#)
- [Gazebo World Examples](#)





Megegyezés alapján.

## 5. TurtleSim

- [Turtlesim Tutorial](#)
- [Koch Görbe](#)



### 5.1 Turtlesim Fraktál/Szöveg

- **Basic:** Arányos szabályozó implementálása.
- **Advanced:** Fraktál/szöveg rajzolása.
- **Epic:** Nyűgözz le!

## 6. DVRK

- [Download and compile dVRK 2](#)
- [Marker examples](#)



## 6.1 DVRK Interaktív Marker

Megfogható, mozgatható marker implementálása a DVRK szimulátorához.

## 7. YouBot (Windows)

||

- [YouBot controller GitHub](#)

### 7.1. YouBot ROS integráció

- **Basic:** YouBot repo build-elése, megismerése
- **Advanced:** Szimulált robot mozgatása csuklótérben ROS környezetben
- **Epic:** Tesztelés valós roboton és/vagy nyugózz le!

## X. Saját téma

Megegyezés alapján.

## Hasznos linkek

- [Gazebo install](#)
- [Setting up a robot simulation \(Gazebo\)](#)
- [Gazebo World Examples](#)
- [YouBot controller GitHub](#)
- [Download and compile dVRK 2](#)
- [Marker examples](#)
- [Turtlesim Tutorial](#)
- [Koch Görbe](#)