

# Zadání soutěže

Zkontrolujte si stav baterie robota!!!

## Zadání

- Rozsviňte celého robota červeně (diody na gimbalu i na těle). Takto bude svítit po dobu konání následujících úkolů.
- Na začátku s robotem opište rovnostranný trojúhelník, tak aby se robot vrátil do výchozí pozice. První hranu vedte pod úhlem 0° a druhou pod kladným úhlem. Délka hrany bude 60 cm. Robot se bude dívat stále stejným směrem.
- Hledejte markery s čísly a projedte dráhu od 1. k 5. markeru. Markery budete hledat otáčením se o 90° na místě, následně k markeru dorazíte na vzdálenost přibližně 25 cm a budete hledat další marker.
- Až dorazíte k markeru číslo 5 na požadovanou vzdálenost, zamiřte na terč a rozsviňte diodu pod hlavní na 2 sekundy.

## Omezení tratě

- Při hledání nových markerů bude stačit, když se budete otáčet po 90° (nemusíte hledat markery pod jinými úhly)
- Pokud dojedete k následujícímu markeru, tak ze své pozice robot uvidí pouze marker ke kterému dojel a po otočení uvidí právě jeden další

## Bodování

- Za každý splněný úkol dostanete 1 bod (rozsvícená světla, navštívená číslíce)
- Dráha se jede na čas. V případě shodného zisku bodů rozhodne o pořadí čas projetí tratě a kvality zdrojového kódu
- Zdrojový kód bude ohodnocen 0 až 3 body
- Náповěda za -2 body
- Pokud se vám povede projet trať pod 50 sec, tak dostanete bonus 2 bodů, pokud pod 40 sec, dostanete 3 body
- Pokud sedotknete tratě a nebo markeru, je to za postih -1 bod
- Pokud narazíte do tratě nebo do markeru (posunout krabici, nebo shodit marker) -2 body

## Povolené pomůcky

- Můžete používat jakékoliv přechozí lekce a dostupné materiály (ať už odkazované nebo nalezené na internetu)
- Zakázáno je pouze hledat hotové řešení a komunikovat s živými lidmi (robotů se ptát můžete ;) )

## Kód ze kterého můžete vyjít

Pokud chcete, můžete vyjít z připraveného kódu, který si můžete upravit tak, aby odpovídal vašim požadavkům. Pomoci vám může hlavně funkce `dojed_k_markeru()`, která se postará o to, aby robot dojel k markeru, který zrovna vidí.

```

vision_ctrl.enable_detection(rm_define.vision_detection_marker)
robot_ctrl.set_mode(rm_define.robot_mode_chassis_follow)
vision_ctrl.set_marker_detection_distance(3)

markers_recognized = [
    rm_define.cond_recognized_marker_number_one,
    rm_define.cond_recognized_marker_number_two,
    rm_define.cond_recognized_marker_number_three,
    rm_define.cond_recognized_marker_number_four,
    rm_define.cond_recognized_marker_number_five,
]

markers_numbers = [
    rm_define.marker_number_one,
    rm_define.marker_number_two,
    rm_define.marker_number_three,
    rm_define.marker_number_four,
    rm_define.marker_number_five,
]

def dojed_k_markeru():
    '''
    Tuto funkci zavolejte, pokud se chcete robota priblizit markeru, ke kteremu je robot natocen
    Pokud se povede k markeru dojet, funkce vrati True, v opacnem pripade vrati False
    '''

    while True:
        info = vision_ctrl.get_marker_detection_info()
        if not info[0]:
            print("nic nevidim")
            return False
        vzdalenost = info[5]
        cislo = info[1] - 10
        index = cislo - 1

        if vzdalenost < 0.25:
            if vzdalenost > 0.15:
                # marker vidime na vzdalenost mensi nez cca 80 cm, muzeme proto pouzit zamerovac
                # pouziti zamerovace na vetsi vzdalenost neni doporučeno
                vision_ctrl.detect_marker_and_aim(markers_numbers[index])
                if not vision_ctrl.check_condition(markers_recognized[index]):
                    print("robot se ztratil, je potreba znovu spustit vyhledani markeru")
                    return False
            chassis_ctrl.move_with_distance(0, 0.2)
        else:
            return True

```

## Na co si dát pozor

### Python hlásí Syntax error

Dejte si pozor, jestli na řádce s chybou, nebo na předchozí řádce nechybí dvojtečka na konci řádku, pokud se jedná o další blok. Nový blok vytvářejí příkazy například if, while, for, def.

### Python hlásí chybu na řádku, která ani není v programu

Zkontrolujte počty argumentů funkcí (kolik hodnot dáváte do funkce). Např. dáváte do funkce jen jeden vstup a funkce očekává dva.