

# 01LIP – Lineární Programování

## Zápočtový miniprojekt

V případě dotazů: buresj11@cvut.cz

### Hlavní cíl

Cílem miniprojektu je vyzkoušet si formulovat a vyřešit úlohu lineárního či smíšeného celočíselného programování (LP/MILP). Během řešení je potřeba prokázat porozumění převodům mezi standardními tvary, základům duality a citlivostní analýze. Výstupem je **PDF report + zdrojový kód** s krátkou **obhajobou** prokazující porozumění. Alternativně lze odevzdat **Jupyter notebook** (tj. formát používaný na cvičeních), který kombinuje jak formální popis úlohy, tak zdrojový kód.

### Zadání

Zvolte problém inspirovaný praxí (výroba, doprava, směny, směsi, plánování, portfolio, přiřazení, řezání apod.) nebo formulujte vlastní vhodný problém a proveďte následující rozbor:

#### 1. Formulace problému

- Stručný popis kontextu problému, který budete řešit. Vysvětlíte, proč je vhodný pro LP/MILP.
- **Rozměr modelu:** alespoň  $n \geq 8$  rozhodovacích proměnných a alespoň 6 **netriviálních** omezení.
- **Proměnné:** uveďte jejich názvy, jednotky, význam. U MILP případně uveďte, které proměnné jsou binární/celočíselné a proč.
- **Meze:** pro každou proměnnou uveďte dolní/horní mez a volby odůvodněte.
- **Účelová funkce a typ úlohy:** popište optimalizovanou účelovou funkci a definujte typ úlohy, tj. zda jde o maximalizaci nebo minimalizaci, LP vs. MILP.

#### 2. Převod na *maximalizační nerovnicový tvar* (kanonický LP tvar)

- Stručně uveďte nutné kroky: změna znamének, posuny konstant a proměnných, dekompozice volných proměnných, atd.
- **Explicitně запиšte přípustnou množinu** úlohy v tomto tvaru.
- U MILP převeďte pouze LP-relaxaci.

#### 3. Převod na *rovnicový tvar*

- Přidejte pomocné (*slack/surplus*) proměnné podle typu nerovností.
- **Explicitně запиšte přípustnou množinu** úlohy v tomto tvaru.

#### 4. Řešení pomocí kódu

- Pro řešení použijte libovolný dostupný nástroj: Python (Pyomo, PuLP, OR-Tools), Julia (JuMP), MATLAB (linprog/intlinprog), R (lpSolve), AMPL, GAMS, CBC, GLPK, SCIP, Gurobi, CPLEX apod., nebo použijte vlastnoručně naimplementovaný řešič.
- Odevzdejte všechny použitý kód společně s PDF reportem v jednom archivu **.zip**.
- Uveďte optimální řešení nalezené řešičem.

## 5. Duální úloha

- Pro uvažovanou úlohu odvoďte úlohu duální.
- Uveďte význam duálních proměnných v kontextu úlohy.
- U MILP odvoďte duál *LP-relaxace*.
- Pomocí řešiče vyřešte odvozenou duální úlohu, uveďte optimální řešení a okomentujte ho.

## 6. Základní citlivostní analýza

- Zvolte alespoň **dva** scénáře a okomentujte dopad na optimum. Příklady: změna složek vektoru **b** pravé strany (kapacity, poptávky, atp.), změna koeficientů **c** (ceny, marže, atp.), přidání/odebrání omezení, uvolnění/zpřísnění mezí.
- Vhodně prezentujte – tabulkou/grafem, stručnou interpretací.

## 7. Prezentace a obhajoba

- Krátká demonstrace kódu a pipeline řešení. Vysvětlete volbu formulace a převody tvarů.
- Připravte se na dotazy k interpretaci proměnných, aktivním omezením, a k dualitě (komplementární podmínky).

## TL;DR: Co odevzdat

- **PDF report:** stručný, strukturovaný, s rovnicemi a případnými obrázky.
- **Zdrojový kód** použitý k řešení úlohy.

nebo alternativně lze odevzdat

- **Jupyter notebook**, který kombinuje jak formální popis úlohy, tak zdrojový kód.

## Dodatečná pravidla a doporučení

- **Rozsah a složitost:** Vyhněte se řešení zjevně triviálních úloh.
- **Jednotky a škálování:** Uveďte jednotky a případné škálování dat.
- **Spolupráce a zdroje:** Každý student odevzdává samostatnou práci. Externí zdroje a nástroje uveďte v seznamu referencí. AI nástroje využijte v souladu s Rámcovými pravidly používání umělé inteligence na ČVUT.
- **Termín a odevzdání:** Termín odevzdání konzultujte individuálně s přednášejícím.

**Odevzdání a obhájení miniprojektu je nutná podmínka k získání zápočtu a možnosti skládat zkoušku z 01LIP!**