|  |  |
| --- | --- |
| **Univerzita Hradec Králové**  **Fakulta informatiky a managementu**  **Katedra informačních technologií** | |
| **Aplikovaná teorie her**  **Párovací problém přijímání studentů do školy**  **Gale-Shapley algoritmus** | |
| Autor: Martin Malíř | |
|  | |
| Hradec Králové | 04. 2025 |

Obsah

[1. Úvod 4](#_Toc195103383)

[2. Teoretický popis použitého algoritmu 5](#_Toc195103384)

[Princip algoritmu 5](#_Toc195103385)

[Využití v modelu 5](#_Toc195103386)

[Výhoda tohoto přístupu 6](#_Toc195103387)

[3. Přehled proměnných 7](#_Toc195103388)

[Globální proměnné 7](#_Toc195103389)

[Vlastnosti studentů – turtles 7](#_Toc195103390)

[Vlastnosti škol – patches 7](#_Toc195103391)

[4. Popis procedur 8](#_Toc195103392)

[setup 8](#_Toc195103393)

[setup-schools 8](#_Toc195103394)

[setup-students 8](#_Toc195103395)

[go 8](#_Toc195103396)

[assign-students 8](#_Toc195103397)

[calculate-satisfaction 8](#_Toc195103398)

[move-students 8](#_Toc195103399)

[color-schools 8](#_Toc195103400)

[current-matches 9](#_Toc195103401)

[calculate-preference-satisfaction 9](#_Toc195103402)

[update-monitors 9](#_Toc195103403)

[update-score-histogram 9](#_Toc195103404)

[5. Vizualizace a sledování výsledků 10](#_Toc195103405)

[6. Experimenty 12](#_Toc195103406)

[Minimální počet přijímaných studentů 12](#_Toc195103407)

[Poloviční počet přijímaných studentů 13](#_Toc195103408)

[Maximální počet přijímaných studentů 13](#_Toc195103409)

[7. Zdroje 14](#_Toc195103410)

# Úvod

Tento simulační model vytvořený v prostředí NetLogo simuluje proces přidělování studentů do škol na základě jejich preferencí, dosažených bodů a kapacit jednotlivých škol. Model umožňuje sledovat spokojenost studentů i škol, vizualizovat výsledky výběrového řízení a analyzovat distribuci skóre studentů.

# Teoretický popis použitého algoritmu

Model přidělování studentů do škol je inspirován principy Gale-Shapley algoritmu, známého také jako algoritmus stabilního párování. Tento algoritmus byl představen v roce 1962 matematiky Davidem Galeem a Lloydem Shapleym a jeho cílem je nalezení stabilního přiřazení mezi dvěma skupinami, například studenty a školami.

## Princip algoritmu

Každý účastník jedné skupiny (např. student) má seřazený seznam preferencí druhé skupiny (např. škol). Párování probíhá iterativně tak, že:

* Studenti (žadatelé) si vybírají školy podle svých preferencí.
* Školy (příjemci) přijímají žadatele, kteří splňují jejich kritéria (např. bodový limit), dokud nenaplní kapacitu.
* Pokud škola již má plno a přijde lepší uchazeč, může „odmítnout“ méně vhodného kandidáta.
* Tento proces pokračuje, dokud nedojde ke stabilnímu párování, kde žádný student ani škola nemají motivaci změnit rozhodnutí.

## Využití v modelu

V našem modelu není Gale-Shapley algoritmus implementován doslova, ale jeho základní principy jsou použity:

* Studenti mají seřazený seznam preferovaných škol (preferences).
* Při přiřazení student prochází své volby v pořadí a pokouší se o přijetí.
* Škola přijme studenta, pokud má volnou kapacitu a pokud student splňuje minimální požadované skóre.
* Jakmile student najde školu, která jej přijme, přestává hledat – což napodobuje stabilní stav.

Rozdíl oproti klasickému Gale-Shapley algoritmu je v tom, že školy v našem modelu neodmítají méně vhodné studenty ve prospěch lepších, jakmile už mají plno. Přijetí je pouze „první kdo přijde a splní podmínky“. Tím pádem model nevede vždy k optimálnímu stabilnímu párování, ale spíše k realistickému, jednosměrnému výběrovému řízení.

## Výhoda tohoto přístupu

Tato forma přidělování je výpočetně jednodušší a vhodná pro simulace. Umožňuje pozorovat, jak změny v parametrech (např. skóre, kapacitách, preferencích) ovlivňují výsledky, aniž by bylo nutné implementovat plnou rekurzivní logiku klasického algoritmu.

# Přehled proměnných

## Globální proměnné

|  |  |
| --- | --- |
| students | Seznam všech studentů (turtles) |
| schools | Seznam všech škol (patches) |
| students-with-school | Počet studentů, kteří byli přiřazeni do školy |
| students-without-school | Počet studentů bez přiřazené školy |
| first-choice-count | Počet studentů, kteří byli přiřazeni na 1. volbu |
| second-choice-count | Počet studentů, kteří byli přiřazeni na 2. volbu |
| third-choice-count | Počet studentů, kteří byli přiřazeni na 3. volbu |

## Vlastnosti studentů – turtles

|  |  |
| --- | --- |
| preferences | Seznam preferovaných škol |
| score | Bodové ohodnocení studenta |
| assigned-school | Škola, do které byl student přiřazen |
| satisfaction | Spokojenost studenta podle přiřazené volby |

## Vlastnosti škol – patches

|  |  |
| --- | --- |
| capacity | Maximální kapacita školy |
| requirements | Minimální bodová hranice pro přijetí |
| assigned-students | Seznam přiřazených studentů |
| school-satisfaction | Hodnocení školy na základě naplnění a spokojenosti studentů |

# Popis procedur

## setup

Inicializuje celý model: vyčistí plátno, nastaví školy a studenty a resetuje čas (tick).

## setup-schools

Náhodně vybere určitý počet patchů, které budou představovat školy. Každé škole nastaví kapacitu, minimální požadované body a barvu (černou).

## setup-students

Vytvoří požadovaný počet studentů, náhodně jim nastaví skóre, umístění na plátně a seznam tří preferovaných škol.

## go

Hlavní simulační cyklus. Postupně provádí přiřazení studentů, výpočet spokojenosti, pohyb studentů, aktualizace barev škol, aktualizace statistik a vykreslení grafů.

## assign-students

Pro každý nepřiřazený student kontroluje, zda má preferovaná škola volnou kapacitu a jestli student splňuje požadavky. Pokud ano, student je do školy přiřazen.

## calculate-satisfaction

Určuje spokojenost studenta podle toho, na kolikátou preferenci byl přijat (1. – 100 %, 2. – 70 %, 3. – 40 %, žádná – 0 %). Také počítá průměrné skóre a spokojenost škol.

## move-students

Studenti, kteří byli přiřazeni do školy, se k ní začnou přibližovat na plátně.

## color-schools

Vizuálně označí školy změnou barvy.

## current-matches

Funkce, která vrací seznam škol, do kterých byli studenti přiřazeni. Slouží pro export nebo sledování přiřazení.

## calculate-preference-satisfaction

Počítá, kolik studentů se dostalo na svou 1., 2., 3. volbu nebo vůbec žádnou, a vykreslí tyto hodnoty do grafu „Preference Satisfaction“.

## update-monitors

Aktualizuje počítadla a monitory v uživatelském rozhraní (počet studentů se školou, bez školy, rozdělení dle preferencí).

## update-score-histogram

Vykreslí histogram bodového hodnocení studentů podle intervalů po 10 bodech.

# Vizualizace a sledování výsledků

Model využívá dvě základní vizualizace:

* Pozice studentů a škol na plátně
  + Studenti (modré postavičky) se pohybují směrem ke školám (černé čtverce).
* Grafy
  + Preference Satisfaction: rozdělení podle toho, jakou volbu studenti získali.
  + Student Scores: histogram bodového ohodnocení studentů.

Obsah obrázku text, snímek obrazovky, Obdélník, diagram

Obsah vygenerovaný umělou inteligencí může být nesprávný.

Obsah obrázku text, snímek obrazovky, diagram, řada/pruh

Obsah vygenerovaný umělou inteligencí může být nesprávný.

# Experimenty

**Jak moc jsou školy schopny uspokojit preference studentů?**

## Minimální počet přijímaných studentů

Obsah obrázku text, snímek obrazovky, diagram, řada/pruh

Obsah vygenerovaný umělou inteligencí může být nesprávný.

## Poloviční počet přijímaných studentů

Obsah obrázku text, snímek obrazovky, diagram, řada/pruh

Obsah vygenerovaný umělou inteligencí může být nesprávný.

## Maximální počet přijímaných studentů

Obsah obrázku text, snímek obrazovky, diagram, Obdélník

Obsah vygenerovaný umělou inteligencí může být nesprávný.

# Zdroje

[1] SINGH, Anmolika. Online. BuiltIn. 2024. Dostupné z: <https://builtin.com/articles/gale-shapley-algorithm>. [cit. 2025-04-09].

[2] BHARGAV, Nikhil. *The Stable Marriage Problem*. Online. Baeldung. 2024. Dostupné z: <https://www.baeldung.com/cs/stable-marriage-problem>. [cit. 2025-04-09].