**Žilinská univerzita v Žiline**

**Fakulta riadenia a informatiky**

**Diskrétna simulácia**

Semestrálna práca 1

**Bc. David Kučera**

Akademický rok 2024/25, letný semester

Obsah

[Zadanie a požiadavky práce 3](#_Toc192274624)

[Architektúra práce 4](#_Toc192274625)

[Modul DSLib 4](#_Toc192274626)

[SimCore 4](#_Toc192274627)

[Jan 5](#_Toc192274628)

[EmpiricalGenerator 5](#_Toc192274629)

[DiscreteEmpirical 5](#_Toc192274630)

[ContinousEmpirical 5](#_Toc192274631)

[UniformGenerator 5](#_Toc192274632)

[DiscreteUniform 5](#_Toc192274633)

[ContinousUniform 5](#_Toc192274634)

[Modul MonteCarloLib 5](#_Toc192274635)

[Modul MonteCarloVisualizer 5](#_Toc192274636)

[Vyhodnotenie experimentov jednotlivých stratégií 8](#_Toc192274637)

[Návrh vlastných stratégií 8](#_Toc192274638)

[Obrázok 1 Diagram modulov 4](#_Toc192274646)

[Obrázok 2 Okno po behu štandardných stratégii 6](#_Toc192274647)

[Obrázok 3 Okno po behu mojich stratégii 7](#_Toc192274648)

[Obrázok 4 Okno po behu 1 replikácie 7](#_Toc192274649)

# Zadanie a požiadavky práce

**Ján chce minimalizovať náklady za 30 týždňov prevádzky skladu súvisiace so skladovaním súčiastok (zahrnutá je aj pokuta za nedodanie) a rozhoduje sa medzi stratégiami objednávania:**

* Stratégia A: Objednávanie každý týždeň od dodávateľa 1,
* Stratégia B: Objednávanie každý týždeň od dodávateľa 2,
* Stratégia C: Objednávanie každý nepárny týždeň od dodávateľa 1 a každý párny týždeň od dodávateľa 2.
* Stratégia D: Objednávanie každý nepárny týždeň od dodávateľa 2 a každý párny týždeň od dodávateľa 1

**Úloha 1:** Ktorú stratégiu má Ján použiť, ak začína s prázdnym skladom a vždy objednáva rovnaké počty súčiastok?

**Úloha 2:** Navrhnite a podrobne odôvodnite vlastnú stratégiu, pričom môžete ľubovoľne meniť dodávateľov ale aj objednané množstvá jednotlivých typov súčiastok. Pripravte teda 30 objednávok (postačí tabuľka) a pre každú uveďte ktorému dodávateľovi bude zadaná. Nie je bezpodmienečne nutné nájsť optimálne

riešenie, ale Vaše najlepšie riešenie, ktorého postup získania rozumne odôvodnite. Využite metódu Monte Carlo.

**Postup a požiadavky:**

Vytvorte model uvedenej situácie a s využitím metódy **Monte Carlo** vykonajte s týmto modelom experimenty tak, aby ste boli schopní zodpovedne odpovedať na položenú otázku. Výsledky experimentov vypisujte prehľadnou formou na obrazovku tak, aby ste správnosť Vašej činnosti mohli ľahko preukázať.

Zobrazte **graficky na grafe v programe v priebehu modelovania** ustaľovanie nákladov na skladovanie **(zahrnutá je aj pokuta za nedodanie)** pre jednotlivé stratégie pri zvyšujúcom sa počte opakovaní experimentu (pre každú stratégiu urobte samostatný graf). Zabezpečte (napr. pomocou vhodných nastavení), aby grafy mali čo najväčšiu čitateľnosť zobrazovaných dát a mali aj reálny prínos pre užívateľa **podľa** **pokynov** **z cvičenia**. Simulácia sa musí dať predčasne zastaviť a zobraziť dosiahnuté výsledky. Pozastavenie a spomalenie nie je potrebné implementovať.

Pre spustenie jedinej simulácie vykreslite do grafu vývoj nákladov na skladovanie **(zahrnutá je aj pokuta za nedodanie)** **po jednotlivých dňoch**.

**Implementujte všeobecné jadro pre statické modelovanie metódou Monte Carlo**. Pri implementácií semestrálnej práce dodržte oddelenie užívateľského prostredia od jadra aplikácie. V tejto semestrálnej práci je na generovanie čísel dovolené používať iba v danom jazyku štandardné knižnice (napr. triedu Random v jazyku java a C#). **Naprogramujte vlastný flexibilný generátor spojitého a diskrétneho empirického rozdelenia podľa pokynov z cvičenia**.

# Architektúra práce

Práca bola implementovaná v programovacom jazyku **C#** na platforme **.NET 8.0**. Beží len na OS Windows. Aplikácia je rozdelená do 3 hlavných programových modulov.

Prvým najnižším modulom je knižnica **DSLib**, kde je implementácia všeobecného jadra pre statické modelovanie metódou Monte Carlo a vlastné flexibilné generátorypre spojité a diskrétne empirické a rovnomerné rozdelenia. *Pozn. Funkčnosť generátorov bola riadne overená v externej aplikácii.*

Ďalším modulom je modul **MonteCarloLib**, ktorý obsahuje jadro aplikácie. Využíva modul DSLib a vykonáva nad ním operácie pre prácu s jednotlivými stratégiami simulácií – experimentov.

Posledným najvyšším modulom je **MonteCarloVisualizer**, ktorý obsahuje GUI rozhranie pre prácu s modulom MonteCarloLib.

Obrázok, na ktorom je text, písmo, rad, snímka obrazovky

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Obrázok 1 Diagram modulov

## Modul DSLib

Táto knižnica obsahuje triedy generátorov a simulačného jadra Monte Carlo.

Obrázok, na ktorom je text, snímka obrazovky, rad, písmo

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

### SimCore

Trieda obsahuje všeobecné simulačné jadro spolu so všetkými potrebnými metódami. Vykonáva n replikácii, ak nie je predčasne pozastavená.

### Jan

Potomok triedy SimCore, obsahuje model problému a jednotlivé stratégie. Používa generátory popísané nižšie. Pomocou eventov posiela priebežne hodnoty vyššie (kumulatívne náklady a náklady za jednotlivé dni).

### EmpiricalGenerator

Potomok triedy Random, kontroluje zadané parametre a uchováva generátory pre jednotlivé intervaly – rozdelenia.

### DiscreteEmpirical

Potomok triedy EmpiricalGenerator, prekrýva metódu Next pre generovanie diskrétnych hodnôt z daných intervalov.

### ContinousEmpirical

Potomok triedy EmpiricalGenerator, prekrýva metódu NextDouble pre generovanie spojitých hodnôt z daných intervalov.

### UniformGenerator

Potomok triedy Random, berie minimum a maximum intervalu a uchováva generátor.

### DiscreteUniform

Potomok triedy UniformGenerator, prekrýva metódu Next pre generovanie diskrétnych hodnôt z intervalu <min, max).

### ContinousUniform

Potomok triedy UniformGenerator, prekrýva metódu NextDouble pre generovanie spojitých hodnôt z intervalu <min, max).

## Modul MonteCarloLib

Tento modul obsahuje triedu reprezentujúcu „jadro“ aplikácie, ktoré sa využíva v GUI. Trieda obsahuje metódy na štart experimentov, zastavenie vykonávania experimentov a obsahuje mechanizmy (eventy) pre poslanie priebežných výsledkov experimentov na GUI.

## Modul MonteCarloVisualizer

Tento modul obsahuje GUI rozhranie pre grafické zobrazenie experimentov na grafe. Obsahuje jediné okno, ktoré sa skladá z dvoch častí – konfiguračná časť a časť obsahujúca 4 grafy.

V konfiguračnej časti je možné nastaviť tieto parametre:

* počet replikácií,
* násadu generátorov,
* vykresliť len každý n-tý bod do grafu,
* preskočiť prvých n percent hodnôt.

Taktiež sa v tejto časti nachádzajú tlačidlá pre spustenie a zastavenie simulácie. Tlačidlo „Spusti moje stratégie“ spustí do grafov mnou navrhnuté stratégie popísané neskôr v dokumente.

V druhej časti okna je možné vidieť graf stratégie a aktuálnu hodnotu danej simulácie. Okno vyzerá nasledovne:

Obrázok, na ktorom je text, snímka obrazovky, rad, vývoj

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.

Obrázok Okno po behu štandardných stratégii

*Obrázok, na ktorom je text, snímka obrazovky, rad, vývoj

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.*

Obrázok Okno po behu mojich stratégii

*Obrázok, na ktorom je text, snímka obrazovky, vývoj, rad

Obsah vygenerovaný umelou inteligenciou môže byť nesprávny.*

Obrázok Okno po behu 1 replikácie

*Pozn. grafy boli vytvorené v  pomocou knižnice ScottPlot – voľne dostupná na* [*https://scottplot.net*](https://scottplot.net)*.*

# Vyhodnotenie experimentov jednotlivých stratégií

V nasledovnej tabuľke je prehľad výsledkov pre jednotlivé stratégie A,B, C a D a pre rôzne počty replikácií. Ako násada generátorov v GUI bola použitá hodnota 0.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| P.rep.\Strat. | A | B | C | D |
| 100 000 | **23 075** | **11 633** | **15 938** | **15 688** |
| 1 000 000 | **23 093** | **11 627** | **15 954** | **15 687** |
| 10 000 000 | **23 081** | **11 625** | **15 949** | **15 688** |

Na základe týchto hodnôt, môžeme jednoznačne prehlásiť, že z poskytnutých stratégií je stratégia B najvhodnejšia.

## Návrh vlastných stratégií

Mnou navrhnuté stratégie sú nasledovné:

* Stratégia 1: 1. až 15. týždeň zásobuje dodávateľ 1, zvyšné týždne (16. až 30.) zásobuje dodávateľ 2,
* Stratégia 2: 1. až 15. týždeň zásobuje dodávateľ 2, zvyšné týždne (16. až 30.) zásobuje dodávateľ 1,
* Stratégia 3: 1. až 8., 16. až 23. týždeň zásobuje dodávateľ 1, zvyšné týždne (9. až 15. a 24. až 30.) zásobuje dodávateľ 2,
* Stratégia 4: 1. až 8., 16. až 23. týždeň zásobuje dodávateľ 2, zvyšné týždne (9. až 15. a 24. až 30.) zásobuje dodávateľ 1.

Výsledky daných stratégií sú v nasledovnej tabuľke. Ako v predchádzajúcom prípade, aj tu bola použitá ako násada generátorov v GUI hodnota 0.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| P.rep.\Strat. | 1 | 2 | 3 | 4 |
| 100 000 | **11 799** | **11 783** | **11 767** | **11 782** |
| 1 000 000 | **11 771** | **11 778** | **11 773** | **11 779** |
| 10 000 000 | **11 772** | **11 773** | **11 771** | **11 775** |

Na základe vykonaných experimentov je možné zhodnotiť navrhnuté stratégie ako takmer rovnaké.