



Modelování a simulácie

**Výrobní proces z oblasti:
elektrotechnické nebo potravinárske**

Michal Šajdík (xsajdi00)
Petr Medek (xmedek07)

Obsah

1	Úvod	3
1.1	Autori a zdroje informácií.....	3
1.2	Validita modelu	3
2	Rozbor témy a použitých metód/technológií.....	3
2.1	Výsledný model systému.....	3
2.2	Zhrnutí informácií z pozorovania.....	3
2.3	Použité technológie	4
3	Koncepcia	4
3.1	Konceptuálny model	4
3.2	Petriho sieť	5
3.2.1	Proces generovania zákazníkov	5
3.2.2	Proces výroby	6
3.3	Koncepcia – implementačné detaily	7
4	Architektúra simulačného modelu/simulátora	7
5	Experimenty	8
5.1	Experiment 1	8
5.2	Experiment 2	8
5.3	Experiment 3	9
5.4	Experiment 4	9
5.5	Experiment 5	10
5.6	Experiment 6	10
5.7	Experiment 7	11
6	Záver	11

1 Úvod

Práce je zameraná na simuláciu fast foodu, ktorá sa zameriava na prípravu kebabu. Príprava jedla je modelovaná pomocí systému hromadné obsluhy. Zamerali sme sa na aktuálny počet zamestnancov, aby sme zistili, či je ich počet optimálny, alebo nie. Pre simuláciu boli zvolené 2 smeny: denní a noční.

1.1 Autori a zdroje informácií

Práce bola vypracovaná vo dvojici: Michal Šajdík a Petr Medek. Všetky potrebné zdroje informácií pre technické účely boli nazhromaždené z prednášok IMS a z dokumentácie SYMLIB. Údaje o výrobe sme nazhromaždili hlavne z role „Zákazník“, kde sme v dané roli zhromažďovali informácie behom našich „obedových“ a „večerných“ prestávok v nami zvolenom fast foode“. Dodatočné informácie o funkcionalite fast foodu sme sa dozvedeli od tunajších zamestnancov, vzhľadom k tomu, že nás už poznali ako častých a pravidelných zákazníkov, tak nebol problém získať informácie podstatné pre validáciu nášho modelu. Jedinú informáciu ktorú sme si od zamestnancov nedozvedeli bola výška ich platu, túto informáciu sme určili podľa informácií dostupných na internetu (porovnali sme mzdy 4 rôznych fast foodu) na 110Kč/hod pre zamestnancov a 150Kč/hod pre vedúceho (v predajni sa vždy vyskytuje jeden vedúci a ostatní sú zamestnanci).

1.2 Validita modelu

Validitu modelu sme overili porovnaním získaných údajov zo strany fast foodu a našimi nasimulovanými hodnotami. Naše a ich hodnoty síce rovnaké neboli, ale mali len zanedbateľné odchýlky. Danému javu sme prisudzovali fakt, že „dokonalý“ model a simuláciu je ťažko dosiahnuteľná a zanedbateľná pre náš prípad.

2 Rozbor témy a použitých metód/technológií

2.1 Výsledný model systému

Fast food je zameraný na výrobu kebabu, ktorého výroba zaberie 2.5 minúty a zaplatení zaberie 20 sekúnd. Výrobu a zaplatení môže previesť jeden zamestnanec, ale tiež každú akciu môže vykonať zamestnanec zvlášť.

Podľa nášho pozorovania sme si všimli, že do fast foodu chodia ľudia s exponenciálnym rozložením so stredom 2.0 minúty v dennej smene a s 1.2 minúty v nočnej smene.

2.2 Zhrenutí informácií z pozorovania

Tieto informácie sme využili k testovaniu validity modelu a vytvoreniu samotného simulačného modelu systému:

- Vo fast foodu pracujú 3 zamestnanci, ako v dennej smene, tak v nočnej smene
- Spracovanie jednej objednávky trvá 2.5 minút
- Platenie jedného zákazníka trvá 20 sekúnd
- Priemerný počet spracovaných objednávok za deň „553“
- Dĺžka jednej smeny je 8 hodín

- Pripravenie masa trvá 30 sekúnd a je pripravené pre 5 porcií
- Pripravenie zeleniny trvá 5-6 minút a je pripravené 40-70 kusov zeleniny

2.3 Použité technológie

- C++
- Knižnica SIMLIB – pre prácu s procesmi a eventami
- Princíp Petriho Sietí pre návrh nášho simulačného modelu

3 Konceptcia

Ako bolo už spomenuté vyššie, tak naším hlavným cieľom je zistiť optimálny počet zamestnancov pre danú smenu.

Fast food sa zameriava aj na dovoz jedla, ale k tomu nám neboli poskytnuté údaje pre validáciu, ale aspoň nám bolo povedané, že objednávky s dovozom, pripravujú zamestnanci v rámci „voľného času“, alebo si danú objednávku pripraví ten, ktorý ju aj odvezie. Aj z toho dôvodu nám vzniká menšia odchýlka v našom modeli na rozdiel od reality.

3.1 Konceptuálny model

Na začiatku simulácie zamestnanci už majú pripravených 80 kusov zeleniny, keď počet kusů spadne na 40 a jeden zo zamestnancov nemá nič iného na robote, tak ide pripraviť zeleninu a následne ju pridá.

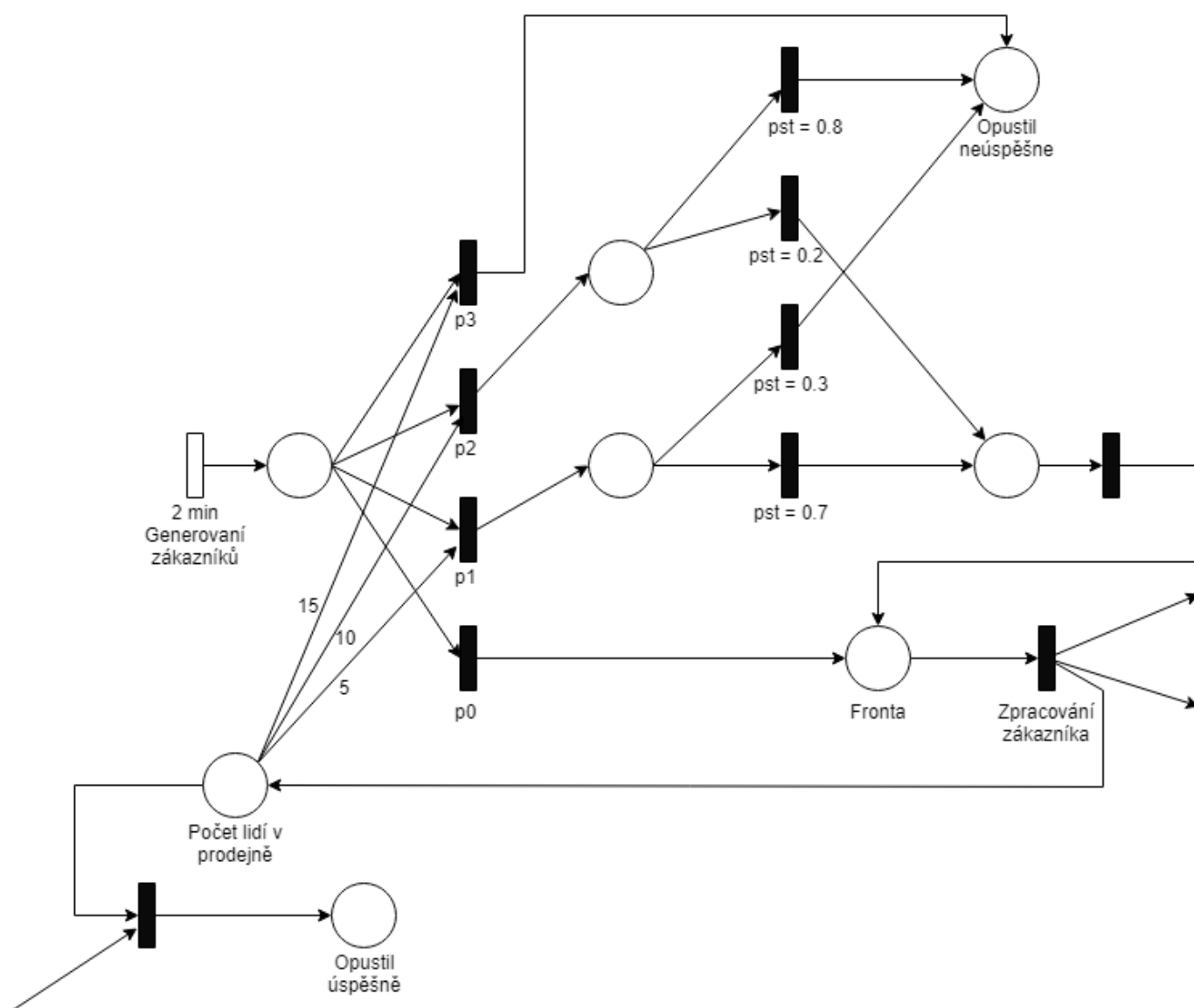
Pokiaľ je počet kusov 10 alebo menej, tak prvý voľný zamestnanec ide pripraviť zeleninu a následne ju pridá. Do našej objednávky je potrebné aj nakrájať mäso, ktoré sa krája, až keď už žiadne iné nie je. V prípade že nie je mäso a už ho niekto krája, tak ďalší zamestnanec ide k pokladni a vyberie peniaze za objednávku. Po pridaní mäsa sa pridá zelenina, ktorá by už mala byť pripravená.

Následne sa objednávka zapečie, v prípade že zákazník už zaplatil, tak dostane kebab a odchádza úspešne, zamestnanec sa posúva k ďalšiemu zákazníkovi, v prípade, že zákazník ešte nezaplatil, tak daný zamestnanec ide ku kase a vykoná inkasovanie danej peňažnej sumy.

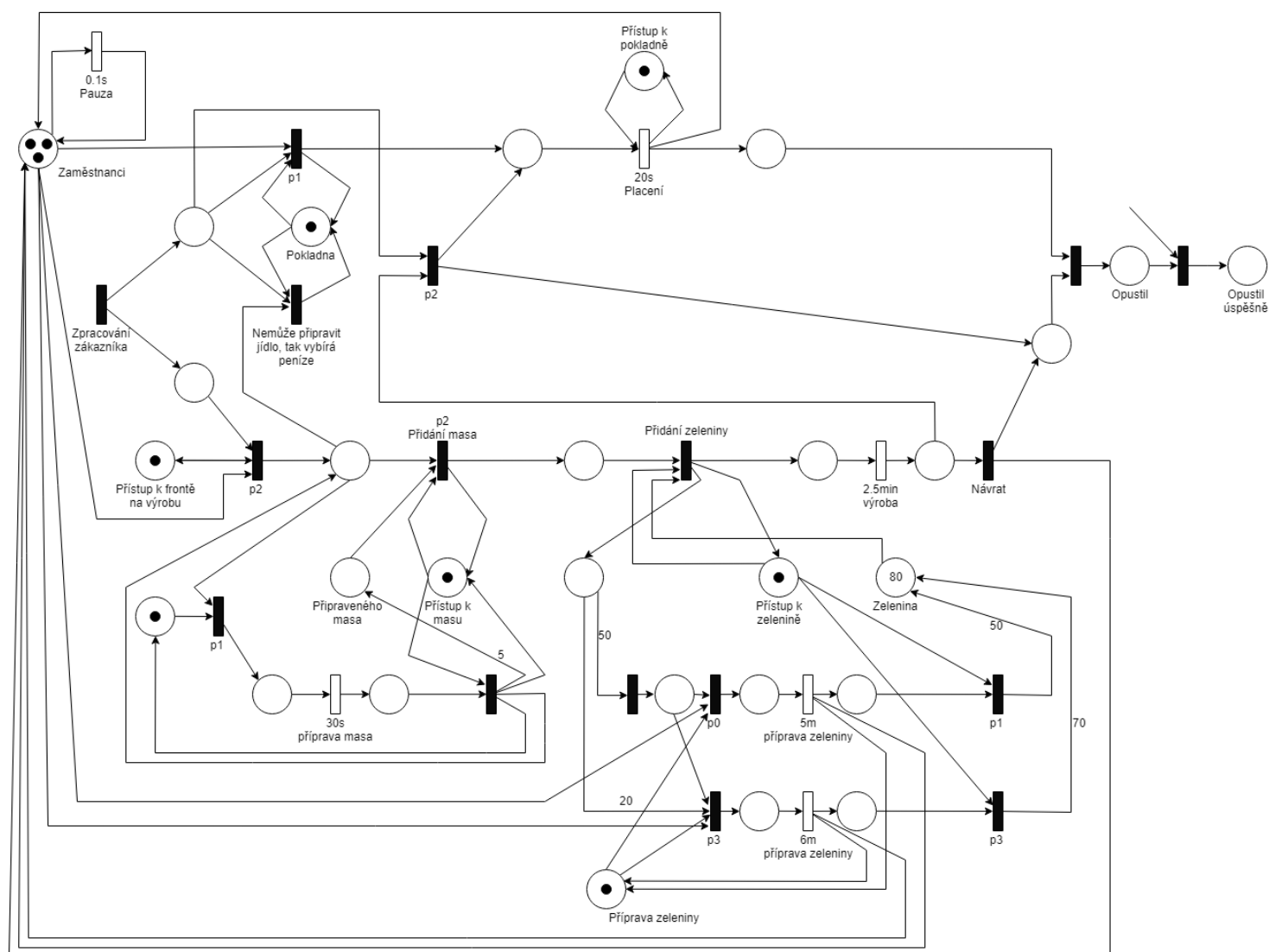
Exponenciálne prichádzajú zákazníci do fast foodu podľa toho či je denná, alebo nočná smena.

3.2 Petriho sieť

3.2.1 Proces generovania zákazníkov



3.2.2 Proces výroby



3.3 Koncepcia – implementačné detaily

Príchod nových zákazníkov, nám zaisťuje trieda, ktorá dedí z triedy **Event**. Udalosť je aktivovaná pri zahájení simulácie, ktorý raz za daný čas spustí proces, ktorý rozhodne o príchodu alebo odchodu zákazníka.

Pri vygenerovaní zákazníka je pravdepodobnosť, že sa zákazník rozhodne kvôli veľkej fronte odísť a nič si neobjedná. Podľa veľkosti fronty sa mení hrana pravdepodobnosti odchodu zákazníka. Pre získanie náhodnej hodnoty sme použili metódu **Random()**. V prípade že je 0 ľudí vo fronte tak zákazník vždy vstúpi, ak 5 a viac ľudí, tak je 30% pravdepodobnosť, že odíde. V prípade 10 a viac ľudí je 80% pravdepodobnosť na odchod a v prípade že tam je 15 ľudí, tak už zákazníci odchádzajú vždy. Daná funkcionality je implementovaná v triede, ktorá dedí z knižnice tried **Process**.

Zamestnancov sme implementovali pomocou dátového typu **Store**, pomocou ktorého si môžeme jednoducho určiť počet našich zamestnancov a daná trieda za nás aj rieši prístup k ich počtu. Máme implementovanú triedu „Výroba“, ktorá dedí z knižnice tried **Process**. Danú triedu spustíme na začiatku simulácie toľko krát, koľko máme zamestnancov.

Nami využívané zariadenia ako boli: vstupne dvere, prístup k mäsu, prístup k zelenine, nástroj pre prípravu mäsa, nástroj pre prípravu zeleniny, atď. boli implementované pomocou dátovej triedy **Facility**. Princíp prístupu a odchodu z „facility“ nám poslúžilo ako spoľahlivé zámky, a preto sme nami potrebné hodnoty, ktoré sa menili počas celej doby simulácie ukladali do globálnych premenných, ktoré boli na konci simulácie vypísané.

Vzhľadom k tomu, že v rôznych experimentoch a simuláciách sme menili len niektoré hodnoty a niekedy zasa veľké množstvo, tak sme sa rozhodli nechať hodnoty nastavené pomocou makier na začiatku zdrojového kódu.

4 Architektúra simulačného modelu/simulátora

Trieda *class GeneratorZakazniku* : *public Event* - zaisťuje generáciu zákazníkov (**Zakaznik**)

Trieda *class Zakaznik* : *public Process* - zaisťuje vstup, alebo neúspešný odchod zákazníka

Trieda *class Vyroba* : *public Process* - zaisťuje priebeh výroby a platenie od zákazníka

5 Experimenty

Experimenty obsahujú ukážky výstupu našej simulácie s rôznymi vstupnými hodnotami pre každý experiment. *Počet zamestnancov* a *smena* sú vstupné položky pomocou, ktorých zistíme *Zisk*. *Zisk*, ale nie je jediná výstupná informácia z našej simulácie, ale tiež tam sú aj ostatné hodnoty, ktoré pomáhajú k analýze situácie. *Zisk* podniku sme vypočítali jednoducho pomocí vzorce:

$$\begin{aligned} \text{ZISK} = & \text{Opustil úspešne} * \text{průměrný výdělek} - \\ & \frac{\text{délka simulace}}{60} * \text{zaměstnanci} * \text{plat zaměstnance} - \\ & \frac{\text{délka simulace}}{60} * \text{vedoucí} * \text{plat vedoucího} \end{aligned}$$

Podľa zisku podniku porovnáваме či sa vyplatí ďalší zamestnanec alebo nie.

5.1 Experiment 1

Experiment 1 bol pridaný iba pre ukážku toho, ako veľmi dokáže byť podnik neefektívny iba s jedným zamestnancom, a ako neefektívne to dokáže byť voči ostatným experimentom reprezentujúcich vyšší počet zamestnancov v rámci nočnej smeny.

Počet zamestnancov	1
Smena	Noc

Opustil úspešne	179
Fronta na výrobu	8
Fronta na platenie	9
Opustil neúspešne	205
Celkom opustilo	384
Zisk	3658.06 Kč

5.2 Experiment 2

Pridali sme jedného zamestnanca a rozdiel v zisku počas danej smeny bol jednoznačný. Podľa našej analýzy za to jednoznačne môže počet úspešne opustenie predajne, čo pre daný podnik znamená zisk.

Počet zamestnancov	2
Smena	Noc

Opustil úspešne	322
Fronta na výrobu	3
Fronta na platenie	4
Opustil neúspešne	30
Celkom opustilo	352
Zisk	6659.08 Kč

5.3 Experiment 3

Po pridání ďalšieho zamestnanca sa nám zasa zvýšil zisk.

Počet zamestnancov	3
Smena	Noc

Opustil úspešne	372
Fronta na výrobu	0
Fronta na platenie	0
Opustil neúspešne	12
Celkom opustilo	384
Zisk	7136.08 Kč

5.4 Experiment 4

Na rozdiel od „experimentu 3“, po pridání 4. zamestnanca sa nám zníži zisk, preto lebo nepríde dosť ľudí na to aby sa im tam 4. zamestnanec vyplatil.

Počet zamestnancov	4
Smena	Noc

Opustil úspešne	371
Fronta na výrobu	0
Fronta na platenie	0
Opustil neúspešne	8
Celkom opustilo	379
Zisk	6228.94 Kč

5.5 Experiment 5

Ak porovnáme „experiment 5“ a „experiment 1“, tak vieme povedať, že počas dňa chodí menej ľudí, a práve preto je možné, že na dennú smenu budeme potrebovať, iný počet zamestnancov.

Počet zamestnancov	1
Smena	Deň

Opustil úspešne	179
Fronta na výrobu	7
Fronta na platenie	8
Opustil neúspešne	140
Celkom opustilo	319
Zisk	3658.06 Kč

5.6 Experiment 6

Podobne ako v „experimente 2“ sa nám *zisk* zväčšil skoro o dvojnásobok, len kvôli pridaniu jedného zamestnanca čo malo za následok drastického zvýšenia úspešného spracovania objednávok.

Rozdiel v zisku pre „experiment 6“ a „experiment 7“ bol občas nepatrný a preto sme vykonali viac simulácií pre dané parametre.

Počet zamestnancov	2
Smena	Deň

Opustil úspešne	309
Fronta na výrobu	0
Fronta na platenie	0
Opustil neúspešne	12
Celkom opustilo	321
Zisk	6306.26 Kč

5.7 Experiment 7

Pomocou „experimentu 7“ môžeme ukázať, že pridaním 3. zamestnanca by sme iba znížili zisk simulovanej predajne. Parametre pre „experiment 7“ sme použili viacero krát a zaujímavosťou je, že občas prišlo dostatok zákazníkov, aby sa nám oplátilo mať 3 zamestnancov, namiesto 2 zamestnancov. Po prevedení viacerých experimentov sme sa dostali k záveru, že počet zamestnancov počas dennej smeny je najziskovejší v počte 2.

Počet zamestnancov	3
Smena	Deň

Opustil úspešne	318
Fronta na výrobu	1
Fronta na platenie	1
Opustil neúspešne	7
Celkom opustilo	325
Zisk	5670.52 Kč

6 Záver

Z experimentov 1 – 4 môžeme posúdiť, že optimálny počet zamestnancov pre nočnú smenu je 3. Daná hodnota zodpovedá počtu zamestnancov, ktorý tam pracujú v skutočnosti.

Z experimentov 5 – 7 môžeme posúdiť, že optimálny počet zamestnancov pre dennú smenu je 2. Daná hodnota je o 1 menšia ako počet zamestnancov, ktorý tam pracujú v skutočnosti. Z toho dokážeme posúdiť, že by pre firmu bolo vhodné znížiť stav zamestnancov počas dňa, alebo sa pokúsiť o navýšenie počtu zákazníkov.