ZÁPADOČESKÁ
UNIVERZITA
V PLZNI

Semestrální práce z KIV/VSP

Simulační model systému hromadné obsluhy

Martin Sloup, A08N0111P
msloup@students.zcu.cz

Obsah

1 Zadání

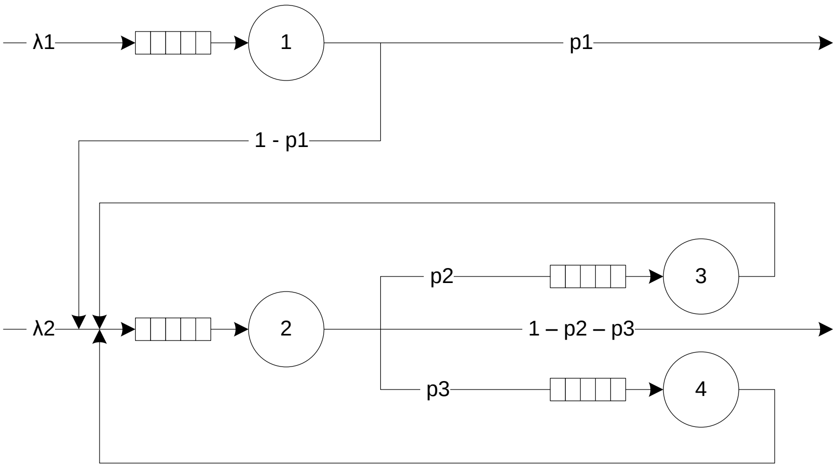
1.1 Obecné zadání

1. Vymyslete si otevřenou síť front (tj. propojení, vstupní proudy) obsahující alespoň 4 obslužné uzly (jeden kanál, fronta FIFO, neomez. délka), alespoň 2 vstupní proudy požadavků, alespoň 2 vnitřní zpětné vazby.
2. Parametry sítě (tj. střední frekvence vstupních proudů, střední doby obsluhy v jednotlivých kanálech a p-ti větvení) zvolte tak, aby síť pracovala ve stacionárním režimu. Doporučená hodnota zatížení pro všechny uzly: $\rho > 0{,}5$.
3. Určete výpočtem střední frekvence toků v uzlech. Dále určete veličiny Lq_i a Tq_i pro jednotlivé uzly a Lq a Tq pro celou síť pro případ, že všechny vstupní toky jsou Poissonovské a doby obsluhy ve všech uzlech mají exponenciální rozdělení.
4. Vypočtené hodnoty ověřte vlastnoručně vytvořeným simulačním programem. Použijte simulační knihovnu C-Sim nebo J-Sim.
5. Dále uvažujte případ, kdy všechny náhodné časové intervaly v modelu (příchody, obsluhy) mají Gaussovské pravděpodobnostní rozdělení $N(a, \sigma)$ s (různou) střední hodnotou zvolenou v bodě 2. Vytvořte generátor tohoto rozdělení jako funkci v jazyce **C** nebo **Java** (s parametry např. a, σ) a testováním ověřte správnou funkci generátoru - chce se tedy po Vás vytvoření a prokázání správné funkce generátoru, který napíšete VY - využít můžete pouze knihovní funkce pro generování rovnoměrného rozdělení (jako v průběžném příkladu č. 1). Použití knihovní funkce pro Gaussovo rozdělení z J-SIMu se nepočítá!
6. Simulací ověřte chování sítě (tj. určete stejné veličiny jako v bodech 3) a 4) pro případ, že všechna rozdělení (příchody, obsluhy) budou mít hustotu $N(a, \sigma)$ se stejnou střední hodnotou jako pro exponenciální rozdělení **alespoň pro 3 různé hodnoty** koeficientu variace $C = \frac{\sigma}{a}$. Pokuste se o poměrně odlišné koeficienty, ať je vidět rozdíl v chování systému (např. 0.05 - tj. skoro konstantní generátor, 0.2 a 0.7 - ale to je pouze příklad). **Poznámka**: Simulační program je stejný jako v bodě 4, ale volá se jiný generátor podle bodu 5.
7. Simulační program upravte pro sledování dalších individuálně zadaných výkonnostních charakteristik sítě.
8. Řešení zpracujte formou písemného referátu (cca 10 stran, grafy, tabulky, barevné obrázky, hudební vložky, multimédia ap. - berte to jako přípravu na diplomku, navíc vlastní referát lze využít při zkoušce).

1.2 Konkrétní zadání

Na obrázku 1. se nachází konkrétní zadání semestrální práce. Jedná se o simulaci polikliniky a nemocnice. Očíslované uzly představují následující reálné objekty:

1. Obvodní lékař
2. Chirurgie
3. Sádrovna
4. Rentgen



Obrázek 1: Konkrétní zadání SHO

Hodnoty proměnných na obrázku:

$$\lambda_1 = 5 \quad \lambda_2 = 5 \quad p_1 = \frac{3}{5} \quad p_2 = \frac{1}{5} \quad p_3 = \frac{2}{5}$$

Doby obsluhy uzlů:

$$\mu_1 = 6 \quad \mu_2 = 18 \quad \mu_3 = 4 \quad \mu_4 = 9$$

1.2.1 Další sledované veličiny

Při schválení zvoleného SHO bylo rozšířeno zadání o sledování dalších statistik. Konkrétně se jedná o statistiku fronty uzlu č. 2. Statistika fronty zahrnuje získání následujících veličin během simulace:

- střední délku fronty
- rozptyl délky fronty
- směrodatnou odchylku délky fronty
- histogram

2 Teoretický úvod

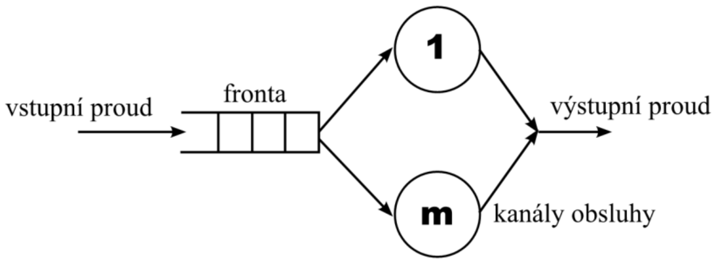
2.1 Systémy hromadné obsluhy

Pojmem *systémy hromadné obsluhy* (*SHO*) se vyznačují modely reálných systémů, jejichž funkce spočívá v realizaci *obsluhy* (poskytují *služby*) pro velké počty paralelně přicházejících *požadavků* (*transakcí*). Služby poskytují prvky nazývané *kanály obsluhy*. Před kanálem se obvykle vytváří *fronta* vstupních požadavků požadavků.

Významnou vlastností je abstrakce od sémantiky služby, modeluje se pouze časová posloupnost přicházejících požadavků a jejich časová náročnost na poskytované služby. Kvůli velkým počtům vstupních požadavků je využit *pravděpodobnostní popis* časových charakteristik proudu požadavků.

2.1.1 Elementární SHO

Konceptuální model systému hromadné obsluhy je na obrázku 2. Stavem systému zde figuruje celkový počet požadavků akumulovaných v systému (jak ve frontě, tak i v obslužných kanálech).



Obrázek 2: Elementární systém hromadné obsluhy

Primárním předpokladem pro následující popis elementárního SHO je *stacionární režim činnosti systému*, tj. časová neměnnost zdrojových statistických charakteristik.

2.1.2 Vstupní proud požadavků

Nyní předpokládáme, že požadavky vstupují do systému náhodně v časové posloupnosti $\{t0 < t1 < t2\ldots\}$. Náhodná veličina $\tau = t_k - k_{k-1}$ se pak nazývá *interval příchodů* a znamená velikost časového intervalu mezi vstupem předchozího požadavku a následujícího požadavku.

Nejčastěji je vstupní proud zadán pomocí *distribuční funkce pravděpodobnostního rozdělení* $Fa(t) = P\{\tau \leq t\}$ nebo také odpovídající *hustotou pravděpodobnosti*. Nejobvyklejším typem vstupního proudu u SHO je *poissonovský vstupní proud*. Pro tento proud je interval příchodů definován pomocí *exponenciálního rozdělení*. Dále lze vstupní proud popsat následujícími vztahy:

λ – střední frekvence příchodů

$T_a = \frac{1}{\lambda}$ – střední perioda příchodů (střední hodnota intervalu mezi příchody).

$F_a(t) = 1 - e^{-\lambda t}$ – distribuční funkce časových intervalů mezi příchody v poissonovském proudu

$C_a = \frac{\sigma\{\tau\}}{T_a}$ – koeficient variance, kde $\sigma\{\tau\}$ je směrodatná odchylka intervalu mezi příchody (udává nahodilost příchodů, pro pravidelné má hodnotu 0, pro poissonovské hodnotu)

2.1.3 Fronta požadavků

Frontu lze charakterizovat *maximální délkou* a *frontovou disciplínou*. Délka může být jak omezená, tak i neomezená. Nejčastější frontovou disciplínou je FIFO, kdy se vstupní požadavky zařazují na konec fronty a požadavky ke zpracování se vybírají ze začátku fronty. Frontu je možné popsat následovně:

w – okamžitý počet požadavků ve frontě

L_w – střední délka fronty, tj. středí počet požadavků ve frontě

t_w – doba čekání jednoho konkrétního požadavku ve frontě

T_w – střední doba čekání požadavků ve frontě

2.1.4 Kanály obsluhy

Nejobvyklejší je jednokanálová obsluha. Počet kanálů je označen písmenem *m*. Doba obsluhy jednoho konkrétního požadavku t_s je *náhodná veličina*. Základní charakteristikou této náhodné veličiny je distribuční funkce uvažovaného rozdělení $F_s(t) = P\{t_s \leq t\}$. Dále platí následující vzorce:

$T_s = \frac{1}{\mu}$ – střední doba obsluhy (veličina μ představuje střední frekvenci obsluh, tj. střední počet požadavků obsloužených za jednotku času (je-li kanál permanentně zatížený))

$F_s(t) = 1 - e^{-\mu t}$ – distribuční funkce exponenciálního rozdělení doby obsluhy

$C_s = \frac{\sigma\{t_s\}}{T_s}$ – koeficient variace doby obsluhy ($\sigma\{t_s\}$ je směrodatná odchylka doby obsluhy); vyjadřuje nahodilost obluh (pro konstatní má hodnotu 0, pro exponenciální máhodnotu 1)

2.1.5 Vztahy pro elementární SHO jako celek

Elementární SHO jako pro celek je možné charakterizovat pomocí následujících vztahů a veličin:

q – okamžitý celkový počet požadavků v SHO (tj. ve frontě i v kanálech obsluhy)

L_q – střední celkový počet požadavků v SHO

t_q – doba odezvy (doba průchodu celým systémem pro jeden konkrétní požadavek)

T_q – střední doba odezvy

Zatížení kanálů Nutnou podmínkou pro dosažení *stacionárního režimu* je hodnota zatížení $\sigma < 1$.

$$\sigma = \frac{1}{m} \cdot \frac{Ts}{Ta} = \frac{1}{m} \cdot \frac{\lambda}{\mu} \qquad (1)$$

Veličina $u = \frac{\lambda}{\mu}$ se nazývá *intenzita provozu* a k ní nejbližší vyšší celé číslo představuje maximální počet obslužných kanálů potřebných pro zajištění stacionárního režimu činnosti SHO.

Je zřejmé, že dále platí následující:

$$L_q = Lw + L_s = L_w + m \cdot \frac{\lambda}{\mu}$$

$$T_q = T_w + T_s \ = T_w + \frac{1}{\mu}$$

Littleovy vzorce

$$L_q = \lambda T_q, \qquad L_w \ = \ \lambda T_w \qquad (2)$$

2.1.6 M/M/1

Nyní budeme uvažovat jednokanálové SHO. Vstupní proud požadavků je poissonovský, jež je charakterizovatelný parametrem λ. Tento parametr odpovídá střední frekvenci proudu a je i parametrem exponenciální rozdělení

$$F_a(t) = 1 - e^{-\lambda t}.$$

Doba obsluh pro M/M/1 má také exponenciální rozdělení $F_s(t) = 1 - e^{-\mu t}$. Zde parametr μ je zároveň střední frekvence doby obsluhy. Pro dosažení stacionárního režimu musí samozřejmě platit $\sigma = \frac{\lambda}{\mu} < 1$.

Další vzorce pro veličiny L_q, T_q a T_s:

$$Lq = \frac{\rho}{1-\rho} \qquad \text{pro } \sigma < 1 \qquad (3)$$

$$T_q = T_w + T_s, \quad T_s = \frac{1}{\mu} \qquad (4)$$

2.2 Otevřené sítě front

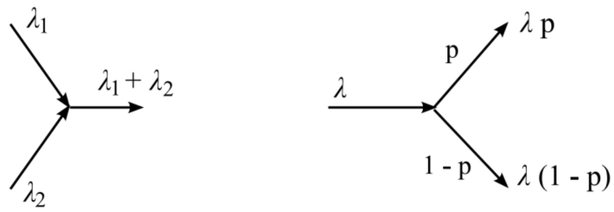
Systém s více kanály obsluhy, kde každý kanál obsluhy má vlastní frontu, je možné popsat jako síť, jejíž prvky jsou elementární SHO. Sítě s explicitními vstupy toků z vnějšího prostředí se nazývají *otevřené*.

Otevřenou síť front je možné znázornit orientovaným grafem. Počet elementárních SHO v síti označíme jako n, číslování uzlů bude v intervalu $i \in \langle 1, n \rangle$. Váhami hran se jsou obvykle *pravděpodobnosti větvení*, tj. hrana vedoucí z uzlu i do uzlu j je vážena pravděpodobností p_{ij} přechodu do j po ukončení obsluhy v i. Samozřejmě celkový součet vah hran vycházejících z uzlu i musí být roven jedné.

2.2.1 Analýza středních frekvencí a zatížení

Střední frekvenci vnitřního toku uzlem i označíme jako Λ_i. Frekvence Λ_0 odpovídá celkového toku požadavků vstupujících z okolí do systému. Jsou-li požadavky po dokončení obsluhy i-tého uzlu rozděleny podle pravděpodobnosti větvení p_{ij}, bude možné střední frekvence toku požadavků z uzlu i do j spočítat jako

$$\lambda_{ij} = \Lambda_i \cdot p_{ij}.$$



Obrázek 3: Pravidla pro sloučení a rozdělení toků

Také platí, že vstupu i-tého uzlu se sčítají frekvence toků z hran přicházejících do uzlu i. Pro stacionární režim platí následující vzorec:

$$\sum_k \Lambda_k p_{ki} = \Lambda_i = \sum_j \Lambda_i p_{ij} \qquad (5)$$

V tomto vzorci index k odpovídá vstupním hranám a index j výstupním hranám uzlu i. Jedná se o tzv. *rovnici kontinuity* toku uzlem. Tato rovnice říká, že součet frekvencí vstupních toků je roven frekvenci vnitřního toku a zároveň součtu frekvencí všech vystupujících hran.

K určení hodnot středních frekvencí všech toků v síti musí být známa hodnota frekvence Λ_0 a matice pravděpodobnosti větvení $\{p_{ij}\}$. Hodnoty frekvencí Λ_i je možné vypočítat vyřešením soustavy n lineárních rovnic pro uzly 1 až n podle vzorce 5.

Výpočtem středních frekvencí vnitřních toků je možné ověřit stacionaritu systému. K tomuto výpočtu je nutné pro každý uzel i také znát počet kanálů m_i a střední dobu obsluhy T_{s_i}. Zatížení uzlu je pak

$$\rho_i = \frac{1}{m} \cdot \Lambda_i T_{s_i} < 1 \qquad \text{pro } i = 1 \dots n \qquad (6)$$

2.2.2 Určení délky front a doby odezvy

Doposavad nebylo uvažováno pravděpodobnostní rozdělení vstupu požadavků do systému a jejich obsluhy. Je zřejmé, že střední frekvence toků a zatížení uzlů na pravděpodobnostním rozložení nezávisí. Naopak u středních délek front a dob odezev narůstají hodnoty s nepravidelností intervalů příchodů požadavků. Matematické řešení systému je možné určit pouze při stanovení následujících pravidel (tzv. *Jacksonův teorém*):

- Všechny vstupní toky mají poissonovský charakter.
- Všechny obslužné uzly mají exponenciální rozdělení doby obsluhy se střední hodnotou T_{s_i}.
- Po ukončení obsluhy v uzlu i přechází požadavek s pravděpodobností p_{ij} do uzlu j bez zpoždění.

Pak na jednotlivé uzly je možné pohlížet jako na elementární SHO a lze aplikovat vzorce z 2.1.5. Pro celou síť platí následující vzorec pro výpočet požadavků L_q akumulovaných v celé síti

$$L_q = \sum_{i=1}^{n} L_{q_i} \qquad (7)$$

a pro střední dobu T_q průchodu požadavku

$$T_q = \frac{1}{\Lambda_0} \cdot L_q \qquad (8)$$

Pokud ale nejsou splněny stanovené předpoklady a některé příchody či obsluhy jsou pravidelnější než exponenciální (koeficienty variance jsou menší než jedna), je možné uvedený postup použít k analýze nejhoršího případu (*worst case analysis*).

2.3 Generování náhodných čísel

2.3.1 Generování čísel s exponenciálním rozdělení

Při simulaci je potřeba generovat čísla s exponenciálním rozdělením. Vzorec pro vytváření takových čísel získáme inverzní transformací distribuční funkce tohoto rozdělení. Použijeme tedy distribuční funkci:

$$F(x, \lambda) = 1 - e^{-\lambda x}$$

a transformujeme ji na inverzní funkci ($x = F^{-1}(y)$) ve tvaru:

$$x = -\frac{1}{\lambda} ln(1 - y)$$

ale vzhledem k tomu, že soubor prvků 1 − y a y má stejné normované rovnoměrné rozložení, lze vzorec upravit na

$$x = -\frac{1}{\lambda} ln(a).$$

Za a již pak stačí dosadit hodnotu z generátoru normalizovaného rozdělení (např. Math.random() z JDK Javy).

2.3.2 Generování čísel s normálním rozdělením

Normální (taktéž gaussovské) rozdělení je definováno jako

$$\aleph(\mu, \sigma),$$

kde μ je střední hodnota a σ je směrodatná odchylka.

Pro generování využijeme tzv. *centrální limitní větu*. Tato věta tvrdí, že součet náhodných čísel s libovolným rozdělením má asymptoticky normální rozdělení se střední hodnotou a rozptylem danými součtem středních hodnot a rozptylů jednotlivých rozdělení prvků součtu. Součtem získáme hodnoty v intervalu $\langle 0, 12 \rangle$. Odečteme polovinu intervalu, tedy hodnotu 6. Nyní již stačí interval $\langle -6, 6 \rangle$ roztáhnou pomocí směrodatné odchylky a posunout střed intervalu pomocí střední hodnoty. Použijeme tedy vzorec:

$$x = u + \sigma \left(\left(\sum_{j=1}^{12} y_j \right) - 6 \right)$$

3 Matematický výpočet

3.1 Střední doba obsluhy

Ze zvolených hodnot v zadání nejprve spočítáme střední dobu obsluhy na uzlech. Tu potřebujeme dále ve výpočtu.

$$T_{s_1} = \frac{1}{\mu_1} = \frac{1}{6} = 0{,}1\overline{6}$$

$$T_{s_2} = \frac{1}{\mu_2} = \frac{1}{18} = 0{,}0\overline{5}$$

$$T_{s_3} = \frac{1}{\mu_3} = \frac{1}{4} = 0{,}25$$

$$T_{s_4} = \frac{1}{\mu_4} = \frac{1}{9} = 0{,}\overline{1}$$

3.2 Střední frekvence toků v uzlech

Dle vzorce 5 sestavíme soustavu algebraických rovnic pro výpočet střední frekvence toků v uzlech:

$$\Lambda_A = \lambda_1 = 5$$

$$\Lambda_B = \lambda_2 = 5$$

$$\Lambda_1 = \Lambda_A = 5$$

$$\Lambda_2 = \Lambda_B + (1 - p_1)\Lambda_1 + \Lambda_3 + \Lambda_4 = \Lambda_B + (1 - p_1)\Lambda_1 + p_2\Lambda_2 + p_3\Lambda_2 = 5 + \frac{2}{5}5 + \frac{1}{5}\Lambda_2 + \frac{2}{5}\Lambda_2$$

$$= 5 + 2 + \frac{3}{5}\Lambda_2 = 7 + \frac{3}{5}\Lambda_2 \Rightarrow \Lambda_2 = \frac{35}{2} = 17{,}5$$

$$\Lambda_3 = p_2\Lambda_2 = \frac{1}{5} \cdot \frac{35}{2} = \frac{7}{2} = 3{,}5$$

$$\Lambda_4 = p_3\Lambda_2 = \frac{2}{5} \cdot \frac{35}{2} = 7$$

3.3 Zatížení uzlů

Provedeme kontrolu stacionarity systému, tedy že $\rho < 1$:

$$\rho_1 = \Lambda_1 T_{s_1} = 5 \cdot \frac{1}{6} = 0{,}8\overline{3}$$

$$\rho_2 = \Lambda_2 T_{s_2} = \frac{35}{2} \cdot \frac{2}{36} = 0{,}97\overline{2}$$

$$\rho_3 = \Lambda_3 T_{s_3} = \frac{7}{2} \cdot \frac{1}{4} = 0{,}875$$

$$\rho_4 = \Lambda_4 T_{s_4} = 7 \cdot \frac{1}{9} = 0{,}\overline{7}$$

Nyní již víme, že náš zvolený systém je stacionární. Výše uvedený výpočet nám udává také zatížení jednotlivých uzlů systému.

3.4 Střední počet požadavků v uzlech

Použitím vzorce 3 spočítáme střední počet požadavků v jednotlivých uzlech.

$$L_{q1} = \frac{\rho_1}{1-\rho_1} = \frac{0{,}8\overline{3}}{1-0{,}8\overline{3}} = 5$$

$$L_{q2} = \frac{\rho_2}{1-\rho_2} = \frac{0{,}97\overline{2}}{1-0{,}97\overline{2}} = 35$$

$$L_{q3} = \frac{\rho_3}{1-\rho_3} = \frac{0{,}875}{1-0{,}875} = 7$$

$$L_{q4} = \frac{\rho_4}{1-\rho_4} = \frac{0,\overline{7}}{1-0,\overline{7}} = 3{,}5$$

3.5 Střední počet požadavků v celé síti

Pro celou síť určíme průměrný počet požadavků L_q akumulovaných v celé síti podle vzorce 7.

$$L_q = \sum_i L_{q_i} = L_{q1} + L_{q1} + L_{q1} + L_{q1} = 5 + 35 + 7 + 3{,}5 = 50{,}5$$

3.6 Doba odezvy uzlů

Na všech uzlech spočítáme dobu odezvy užitím vzorce 8 a získáme:

$$T_{q1} = \frac{L_{q1}}{\Lambda_1} = \frac{5}{5} = 1$$

$$T_{q2} = \frac{L_{q2}}{\Lambda_2} = \frac{35}{\frac{35}{2}} = 2$$

$$T_{q3} = \frac{L_{q3}}{\Lambda_3} = \frac{7}{\frac{7}{2}} = 2$$

$$T_{q4} = \frac{L_{q4}}{\Lambda_4} = \frac{3{,}5}{7} = 0{,}5$$

3.7 Doba odezvy celého systému

Součtem doby odezvy, nebo také užitím jednoho z Littleových vzorců získáme celkovou dobu odezvy systému.

$$T_q = \frac{L_q}{\Lambda} = \frac{L_q}{\lambda_1 + \lambda_2} = \frac{50{,}5}{5 + 5} = 5{,}05$$

4 Programové řešení simulace

Během vývoje aplikace pro simulaci *SHO* nebylo možné použit předchozí semestrální práce. V semestrální práci č. 1 bylo prováděno počítání trojúhelníkového rozdělení, jež v celkové semestrální práci nelze použit. Je totiž nutné generovat čísla v exponenciálním a normálovém (Gauss) rozdělení. To samé platí pro semestrální práci č. 4, kde nevyhovuje způsob napojení uzlů. Většinu programového kódu byla napsána od začátku. V následující kapitole jsou sepsány důležité informace o jednotlivých třídách použitých při simulaci.

4.1 Balík cz.zcu.kiv.vsp.msloup.generator

Jak již název napovídá, v tomto balíku se nacházejí jednotlivé generátory čísel příslušných rozdělení. Všechny generátory čísel v této semestrální práci používají rozhraní *IGenerator*, které definuje jedinou metodu *nextValue* – vrácení dalšího náhodného čísla. Balík obsahuje 3 generátory čísel: Exponenciální, normálové a rovnoměrné spojité rozdělení. První dvě generují náhodná čísla podle vzorců uvedených v kapitole Generování náhodných čísel. Poslední (třetí) třída je pouze úpravou standardního generátoru z JDK Javy. Tato úprava umožňuje roztažení původního intervalu $(0, 1\rangle$ na interval $\langle a, b\rangle$.

4.2 Balík cz.zcu.kiv.vsp.msloup.object

V tomto balíku se nacházejí třídy na vytvoření simulovaného prostředí (kromě *PatientGenerator*, který se nachází v dalším balíku). Předně se jedná o třídy: *Choice*, *Room* a *Terminátor*. Třída Choice slouží k vytvoření větvení v simulaci na základě pravděpodobnosti. Přidáním větve je u této třídy dosaženo metodou *addChoice* s uvedenou pravděpodobností a cílem, kam větvení vede. Cíl musí implementovat rozhraní *IRequestInput*, viz dále. Například *Choice* právě toto rozhraní implementuje, může být takovým cílem. Toto rozhraní implementuje též i třída *Room*, která je v simulaci jako by uzel systému. Třída *Room* musí mít definovaný generátor čísel pro získání doby zpracování pacienta (požadavku) v systému. Taktéž je vhodné nastavit cíl, kam bude pacient po zpracování předán. Třída *Room* obsahuje i metodu *setStatistics*, pomocí níž je možné přidat počítání statistiky. Nakonec třída *Terminator* slouží k ukončení toku pacientů v systému. Tato třída je v simulaci použita na detekci, zda již všichni vygenerovaní pacienti prošli simulací. Také implementuje rozhraní *IRequestInput*.

4.3 Balík cz.zcu.kiv.vsp.msloup.patient

Balík obsahuje třídu *Patient* (požadavek v systému) a generátor požadavků *PatientGenerator*. *PatientGenerator* na základě dodaného objektu generátoru náhodných čísel generuje objekty *Patient* v simulaci a předává ho cíli. Tento cíl je nastaven pomocí metody *setOutput* a musí implementovat již známé rozhraní *IRequestInput*. Třída *Patient* je pouhou implementací rozhraní *IRequest*, kde je definována metoda *getInputTime* vracející čas vytvoření požadavku. Tento čas je potřebný při generování statistik.

4.4 Balík cz.zcu.kiv.vsp.msloup.request

V balíku se nacházejí pouze dvě rozhraní a to již známé *IRequestInput* a *IRequest*. *IRequestInput* obsahuje pouze metodu *receiveRequest*. Pomocí ní přijímá uzel požadavek od předchozího uzlu.

4.5 Balík cz.zcu.kiv.vsp.msloup.statistic

Obsahuje třídy, které pro *Room* počítají statistiku. Všechny třídy implementují rozhraní *IStatistics*, které obsahuje metodu na zpracování statistiky pro jednoho pacienta (jeden požadavek). Balík obsahuje jak základní statistiku *BasicStatistics* počítající:

- střední délku fronty (L_w)
- průměrnou střední dobu odezvy serveru (T_q)
- součet středních dob odezvy serveru (místnosti)
- zatížení serveru (místnosti)
- počet zpracovaných požadavků

tak i třídu pro výpočet statistiky fronty *QueueStatistics*. Tato třída počítá kromě již zmíněné základní statistiky také různé statistiky ohledně fronty u místnosti. Tedy:

- střední délku fronty
- rozptyl délky fronty
- směrodatnou odchylku délky fronty
- maximální délku fronty
- minimální délku fronty
- histogram

4.6 Balík cz.zcu.kiv.vsp.msloup.test

Seskupuje třídy pro testování vytvořených generátorů rozdělení a tvorbu histogramu těchto rozdělení. Třída *Histogram* tedy počítá histogram a další hodnoty pro visuální kontrolu správnosti řešení. Ostatní třídy *ExponentialTest, GaussianTest* a *UniformTest* slouží jako vstupní body ke spuštění testů.

4.7 Balík cz.zcu.kiv.vsp.msloup.util

V tomto balíku se nachází pouze jedna třída, a to třída *Counter*. Ta je vhodná pro tvorbu čítače, jenž je společný např. pro dva generátory požadavků a zaručuje tak, že snižování bude provedeno atomicky.

4.8 Balík cz.zcu.kiv.vsp.msloup

Obsahuje pouze dvě Třídy. Třídu *Main*, ke spuštění celé simulace a třídu *Config* s nastavením parametrů.

5 Kompilace a spuštění testovacího prostředí

Dodávaná testovací aplikace je již přeložena, ale pokud by ji přesto z nějakých důvodů bylo nutné znova zkompilovat. Lze to provést pomocí skriptu vytvořeném v *Apache Ant*. Stačí v adresáři, kde je soubor build.xml spustit následující příkaz:

```
ant
```

Tento příkaz spustí kompilaci programu, vytvoření dokumentace **API** (pomocí *Javadocu*) a následné zabalení do distribučního balíčku jar. Vytvoření dokumentace API lze také provést samostatně přidáním parametru *javadoc* za příkaz ant.

Spuštění programu se provede v adresáři *dist* buď pomocí příkazu `run.bat [parametry]` (run.sh na OS Linux), nebo příkazem:

```
java -Djava.util.logging.config.file=logging.properties -jar vsp.jar [parametry]
```

Parametrem programu je počet kroků a typ generátoru náhodných čísel. Tedy **„EXP“** pro exponenciální generátor čísel pro generování pacientu a dobu obsluhy u jednotlivých místností nebo **„GAUSS“** pro generátor čísel s normálovým (Gaussovo) rozdělením pro koeficienty 0,05; 0,3 a 0,7. Pokud nejsou tyto dva parametry zadány, spustí se výchozí nastavení a postupně i oba generátory. Výchozí nastavení počtu požadavků je 10 000.

6 Výsledek simulace

Následující tabulka obsahuje jednotlivé výsledky z povinných statistik. Histogram statistiky fronty se nacházení v příloze.

	Lq1	Lq2	Lq3	Lq4	Lq	Tq1	Tq2	Tq3	Tq4	Tq	Statistika fronty Chirurgie		
											E{x}	D{x}	σ{x}
Exp. - výpočet	5	35	7	3.5	50.5	1	2	2	0.5	5.05	---	---	---
Exp. - simulace	5.01102	34.7921	6.99275	3.51856	50.3144	0.98752	2.04613	1.95947	0.50124	5.11382	35.8287	1171.16	34.2223
Gauss. - C=0.05	0.0584	1.02805	0.03935	0.11666	1.24246	0.27588	0.27347	3.40934	0.38911	2.29112	0.46277	0.54837	0.74052
Gauss. - C=0.3	0.10611	1146961	0.04758	0.24953	1146961	1.47799	1016198	1.00856	2.46947	1774517	234815	1.8E+10	135355
Gauss. - C=0.7	0.10526	1027472	0.04664	0.24964	1027473	5.78582	3014927	2.30599	5.75355	5269478	298298	3E+10	172153

7 Závěr

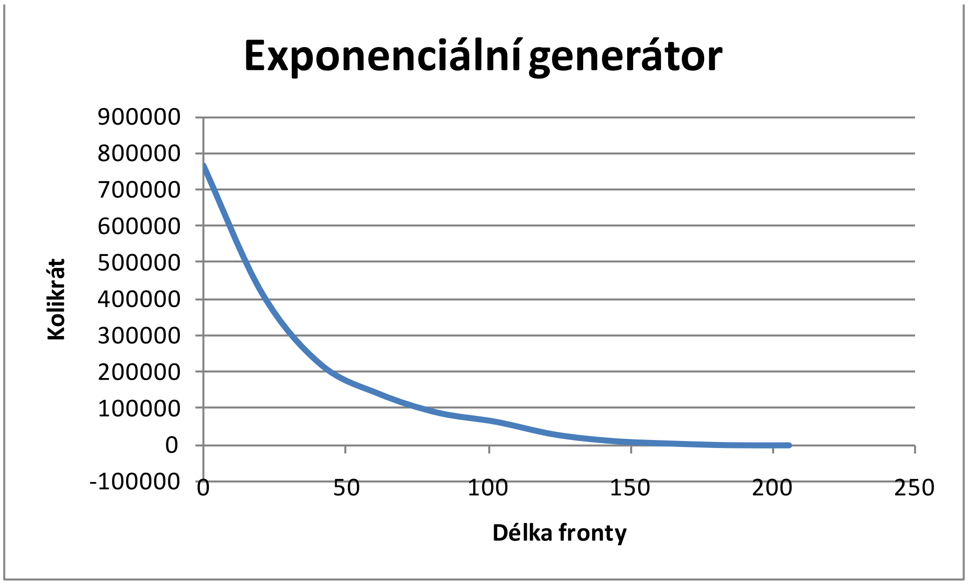
Podle zadání byla navrhnuta otevřená síť front. Byl proveden matematický výpočet zadaných veličin u sítě a i následně vytvořena simulace této sítě použitím J-Sim simulátoru a programovacího jazyka Java. Ze spuštěné simulace byly postupně získány výsledky jak pro doby generované exponenciálním generátorem čísel, tak i pro normálový generátor čísel. U normálového generátoru byly zvoleny koeficienty 0,05; 0,3 a 0,7. Výsledky simulace sítě pro exponenciální generátor byli skoro shodné matematickým výpočtům získaným z prvního kroku.

Přílohy

I. Výstup programu pro generátor exponenciálního rozdělení

```
Spoustim simulaci pro exponencialni rozdeleni:
Simulace prerusena v case 100054.55150407141
Poliklinika - generovano pacientu: 500790
Nemocnice - generovano pacientu: 499210
Statistika mistosti Obvodni lekar:
  Lq = 5.011020524602572
  Tq = 0.9875166551480615
  load = 0.8336388977700061
  Lw = 4.9426883464814475
Statistika mistosti Chirurgie:
  Lq = 34.792063784456154
  Tq = 2.046134289810135
  load = 0.9720608454985409
  Lw = 35.80840016176067
Statistika mistosti Sadrovna:
  Lq = 6.9927545649392275
  Tq = 1.959470329141473
  load = 0.8748866874523372
  Lw = 6.844791216120287
Statistika mistosti Rentgen:
  Lq = 3.518562299919277
  Tq = 0.5012427513133969
  load = 0.7786906689285075
  Lw = 3.514406000109125
Celkova statistika:
 Lq = 50.314401173917226
 Tq = 5.113816715406655
Celkovy pocet pacientu: 1000000
Statistika fronty Chirurgie:
  Stredni delka: 35.82873273992208
  Rozptyl: 1171.1633818852563
  Smerodatna odchylka: 34.222264417850205
  Minimalni delka: 0
  maximalni delka: 206
Histogram:
0000,0000. - 768367 - ******************************
0020,6000. - 418158 - ****************
```

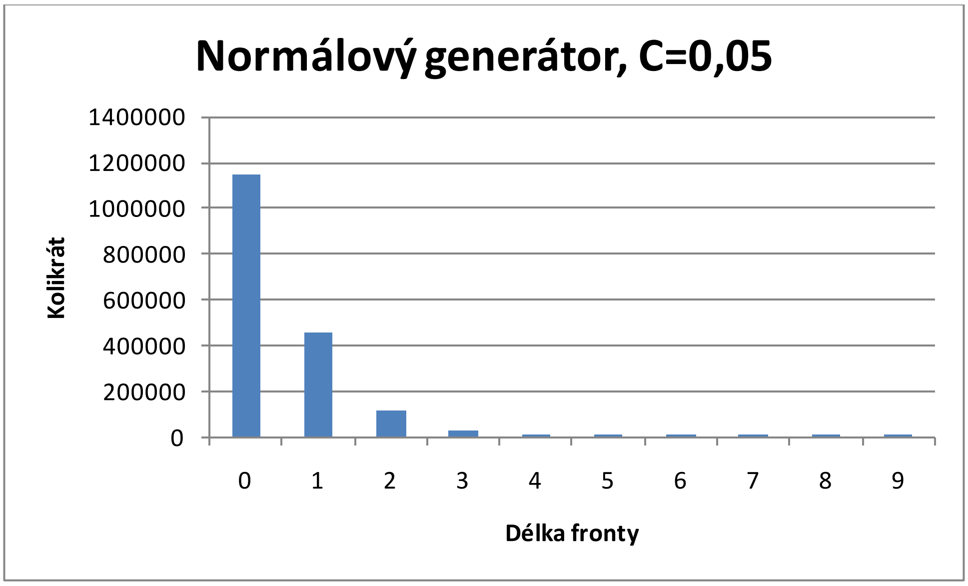
```
0041,2000. - 222599 - ********
0061,8000. - 141256 - *****
0082,4000. - 89774 - ***
0103,0000. - 64415 - **
0123,6000. - 29609 - *
0144,2000. - 11707 -
0164,8000. - 04757 -
0185,4000. - 00363 -
0206,0000. - 00001 -
```



II. Výstup programu pro generátor normálového rozdělení, koeficient 0,05

```
Spoustim simulaci pro normalove (Gauss) rozdeleni:
Koeficient C=0.05
Simulace prerusena v case 2500053.016987219
Poliklinika - generovano pacientu: 500028
Nemocnice - generovano pacientu: 499972
Statistika mistosti Obvodni lekar:
  Lq = 0.05840140963123175
  Tq = 0.27588482612250187
  load = 0.05517888496726395
  Lw = 0.05517888496725732
Statistika mistosti Chirurgie:
  Lq = 1.0280494373518256
  Tq = 1.0166804754437877
  load = 0.5069153731746431
  Lw = 0.7134469004756393
Statistika mistosti Sadrovna:
```

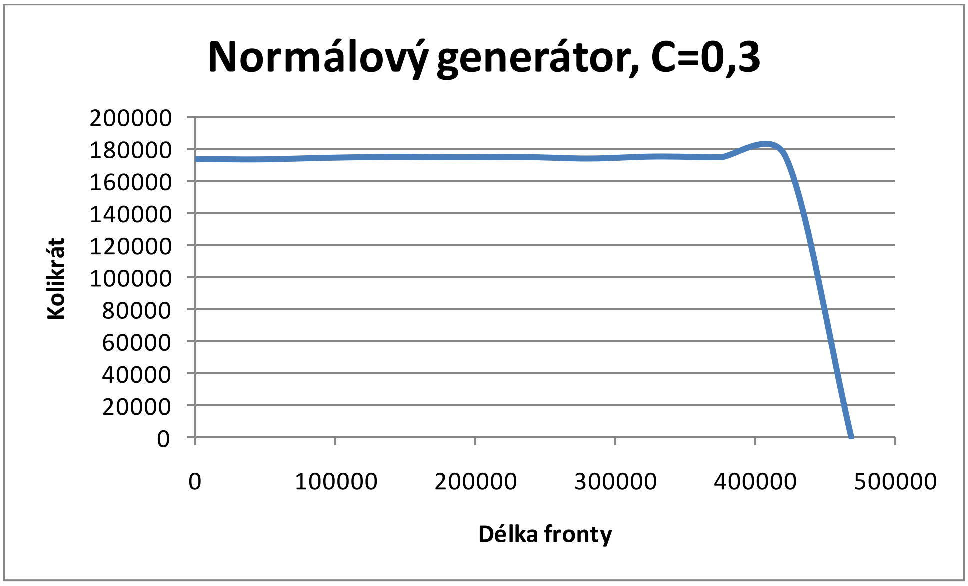
```
  Lq = 0.039353153887304024
  Tq = 0.27347400874050976
  load = 0.03786312067281324
  Lw = 0.038373934235143944
Statistika mistosti Rentgen:
  Lq = 0.11665596395691048
  Tq = 0.3891096488679926
  load = 0.10446902870919696
  Lw = 0.10942994312611852
Celkova statistika:
 Lq = 1.2424599648272718
 Tq = 2.2911227433501176
Celkovy pocet pacientu: 1000000
Statistika fronty Chirurgie:
  Stredni delka: 0.46276514186404283
  Rozptyl: 0.548368865041885
  Smerodatna odchylka: 0.7405193211806732
  Minimalni delka: 0
  maximalni delka: 9
Histogram:
0000,0000. - 1143711 - ******************************
0000,9000. - 457057 - ***********
0001,8000. - 117059 - ***
0002,7000. - 28064 -
0003,6000. - 06551 -
0004,5000. - 01498 -
0005,4000. - 00364 -
0006,3000. - 00071 -
0007,2000. - 00015 -
0008,1000. - 00000 -
0009,0000. - 00001 -
```



III. Výstup programu pro generátor normálového rozdělení, koeficient 0,3

```
Spoustim simulaci pro normalove (Gauss) rozdeleni:
Koeficient C=0.3
Simulace prerusena v case 7559039.274995917
Poliklinika - generovano pacientu: 500372
Nemocnice - generovano pacientu: 499628
Statistika mistosti Obvodni lekar:
  Lq = 0.10611354575879675
  Tq = 1.4779938453444426
  load = 0.09593368254612829
  Lw = 0.09783607538976424
Statistika mistosti Chirurgie:
  Lq = 1146961.069391132
  Tq = 1016198.1028656088
  load = 0.9999991281315863
  Lw = 234753.98558102373
Statistika mistosti Sadrovna:
  Lq = 0.047577401084933475
  Tq = 1.008564658445814
  load = 0.04541659741386125
  Lw = 0.046578393407581145
Statistika mistosti Rentgen:
  Lq = 0.24953353632801795
  Tq = 2.469469628377265
  load = 0.19970135180310383
  Lw = 0.22773625262619385
```

```
Celkova statistika:
 Lq = 1146961.4726156155
 Tq = 1774517.41007083
Celkovy pocet pacientu: 1000000
Statistika fronty Chirurgie:
  Stredni delka: 234815.09387142237
  Rozptyl: 1.832092018311509E10
  Smerodatna odchylka: 135354.79372048515
  Minimalni delka: 0
  maximalni delka: 469066
Histogram:
0000,0000. - 173766 - *****************************
46906,6000. - 173559 - *****************************
93813,2000. - 174517 - *****************************
140719,8000. - 175146 - *****************************
187626,4000. - 174863 - *****************************
234533,0000. - 175033 - *****************************
281439,6000. - 174085 - *****************************
328346,2000. - 175312 - ******************************
375252,8000. - 174865 - *****************************
422159,4000. - 175081 - *****************************
469066,0000. - 00002 -
```

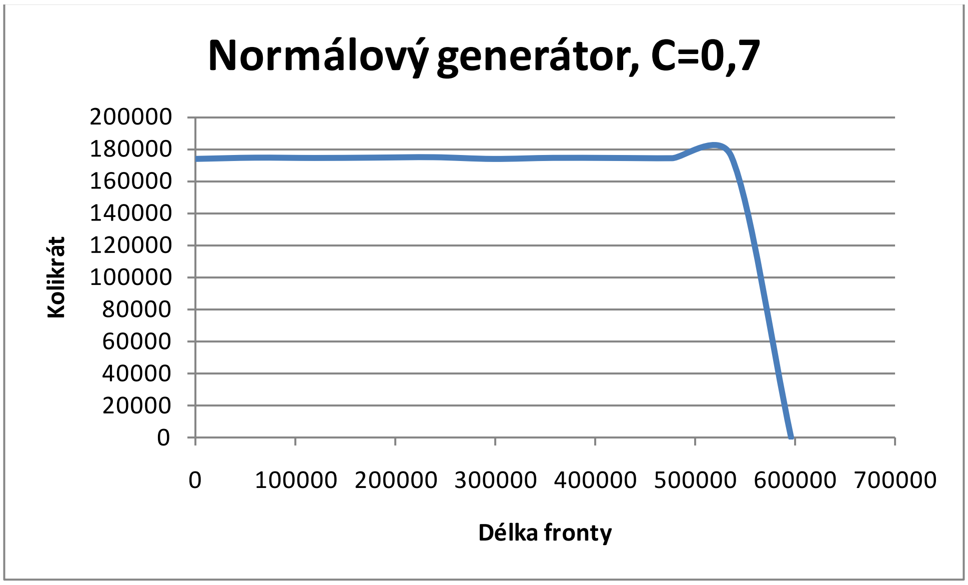


IV. Výstup programu pro generátor normálového rozdělení, koeficient 0,7

```
Spoustim simulaci pro normalove (Gauss) rozdeleni:
Koeficient C=0.7
Simulace prerusena v case 1.7660218918783214E7
```

```
Poliklinika - generovano pacientu: 499927
Nemocnice - generovano pacientu: 500073
Statistika mistosti Obvodni lekar:
  Lq = 0.10525518170244628
  Tq = 5.785823935323083
  load = 0.09523156592699222
  Lw = 0.16378560287481747
Statistika mistosti Chirurgie:
  Lq = 1027472.4619666453
  Tq = 3014926.6807334144
  load = 0.9999990267388531
  Lw = 298380.8290972803
Statistika mistosti Sadrovna:
  Lq = 0.04663837407231149
  Tq = 2.305990473614124
  load = 0.044560160632032476
  Lw = 0.045644726693735385
Statistika mistosti Rentgen:
  Lq = 0.24963999617281493
  Tq = 5.753546096256183
  load = 0.19976953117487428
  Lw = 0.22760140235728007
Celkova statistika:
 Lq = 1027472.8635001973
 Tq = 5269478.481101816
Celkovy pocet pacientu: 1000000
Statistika fronty Chirurgie:
  Stredni delka: 298297.76375820034
  Rozptyl: 2.963665178239792E10
  Smerodatna odchylka: 172152.9894669213
  Minimalni delka: 0
  maximalni delka: 596327
Histogram:
0000,0000. - 174165 - *****************************
59632,7000. - 174934 - *****************************
119265,4000. - 174745 - *****************************
178898,1000. - 174966 - *****************************
238530,8000. - 175210 - *****************************
298163,5000. - 174156 - *****************************
357796,2000. - 174815 - *****************************
417428,9000. - 174748 - *****************************
```

```
477061,6000. - 174592 - ******************************
536694,3000. - 175462 - *******************************
596327,0000. - 00001 -
```



Reference

[Rac02] Racek S.: Pravděpodobnostní modely počítačů, 2002 (nevydáno).