

Kolegij: Raspodijeljeni sustavi
Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu

Programski paket Pretty Damn Quick (PDQ)

Upute za postavljanje i korištenje

Zagreb, 2008

Sadržaj

0. Uvod.....	3
1. Postavljanje okruženja Cygwin	3
2. Postavljanje programskog paketa PDQ	5
3. Korišćenje programskog paketa PDQ	8
4. Izgradnja modela za programski paket PDQ	11
5. Naredbe programskog paketa PDQ	12
6. Literatura.....	15

0. Uvod

Programski paket PDQ omogućava izgradnju i analizu modela sustava zasnovanih na primjeni teorije repova. Programski paket PDQ moguće je koristiti u okruženjima raznovrsnih operacijskih sustava, kao što su UNIX, Linux i Windows te programskih jezika, kao što su Java, Perl, PHP i C. U ovom dokumentu bit će objašnjen postupak postavljanja i korištenja programskog paketa PDQ u okolini operacijskog sustava Windows XP primjenom programskog jezika C.

Uporaba programskog paketa PDQ u okolini operacijskog sustava Windows zahtjeva uporabu Cygwin (<http://www.cygwin.com>) okruženja. Okruženje Cygwin je zasebni programski alat koji ostvaruje emulaciju operacijskog sustava Linux u okolini operacijskog sustava Windows. Primjenom okruženje Cygwin moguće je postaviti i koristiti programski paket PDQ na isti način kao što bi se programski paket PDQ koristio u okruženju operacijskog sustava Linux.

1. Postavljanje okruženja Cygwin

Program za postavljanje okruženja Cygwin moguće je dohvatiti s adrese <http://www.cygwin.com/setup.exe>. Nakon dohvaćanja i pokretanja programa započinje postupak postavljanja okruženja na lokalno računalo. Primjenom sustava izbornika programa za postavljanje odabiru se sljedeće postavke:

Install from Internet – datoteke potrebne za postavljanje okruženja Cygwin na lokalno računalo bit će dohvaćene s javnog poslužitelja u globalnoj mreži Internet. Lokalno računalo na koje se postavlja paket mora imati otvorenu aktivnu vezu prema globalnoj mreži Internet.

Root directory – direktorij na lokalnom računalu u koji će biti ostvareno postavljanje okruženja Cygwin. Primjerice, može biti odabran direktorij: `c:\cygwin`. Na istom izborniku, potrebno je također odabrati opcije **All Users** i **Unix / binary**.

Local Package Directory – direktorij na lokalnom računalu u koji će biti snimljene datoteke korištene tijekom postavljanja okruženja Cygwin. Potrebno je upisati direktorij koji se zove **latest** a nalazi se kao poddirektorij osnovnog direktorija okruženja Cygwin. Primjerice, može biti odabran direktorij: `c:\cygwin\latest`.

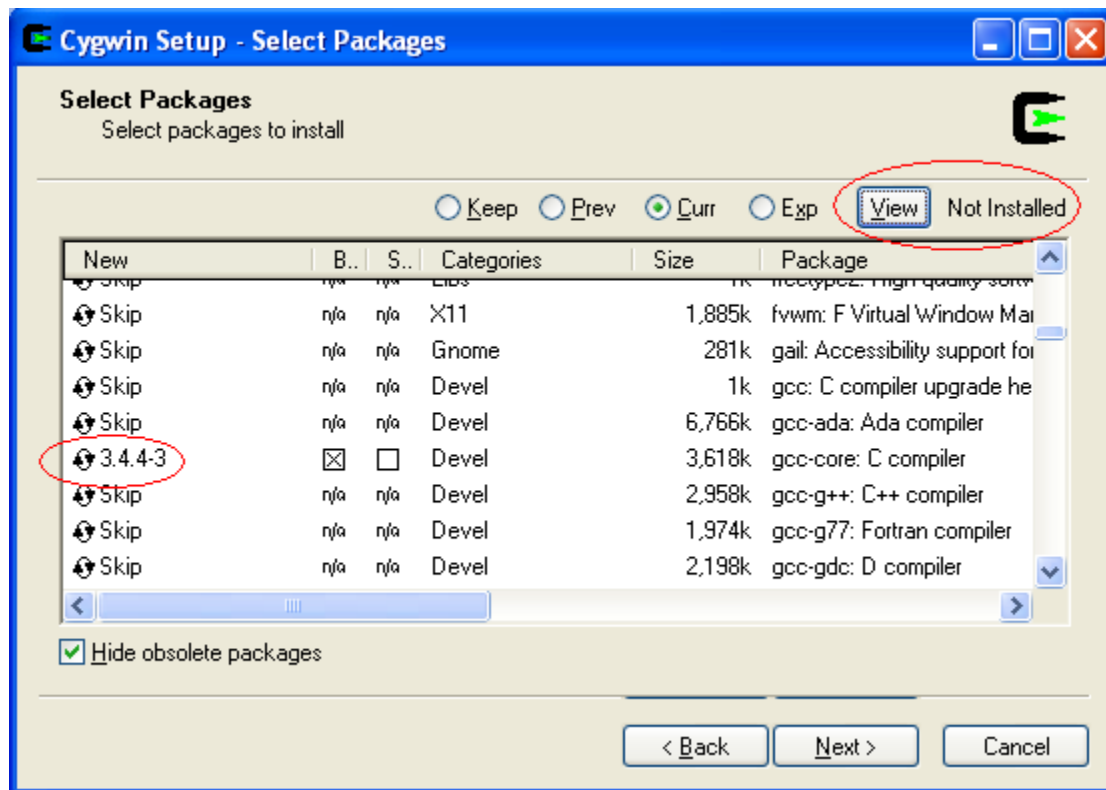
Available download sites – potrebno je odabrati jedan od javnih HTTP ili FTP poslužitelja za dohvat datoteka okruženja Cygwin. Preporučljivo je odabrati neki od geografski bližih poslužitelj. Primjerice, neki od poslužitelja u Njemačkoj (<http://cygwin.basemirror.de>).

Select packages – potrebno je odabrati pakete koji će biti postavljeni u sklopu okruženja Cygwin. U prvom koraku odabire se opcija pregleda svih paketa koji nisu postavljeni na lokalno računalo odabirom tipke View. Tipka View se koristi sve dok se ne ostvari pregled paketa koji nisu postavljeni na lokalno računalo. U listi paketa, potrebno je odabrati sljedeće skupine paketa:

gcc-core – paket koji uključuje jezični procesor za programski jezik C.

make – paket koji uključuje alat za automatizirano prevođenje izvornih datoteka.

Sljedeća slika prikazuje primjer označavanja paketa gcc-core



Osim navedenih paketa, na opisani način moguće je odabrati i dodatne pakete. Primjerice, dodatne programe za obradu tekstualnih datoteka.

2. Postavljanje programskog paketa PDQ

Programski paket Pretty Damn Quick – PDQ (<http://www.perfdynamics.com/Tools/PDQcode.html>) moguće je dohvatiti s adrese <http://www.perfdynamics.com/Tools/pdq.tar.gz>. Dohvaćena arhiva sadrži datoteke programskog paketa PDQ. Arhivu je potrebno postaviti u osnovni direktorij okruženja Cygwin. Primjerice, direktorij c:\cygwin.

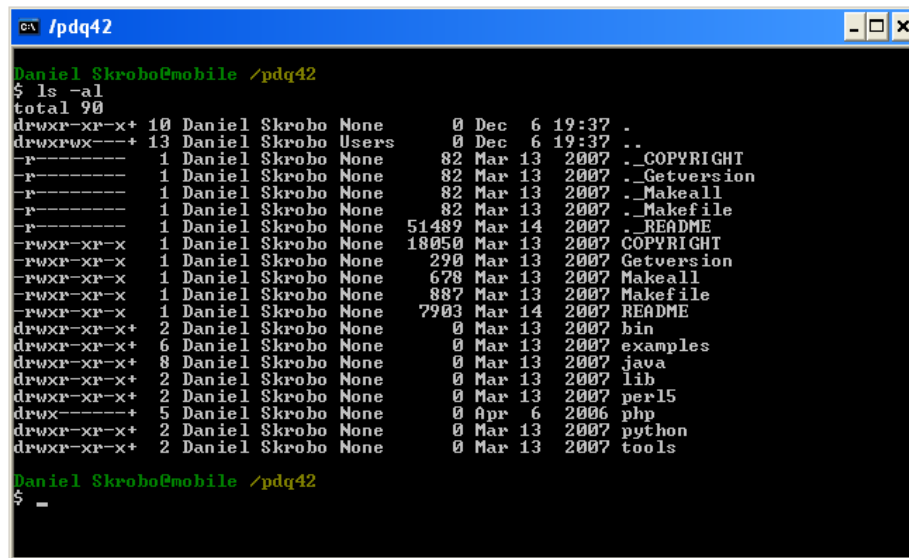
Nakon postavljanja arhive programskog paketa PDQ, potrebno je pokrenuti okruženje Cygwin. Dohvaćenu arhivu potrebno je raspakirati primjenom naredbi:

```
cd /
gunzip pdq.tar.gz
tar -xvf pdq.tar.gz
```

Nakon raspakiravanja arhive, potrebno je pokrenuti naredbe:

```
cd pdq42
ls -al
```

rezultat izvođenja navedenih naredbi bit će:



```

Daniel Skrobo@mobile /pdq42
$ ls -al
total 90
drwxr-xr-x+ 10 Daniel Skrobo None      0 Dec  6 19:37 .
drwxrwx---+ 13 Daniel Skrobo Users    0 Dec  6 19:37 ..
-r-----+  1 Daniel Skrobo None      82 Mar 13 2007  _COPYRIGHT
-r-----+  1 Daniel Skrobo None      82 Mar 13 2007  _Getversion
-r-----+  1 Daniel Skrobo None      82 Mar 13 2007  _Makeall
-r-----+  1 Daniel Skrobo None      82 Mar 13 2007  _Makefile
-r-----+  1 Daniel Skrobo None     51489 Mar 14 2007  _README
-rwxr-xr-x  1 Daniel Skrobo None    18050 Mar 13 2007  COPYRIGHT
-rwxr-xr-x  1 Daniel Skrobo None     290 Mar 13 2007  Getversion
-rwxr-xr-x  1 Daniel Skrobo None     678 Mar 13 2007  Makeall
-rwxr-xr-x  1 Daniel Skrobo None     887 Mar 13 2007  Makefile
-rwxr-xr-x  1 Daniel Skrobo None    7903 Mar 14 2007  README
drwxr-xr-x+  2 Daniel Skrobo None      0 Mar 13 2007  bin
drwxr-xr-x+  6 Daniel Skrobo None      0 Mar 13 2007  examples
drwxr-xr-x+  8 Daniel Skrobo None      0 Mar 13 2007  java
drwxr-xr-x+  2 Daniel Skrobo None      0 Mar 13 2007  lib
drwxr-xr-x+  2 Daniel Skrobo None      0 Mar 13 2007  perl5
drwx-----  5 Daniel Skrobo None      0 Apr  6 2006  php
drwxr-xr-x+  2 Daniel Skrobo None      0 Mar 13 2007  python
drwxr-xr-x+  2 Daniel Skrobo None      0 Mar 13 2007  tools
Daniel Skrobo@mobile /pdq42
$ -
```

Ostvareni prikaz sadrži popis svih datoteka i direktorija programskog paketa PDQ koj su dohvaćeni iz sažete arhive pdq.tar.gz.

Nakon pozicioniranja u direktorij okruženja PDQ, potrebno je pokrenuti naredbu za prevođenje izvornog koda koji ostvaruje logiku okruženja PDQ. Prevođenje se ostvaruje primjenom sljedećih naredbi:

```
./Makeall
```

Rezultat izvođenja navedene naredbe bit će sljedeći ispis:

```
cc -ggdb -c -o MVA_Approx.o MVA_Approx.c
cc -ggdb -c -o MVA_Canon.o MVA_Canon.c
cc -ggdb -c -o MVA_Solve.o MVA_Solve.c
cc -ggdb -c -o PDQ_Build.o PDQ_Build.c
cc -ggdb -c -o PDQ_Exact.o PDQ_Exact.c
cc -ggdb -c -o PDQ_Globals.o PDQ_Globals.c
cc -ggdb -c -o PDQ_Report.o PDQ_Report.c
cc -ggdb -c -o PDQ_Utils.o PDQ_Utils.c
ar -rv libpdq.a MVA_Approx.o MVA_Canon.o MVA_Solve.o PDQ_Build.o
PDQ_Exact.o PDQ_Globals.o PDQ_Report.o PDQ_Utils.o
ar: creating libpdq.a
a - MVA_Approx.o
a - MVA_Canon.o
a - MVA_Solve.o
a - PDQ_Build.o
a - PDQ_Exact.o
a - PDQ_Globals.o
a - PDQ_Report.o
a - PDQ_Utils.o
# -ranlib -sc libpdq.a # MacOS X
ranlib libpdq.a # Cygwin
./setup.sh: line 5: $'\r': command not found
'. Stop. No rule to make target `clean
./setup.sh: line 7: $'\r': command not found
: No such file or directory
./setup.sh: line 9: $'\r': command not found
./setup.sh: line 11: $'\r': command not found
": No such file or directoryfile.PL
./setup.sh: line 13: $'\r': command not found
'. Stop. No rule to make target `install
./setup.sh: line 15: $'\r': command not found
python setup.py build
make: python: Command not found
make: *** [build] Error 127

Making chapter 2 PDQ files ...
gcc -o erlang erlang.c -lm
gcc -o closed1 closed1.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o open1 open1.c -I../.../lib -L../.../lib -lpdq -lm

Making chapter 3 PDQ files ...
gcc -o closed closed.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o feedback feedback.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o fesc fesc.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o mc-test mc-test.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o multi_class multi_class.c -I../.../lib -L../.../lib -lpdq -
lm
gcc -o mva mva.c -lm
gcc -o passport passport.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o repair repair.c -lm
gcc -o shadowcpu shadowcpu.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o simple_series simple_series.c -I../.../lib -L../.../lib -
lpdq -lm
```

```
Making chapter 6 PDQ files ...
gcc -o dbc dbc.c -I../.../lib -L../.../lib -lpdq -lm

Making chapter 7 PDQ files ...
gcc -O -o abcache abcache.c -I ../.../lib -L../.../lib -lpdq -lm
gcc -O -o multibus multibus.c -I ../.../lib -L../.../lib -lpdq -lm

Making chapter 8 PDQ files ...
gcc -o baseline baseline.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o scaleup scaleup.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o upgradel upgradel.c -I../.../lib -L../.../lib -lpdq -lm
gcc -o upgrade2 upgrade2.c -I../.../lib -L../.../lib -lpdq -lm

Making chapter 9 PDQ files ...
gcc -o http http.c -I ../.../lib -L../.../lib -lpdq -lm
gcc -o iis iis.c -I ../.../lib -L../.../lib -lpdq -lm
gcc -o ebiz ebiz.c -I ../.../lib -L../.../lib -lpdq -lm
```

Unatoč prijavljenim greškama, programski paket PDQ sada je spreman za korištenje. Postupak korištenja programskog paketa PDQ bit će opisan u sljedećem poglavlju.

3. Korištenje programskog paketa PDQ

Nakon izvođenja postupka postavljanja programskog paketa PDQ, postavljeni paket moguće je koristiti za izvođenje modela. Modeli koji su prikazani na predavanju nalaze se u direktoriju primjeri koji je moguće dohvatiti iz repozitorija na stranici predmeta s adrese www.fer.hr/predmet/rassus.

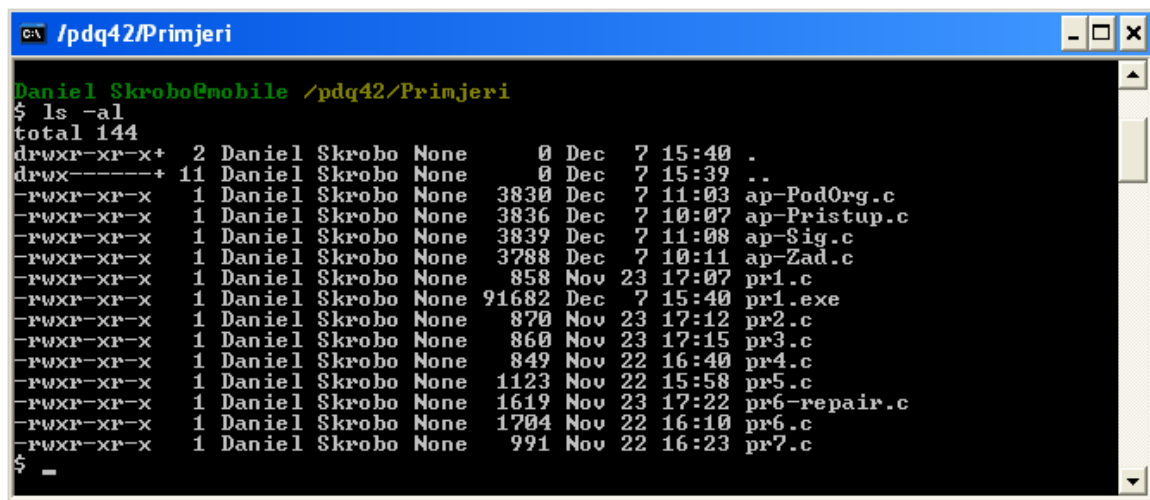
Arhivu s primjerima potrebno je raspakirati u osnovni direktorij programskog paketa PDQ. Osnovni direktorij programskog paketa PDQ je `c:\cygwin\pdq42`. Nakon raspakiravanja arhive s primjerima, potrebno je za radni direktorij postaviti direktorij s primjerima izvođenjem naredbe:

```
cd Primjeri
```

Izvođenjem naredbe:

```
ls -al
```

dobija se ispis svih datoteka u direktoriju s primjerima:



```
Daniel Skrobo@mobile /pdq42/Primjeri
$ ls -al
total 144
drwxr-xr-x+ 2 Daniel Skrobo None    0 Dec  7 15:40 .
drwx-----+ 11 Daniel Skrobo None    0 Dec  7 15:39 ..
-rwxr-xr-x  1 Daniel Skrobo None 3830 Dec  7 11:03 ap-PodOrg.c
-rwxr-xr-x  1 Daniel Skrobo None 3836 Dec  7 10:07 ap-Pristup.c
-rwxr-xr-x  1 Daniel Skrobo None 3839 Dec  7 11:08 ap-Sig.c
-rwxr-xr-x  1 Daniel Skrobo None 3788 Dec  7 10:11 ap-Zad.c
-rwxr-xr-x  1 Daniel Skrobo None   858 Nov 23 17:07 pr1.c
-rwxr-xr-x  1 Daniel Skrobo None 91682 Dec  7 15:40 pr1.exe
-rwxr-xr-x  1 Daniel Skrobo None   870 Nov 23 17:12 pr2.c
-rwxr-xr-x  1 Daniel Skrobo None   860 Nov 23 17:15 pr3.c
-rwxr-xr-x  1 Daniel Skrobo None   849 Nov 22 16:40 pr4.c
-rwxr-xr-x  1 Daniel Skrobo None  1123 Nov 22 15:58 pr5.c
-rwxr-xr-x  1 Daniel Skrobo None  1619 Nov 23 17:22 pr6-repair.c
-rwxr-xr-x  1 Daniel Skrobo None  1704 Nov 22 16:10 pr6.c
-rwxr-xr-x  1 Daniel Skrobo None   991 Nov 22 16:23 pr7.c
$ -
```

Izvođenje modela bit će opisano primjenom primjera `pr1.c`. Datoteka `pr1.c` sadrži programski kod koji ostvaruje logiku modela sustava opisanog na predavanju u sklopu prvog primjera.

Izgrađeni programski kod prevodi se u izvršni kod primjenom jezičnog procesora `gcc`. Prevođenje se ostvaruje primjenom sljedeće naredbe:

```
gcc pr1.c -I../lib -L../lib -lpdq -o pr1
```

Parametar `pr1.c` određuje ime datoteke s programskim kodom modela koji se prevodi. Parametar `"-I../lib"` određuje direktorij gdje se nalaze izvorne datoteke knjižnice

programskog paketa PDQ. Parametar "-I./lib" određuje direktorij gdje se nalaze izvršne datoteke knjižnice programskog paketa PDQ. Parametar "-lpdq" nalaže alatu GCC da u ciljani program ugradi izvršne datoteke programskog paketa PDQ. Parametar "pr1" određuje ime odredišne datoteke u koju je spremljen izvršni kod modela. Nakon izvođenja opisane naredbe, bit će izgrađena datoteka pr1.exe koja sadrži izvršni kod modela. Izgrađeni izvršni kod moguće je pokrenuti izvođenjem naredbe:

```
./pr1.exe
```

Izvođenjem navedene naredbe dati će sljedeći ispis:

```
*****
***** Pretty Damn Quick REPORT *****
*****
*** of : Sat Dec 6 20:13:55 2008 ***
*** for: Diskovni podsustav ***
*** Ver: PDQ Analyzer v4.2 20070228
***

*****
*****
*****
***** PDQ Model INPUTS *****
*****

Node Sched Resource Workload Class Demand
---- -
CEN FCFS Poslužitelj Operacije TRANS 0.0100

Queueing Circuit Totals:

Streams: 1
Nodes: 1

WORKLOAD Parameters

Source per Sec Demand
-----
Operacije 50.0000 0.0100

*****
***** PDQ Model OUTPUTS *****
*****

Solution Method: CANON

***** SYSTEM Performance *****

Metric Value Unit
-----
Workload: "Operacije"
Mean Throughput 50.0000 Job/Sec
Response Time 0.0200 Sec

Bounds Analysis:
```

Max Demand	100.0000	Job/Sec		
Max Throughput	100.0000	Job/Sec		
***** RESOURCE Performance *****				
Metric	Resource	Work	Value	Unit
-----	-----	----	-----	----
Throughput	Posluzitelj	Operacije	50.0000	Job/Sec
Utilization	Posluzitelj	Operacije	50.0000	Percent
Queue Length	Posluzitelj	Operacije	1.0000	Job
Residence Time	Posluzitelj	Operacije	0.0200	Sec

Prikazani ispis sadrži rezultat izvođenja modela. Rezultat uključuje informacije o propusnosti, zaposlenosti, duljine repova i vremenu zadržavanja na pojedinim elementima ostvarenog modela.

4. Izgradnja modela za programski paket PDQ

Modeli sustava opisuju se izgradnjom programa u programskom jeziku C. Postupak izgradnje programa bit će opisan na prvom primjeru koji se predstavljen u sklopu predavanja.

Ostvarenje prvog primjera i opis elemenata ostvarenja ostvarene je na sljedećem ispisu:

01: #include <stdio.h> 02: #include <math.h> 03: #include <PDQ_Lib.h>	Direktive za uključivanje standardnih knjižnica i PDQ_Lib.h knjižnice koja sadrži ostvarenja naredbi programskog paketa PDQ
04: main() {	Početak ostvarenja logike modela
05: extern int nodes, streams;	Globalne varijable koje koristi programski paket PDQ. Nužno ih je navesti za pravilan rad modela.
06: float L = 50; 07: float S = 0.01;	Varijable L definira učestalost dolazaka zahtjeva u sustav, dok varijable S definira vrijeme provedeno na posluživanju zahtjeva
08: PDQ_Init("Diskovni podsustav");	Naredba za postavljanje programskog paketa PDQ. Naredba kao parametar prima ime izgrađenog modela.
09: nodes = PDQ_CreateNode("Poslužitelj", CEN, FCFS);	Naredba za stvaranje repa s poslužiteljem. Naredba kao parametar prima ime poslužitelja te njegove značajke. Parametar CEN definira standardni čvor s repom, dok parametar FCFS definira model posluživanja zahtjeva u repu prema redoslijedu prispjeka.
10: streams = PDQ_CreateOpen("Operacije", L);	Naredba za stvaranje toka zahtjeva. Prvi parametar definira ime toka, dok drugi parametar definira učestalost dolazaka zahtjeva u toku.
11: PDQ_SetDemand("Poslužitelj", "Operacije", S);	Naredba za usmjeravanje toka na poslužitelj. Prvi parametar definira ime poslužitelja na koji je usmjeren tok, drugi parametar definira ime toka koji se usmjerava na poslužitelj, dok treći parametar definira srednje vrijeme posluživanja zahtjeva na poslužitelju.
12: PDQ_Solve(CANON);	Naredba za provođenje postupka analize ostvarenog modela
13: PDQ_Report();	Ispis standardnog obrasca sa statističkim informacijama izvedenog modela
14: }	Završetak ostvarenja logike modela

5. Naredbe programskog paketa PDQ

Programski paket sadrži jednostavni skup naredbi koje je se koriste tijekom izgradnje modela. U ovom poglavlju bit će opisane osnovne naredbe koje su dovoljne za rješavanje domaće zadaće.

Naredba paketa PDQ	Opis naredbe
<code>PDQ_Init(ime)</code>	Postavlja paket PDQ u početno stanje. Potrebno ju je pozvati prije bilo koje druge naredbe paketa PDQ. Parametar <i>ime</i> određuje ime ostvarenog modela. Primjer: PDQ_Init("Aplikacija za kupovinu")
<code>PDQ_CreateOpen(ime, l)</code>	Stvaranje toka zahtjeva. Parametar <i>ime</i> definira ime toka, dok parametar <i>l</i> definira učestalost dolazaka zahtjeva u toku. Primjer: PDQ_CreateOpen("zahtjevi", 2.3)
<code>PDQ_CreateNode(ime, tip, pos)</code>	Stvaranje repa s poslužiteljem. Parametar <i>ime</i> određuje ime poslužitelja. Parametar <i>tip</i> definira tip poslužitelja. Tip CEN definira standardni čvor s repom, dok tip DLY definira poslužitelj bez repa čekanja. Parametar <i>pos</i> definira model posluživanja zahtjeva u repu poslužitelja. Model FCFS definira model posluživanja zahtjeva u repu prema redoslijedu prispjeća. Model LCFS definira model posluživanja zahtjeva u repu u kojem se prvo poslužuje posljednji zahtjev koji je stavljen u rep. Primjer: PDQ_CreateNode("posluzitelj", CEN, FCFS)
<code>PDQ_SetDemand(ime1, ime2, t)</code>	Definiranje vremena posluživanja za zahtjeve na poslužitelju. Parametar <i>ime1</i> definira ime poslužitelja, parametar <i>ime2</i> definira ime toka zahtjeva. Parametar <i>t</i> definira vrijeme posluživanja zahtjeva na poslužitelju. Primjer: PDQ_SetDemand("posluzitelj", "zahtjevi", 3.0)
<code>PDQ_SetVisits(ime1, ime2, v, t)</code>	Definiranje učestalosti obilaska za zahtjeva na poslužitelju. Parametar <i>ime1</i> definira ime poslužitelja, parametar <i>ime2</i> definira ime toka zahtjeva. Parametar <i>v</i> definira učestalost obilazaka zahtjeva na poslužitelju. Parametar <i>t</i> definira vrijeme posluživanja zahtjeva na poslužitelju. Primjer: PDQ_SetVisits("posluzitelj", "zahtjevi", 0.1, 3.0)

<code>PDQ_SetTUnit(ime)</code>	<p>Definiranje imena vremenske jedinice. Parametar <i>ime</i> definira ime vremenske jedinice.</p> <p>Primjer: <code>PDQ_SetWUnit("dan")</code></p>
<code>PDQ_SetWUnit(ime)</code>	<p>Definiranje imena zahtjeva. Parametar <i>ime</i> definira ime zahtjeva.</p> <p>Primjer: <code>PDQ_SetWUnit("korisnici")</code></p>
<code>PDQ_Solve(tip)</code>	<p>Provođenje postupka analize ostvarenog modela. Parametar <i>tip</i> definira vrstu postupka analize. Vrsta APPROX nalaže uporabu aproksimativne metode analize, vrsta CANON nalaže uporabu kanonske metode analize, dok vrsta EXACT nalaže uporabu točnog iterativnog postupka analize,</p> <p>Primjer: <code>PDQ_Solve(CANON)</code></p>
<code>PDQ_Report()</code>	<p>Ispis standardnog obrasca sa statističkim informacijama izvedenog modela</p> <p>Primjer: <code>PDQ_Report()</code></p>
<code>PDQ_GetQueueLength(ime1, ime2, r)</code>	<p>Dohvat duljine repa za zadani poslužitelj. Parametar <i>ime1</i> određuje ime poslužitelja, parametar <i>ime2</i> određuje ime toka zahtjeva, dok parametar <i>r</i> određuje vrstu toka poslova. Vrsta BATCH određuje tok u kojem ne postoji vrijeme razmišljanja i primjenjuje se samo u zatvorenim modelima. Vrsta TERM određuje tok u kojem postoji vrijeme razmišljanja i primjenjuje se samo u zatvorenim modelima. Vrsta TRANS primjenjuje se samo za tokove u otvorenim modelima.</p> <p>Primjer: <code>PDQ_GetQueueLength("posluzitelj", "zahtjevi", TRANS)</code></p>
<code>PDQ_GetResidenceTime(ime1, ime2, r)</code>	<p>Dohvat vremena zadržavanja za zadani poslužitelj. Parametar <i>ime1</i> određuje ime poslužitelja, parametar <i>ime2</i> određuje ime toka zahtjeva, dok parametar <i>r</i> određuje vrstu toka poslova. Vrsta BATCH određuje tok u kojem ne postoji vrijeme razmišljanja i primjenjuje se samo u zatvorenim modelima. Vrsta TERM određuje tok u kojem postoji vrijeme razmišljanja i primjenjuje se samo u zatvorenim modelima. Vrsta TRANS primjenjuje se samo za tokove u otvorenim modelima.</p> <p>Primjer:</p>

	PDQ_GetResidenceTime("poslužitelj","zahtjevi", TRANS)
PDQ_GetResponse(<i>r</i> , <i>ime</i>)	<p>Dohvat vremena odziva za tok podataka. Parametar <i>ime</i> određuje ime toka zahtjeva, dok parametar <i>r</i> određuje vrstu toka poslova. Vrsta BATCH određuje tok u kojem ne postoji vrijeme razmišljanja i primjenjuje se samo u zatvorenim modelima. Vrsta TERM određuje tok u kojem postoji vrijeme razmišljanja i primjenjuje se samo u zatvorenim modelima. Vrsta TRANS primjenjuje se samo za tokove u otvorenim modelima.</p> <p>Primjer: PDQ_GetResponse(TRANS,"zahtjevi")</p>
PDQ_GetThruput(<i>r</i> , <i>ime</i>)	<p>Dohvat propusnost sustava za tok podataka. Parametar <i>ime</i> određuje ime toka zahtjeva, dok parametar <i>r</i> određuje vrstu toka poslova. Vrsta BATCH određuje tok u kojem ne postoji vrijeme razmišljanja i primjenjuje se samo u zatvorenim modelima. Vrsta TERM određuje tok u kojem postoji vrijeme razmišljanja i primjenjuje se samo u zatvorenim modelima. Vrsta TRANS primjenjuje se samo za tokove u otvorenim modelima.</p> <p>Primjer: PDQ_GetThruput(TRANS,"zahtjevi")</p>
PDQ_GetUtilization(<i>ime1</i> , <i>ime2</i>)	<p>Dohvat zaposlenosti za poslužitelj i tok podataka. Parametar <i>ime1</i> određuje ime poslužitelja, parametar <i>ime2</i> određuje ime toka zahtjeva.</p> <p>Primjer: PDQ_GetUtilization("poslužitelj","zahtjevi")</p>

6. Literatura

- [1] N.J. Gunther: "Analyzing Computer System Performance with Perl::PDQ", Springer, 2005. ISBN 3-540-20865-8.
- [2] Pretty Damn Quick - PDQ, <http://www.perfdynamics.com/Tools/PDQcode.html>, prosinac 2008.
- [3] Cygwin, <http://www.cygwin.com>, prosinac 2008.