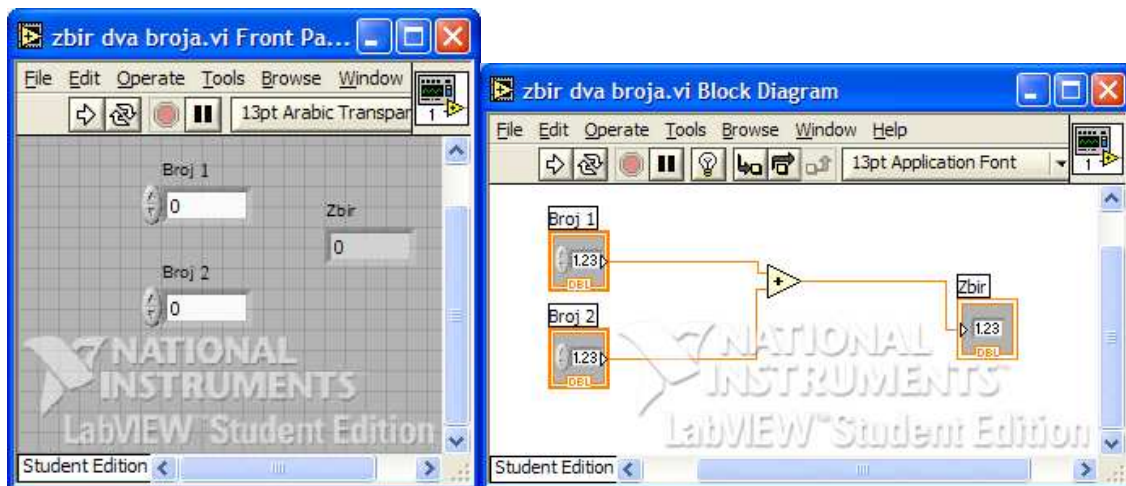


LabVIEW-ZADACI

1. Napisati program u LabVIEW-u koji računa zbir dva broja.

Rješenje:

- Startovati LabVIEW
- Birati **New VI**
- U okviru **Controls Palette** birati numerički kontroler tipa **Numerical Control**, i postaviti ga na Front panel. Kontroler nazvati *broj 1*
- Ponoviti prethodnu tačku i drugi kontroler nazvati *broj 2*
- U okviru **Controls Palette** birati numerički indikator tipa **Numerical Indicator**, i postaviti ga na Front panel. Indikator nazvati *zbir*
- Preći na Block Diagram
- U okviru **Functions Palette** birati funkciju **Numeric**, tipa **Add** i postaviti je u blok dijagram
- U okviru **Tools Palette** birati **Wiring Tool** i izvršiti odgovarajuće povezivanje terminala i ikone funkcije u blok dijagramu
- Vratiti se na Front Panel, upisati željene vrijednosti sabiraka i pokrenuti program.



Front panel i Blok dijagram iz Zadatka 1

2. Napisati program u LabVIEW-u koji proširuje prethodni primjer, pa na osnovu izbora korisnika, računa zbir, razliku, proizvod ili deljenje dva broja.

Rješenje:

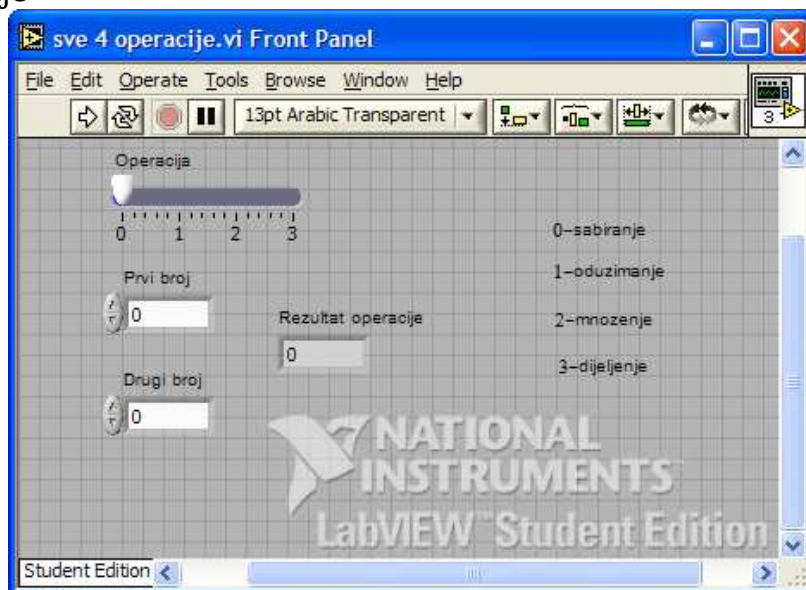
- U front panel prethodnog primjera dodati numerički kontoler tipa **Vertical Pointer Slide**, i nazvati ga *operacija*
- Promijeniti opseg skale kontrolera tako da se opseg kreće od 0 do 3.
- Promijeniti reprezentaciju brojeva na skali kontrolera *operacija* tako što treba birati opciju Byte (I8). Reprezentacija objekata na front panelu nalazi se u okviru menija klikom na desni taster nakon pozicioniranja na željeni objekat.
- Pored brojeva na skali (0, 1, 2 i 3) unijeti oznake koji položaj predstavlja koju operaciju. Birati:

0: zbir

1: razlika

2: proizvod

3: dijeljenje



- Promijeniti naziv postojećeg indikatora *zbir* u *rezultat*
- Preći na Block Diagram
- Izbrisati žice postavljene u primeru 1
- U okviru **Functions Pallette** birati strukturu **Case** i postaviti je u blok dijagram
- Povezati kontroler *operacija* na ulaz **Case** strukture tj. na uslovni terminal iste. Obratiti pažnju na promenu boje uslovnog terminala usled prilagođavanja vrsti podataka vezanim za njega.

- U okviru iste strukture promeniti i nazive okvira, tj. pored dva postojeća (0 i 1) dodati još dva slučaja (2 i 3). To se postiže:
 - strelicom doći do oznake slučaja 1
 - desnim klikom na strelicu otvara se meni u kojem treba birati opciju **Add Case After** nakon čega se pojavljuje i slučaj 2 u datoj **case** strukturi. Ponoviti još jednom ceo postupak kako bi se pojavio i slučaj 3.

- Na taj način dobijena je case struktura sa 4 uslova:

0 - zbir

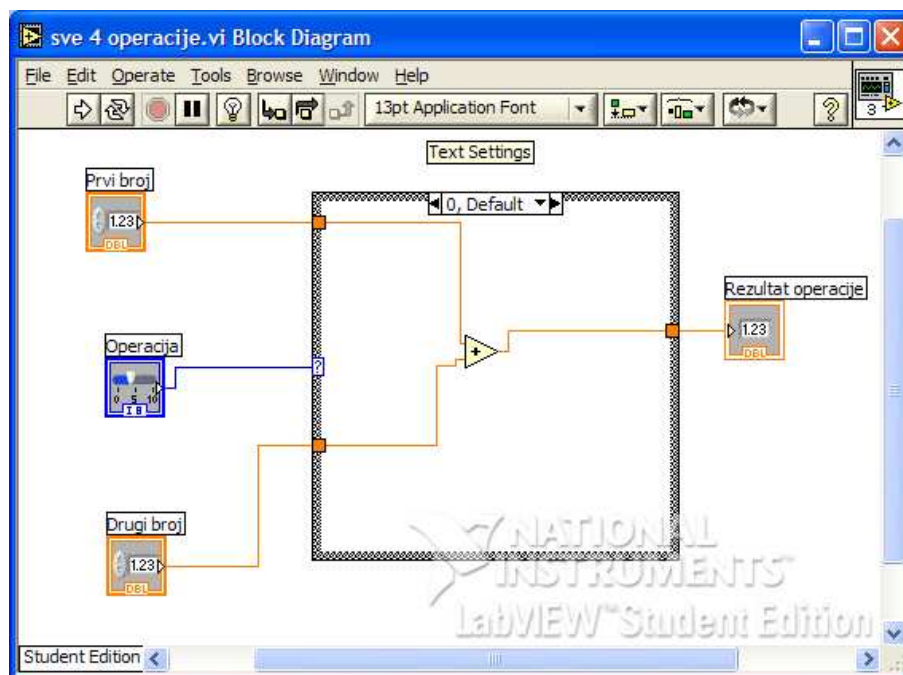
1 - razlika

2 - proizvod

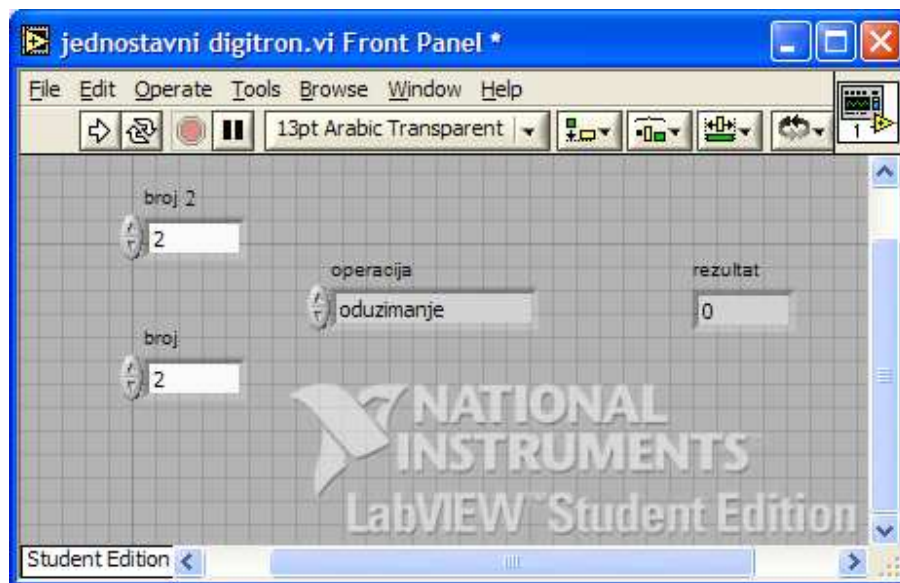
3 – deljenje

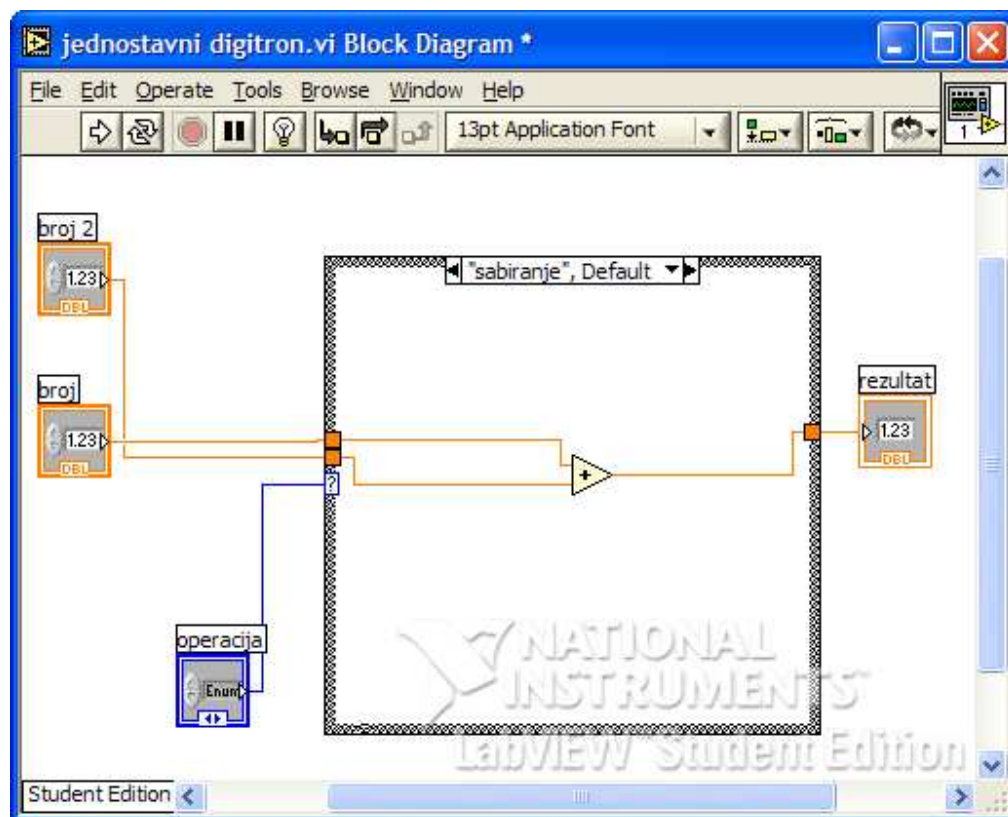
Postaviti da slučaj 0 bude Default.

- Strelicama se zaustaviti na prozoru uslova "0". Obzirom da ovaj uslov znači "zbir", u okvir "0" treba postaviti numeričku funkciju **Add** koja se može iskoristiti iz prethodnog primera
- Strelicom preći na prozor uslova "1". Obzirom da ovaj uslov znači "razlika", u okvir "1" treba postaviti numeričku funkciju **Subtract**
- Strelicom preći na prozor uslova "2". Obzirom da ovaj uslov znači "proizvod", u okvir "2" treba postaviti numeričku funkciju **Multiply**
- Strelicom preći na prozor uslova "3". Obzirom da ovaj uslov znači "deljenje", u okvir "3" treba postaviti numeričku funkciju **Divide**
- Povezati na odgovarajući način sve terminale i ikone funkcija u blok dijagramu
- Preći na front panel, birati operaciju i vrednosti ulaznih promenljivih, pokrenuti program i posmatrati rezultat.



Drugi način:





3. Projektovati virtuelni instrument koji generiše slučajne brojeve u intervalu $[0,1]$ i prikazuje ih na grafiku. Nakon toga podesiti da se brojevi generišu svakih 250ms, a zatim napraviti da se omogući on-line vremenska kontrola generisanja brojeva.

Rješenje:

1. Kreirati Virtualni Instrument pritiskom na New -> BlankVI u *LabVIEW dialog box-u*.
2. Postaviti aktivanu *Control Paletu*.
3. Izabrati *Stop Button* iz **Controls » Boolean** palete.
4. Postaviti ga na *front panel*.
5. Kreirati graf (waveform chart) izborom **Waveform Chart** iz **Controls » Graph** podpalete
6. Postaviti kontrolu na *front panel* i imenovati je sa **Random Plot**.
7. Za promenu skale grafa selektovati *Operating tool* iz *Tools Palette*. Dvaput kliknuti mišem na y-osu i promijeniti skalu sa $[0,1]$.

- Blok dijagram - BLOCK DIAGRAM

1. Izborom **Show Diagram** iz **Windows** menija otvoriti *Diagram* prozor. Dva terminala koja se nalaze na blok dijagramu odgovaraju **Stop Button-u** i grafu **Random Plot** sa front panela.

2. Izabrati funkciju **Random Number (0–1)**  iz **Functions » Numeric** palete. Ova funkcija generiše slučajne brojeve u intervalu (0-1) kao ulaz u VI.
3. Postaviti funkciju **Random Number** na blok dijagram.
4. Izabrati **While Loop** iz **Functions » Structures** podpalete. *While* petlja izvršava kod unutar svojih granica sve dok se ne pritisne dugme STOP na Front panelu.
5. Postavimo pointer na mjesto dijagrama gdje želimo da počne petlja i prevučemo mišem preko dijela koda za koji želimo da bude u *While* petlji. U ovom slučaju ćemo obuhvatiti funkciju **Random Number**, **Stop Button** i **Random Plot** dijagram.
6. Sačuvati VI pod nazivom *Slučajni brojevi.vi*

- Ožičavanje – Wiring

1. Izabrati *Wiring tool* iz *Tools palette*.
2. Povezati funkciju **Random Number** za terminal **Random Plot** grafa.
3. Povezati terminal **Stop** za *conditional terminal* *While* petlje.
4. Uključiti **Run** na *toolbar-u* da bi se startovao VI.

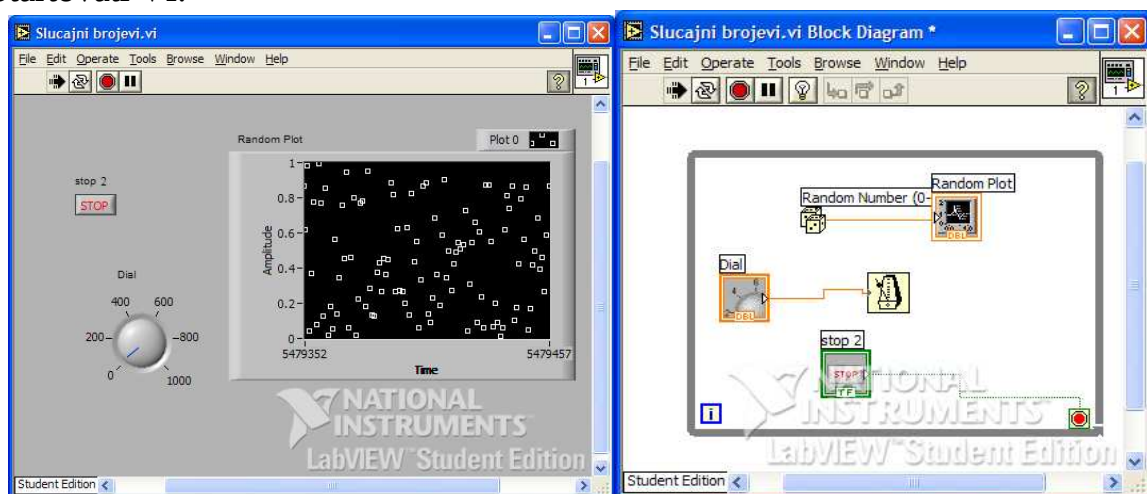
6. Za zaustavljanje VI pritisnuti dugme Stop. Na taj način se prekida izvršavanje *While* petlje.

- Korišćenje vremenske kontrole

1. U blok dijagramu izabrati **Wait Until Next ms Multiple** iz **Functions »Time & Dialog** subpalette.
2. Postaviti funkciju **Wait Until Next ms Multiple** unutar *While* petlje.
3. Izabrati **Numeric Constant** iz **Functions »Numeric** palete, postaviti je unutar *While* petlje i povezati sa **Wait Until Next ms Multiple**.
4. Upisati 250 u konstantu da bi se dobilo kašnjenje od 250 milisekundi između generisanja slučajnih brojeva.
5. Startovati VI.

- On-line vremenska kontrola

1. Iz **Controls »Numeric** izabrati kontrolu **Dial**, postaviti je na *front panel* i imenovati je sa **Kašnjenje**.
2. U *Blok dijagramu* postaviti terminal **Kašnjenje** unutar *While* petlje i povezati ga sa **Wait Until Next ms Multiple** umjesto konstante 250.
3. Startovati VI.

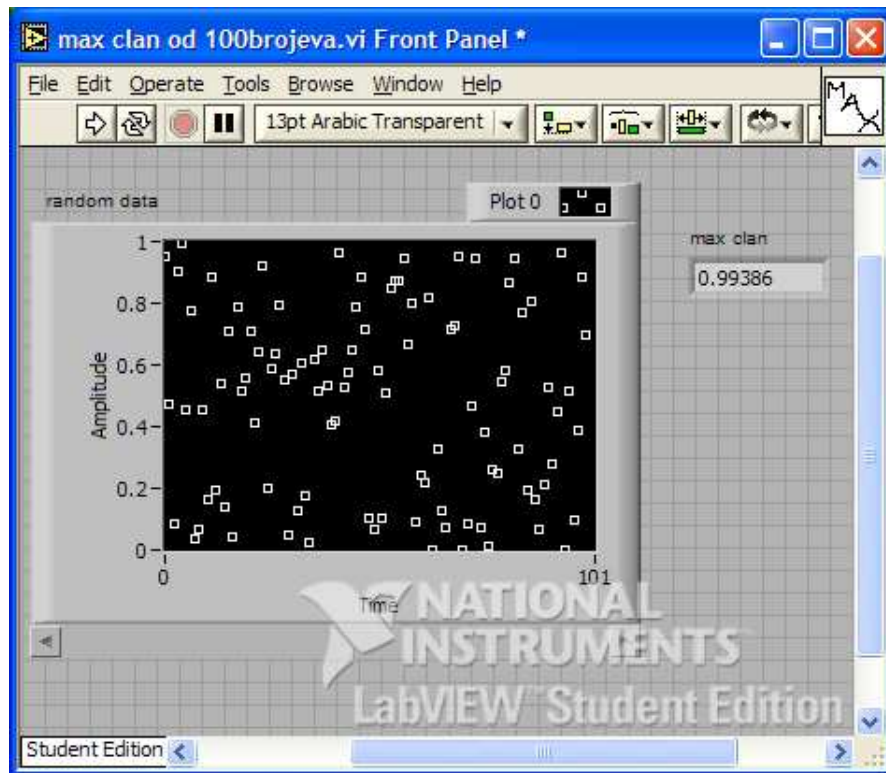


Front panel i Blok dijagram iz zadatka 3

4. Napraviti virtuelni instrument koji generiše 100 slučajnih brojeva vrijednosti od 0 do 100 i prikazuje ih na grafiku. Odrediti najveći dobijeni broj i prikazati ga na Front panelu.

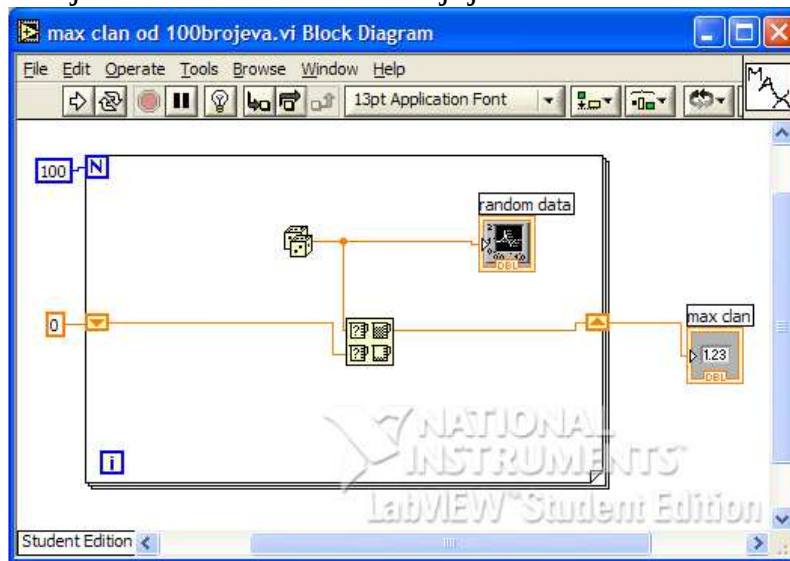
Rješenje:

1. Na Front panelu postaviti grafik iz **Controls->Graph->Waveform Chart**.
2. Postavljeni grafik imenovati sa *Random Data*
3. Zatim postaviti jedan numerički indikator iz **Controls->Numeric->Numeric Indicator**.
4. Postavljeni indikator nazvati *Max clan*.



5. Preći u Blok dijagram.
6. Na Blok dijagram postaviti For petlju iz **Functions->Structures->For loop**.
7. Pomoću For petlje obuhvatiti ikonicu Waveform Chart.
8. U gornjem lijevom ćošku For petlje stoji Loop count. Postaviti numeričku konstantu 100, koja definiše broj izvršavanja For petlje.
9. Postaviti funkciju **Functions->Numeric->Random number (0-1)** unutar For petlje
10. Takođe, postaviti unutar For petlje i funkciju **Functions->Comparison->Max & Min**, a sa lijeve strane For petlje postaviti numeričku konstantu, koja će predstavljati početnu vrijednost pri upoređivanju dva broja.
11. Desnim klikom miša na ivicu For petlje postaviti jedan Shift registar, odabirom opcije iz padajućeg menija **Add Shift Register**.

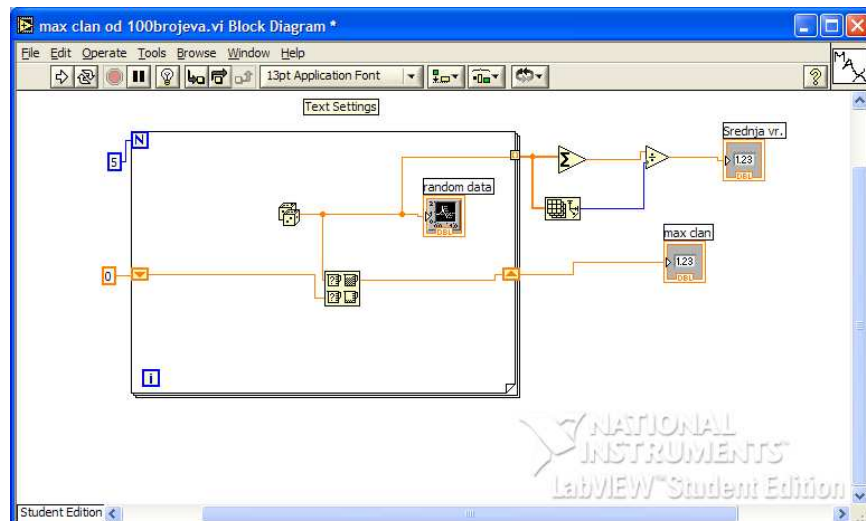
12. Povezati postavljene elemente kao na donjoj slici.



5. Koristeći prethodni zadatak napraviti virtuelni instrument koji vrši izračunavanje srednje vrijednost niza. Omogućiti snimanje dobijenih podataka u tekstualnu datoteku.

Rješenje:

1. Na Front panelu dodati numerički indikator i nazvati ga *Srednja vrijednost*.
2. U Blok dijagram postaviti funkciju *Add Array Elements* iz palete **Functions** »**Numeric**.
3. Izabrati funkciju *Array size* iz palete **Functions** »**Arrays**.
4. Izabrati funkciju *Divide* iz palete **Functions** »**Numeric**.
5. Povezati elemente kao na donjoj slici. Na izlazu signala iz *For* petlje postaviti kursor i kliknuti desnim tasterom miša. U padajućem meniju izabrati opciju *Enable Indexing*.
6. Startovati VI.



- Praćenje toka izvršavanja programa

U slučaju da se želi pratiti tok podataka-izvršavanja koda, u *Diagram* prozoru u *Tool Baru*, treba uključiti opciju *Highlight Execution*.

Startovanjem VI može se posmatrati redosled izvršavanja programa.

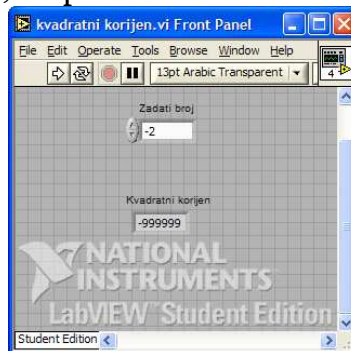
- Snimanje signala u tekstualnu datoteku

1. Izabrati funkciju *Write To Spreadsheet File* iz palete **Functions** » **File I/O**.
2. Startovati VI.
3. Kada se pojavi dialog *Choose file to write* upisati *proba.txt*. Podaci se mogu vidjeti iz bilo kog tekstualnog editora.

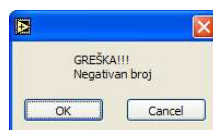
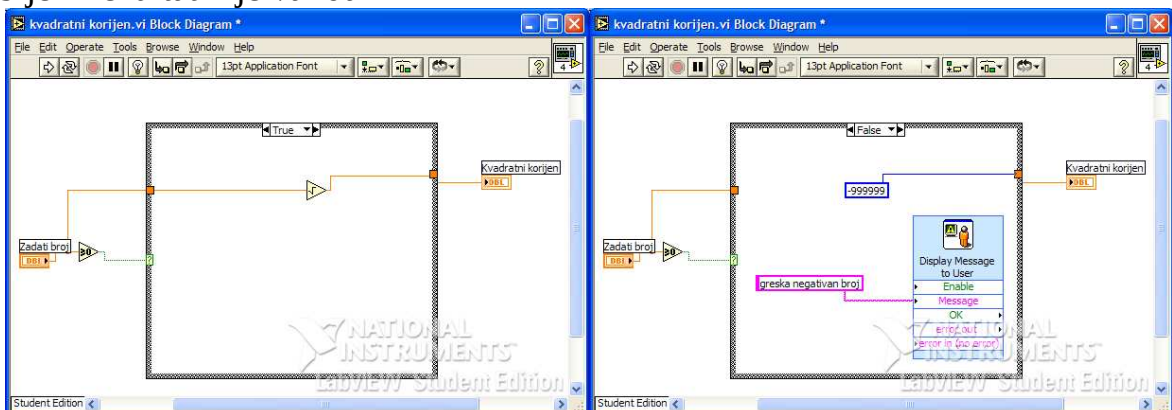
6. Napraviti program koji računa kvadratni korijen iz zadatog broja, i koji će u slučaju zadavanja negativnog broja prijaviti određenu poruku korisniku.

Rješenje:

1. Kreirati novi VI.
2. Na Front panelu postaviti jednu numeričku kontrolu i jedan indikator, i nazvati ih Zadati broj i Kvadratni korijen, respektivno.



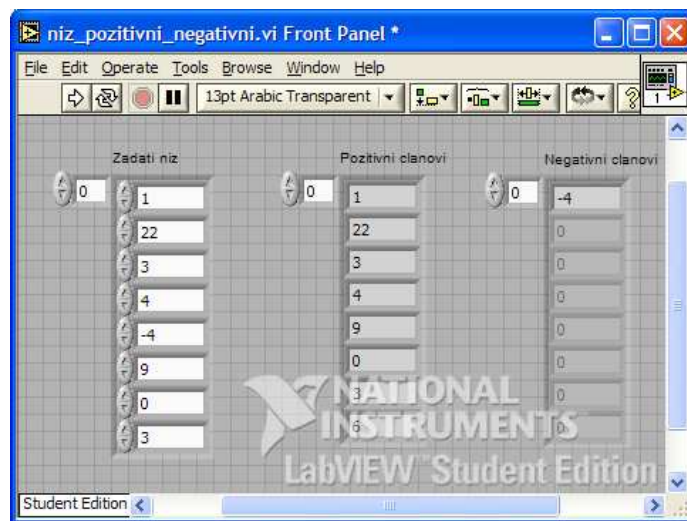
3. U Blok dijagramu postaviti Case strukturu.
4. Izabrati funkciju **Greater or Equal To 0?**, iz **Functions->Comparison**
5. Izabrati funkciju **Square Root** iz **Functions->Numeric**, koja računa kvadratni korijen zadatog broja i postaviti je unutar Case strukture(slučaj **True**).
6. Unutar Case strukture u slučaju **False** postaviti funkciju **Display Message to User**, koja se nalazi u **Functions->Time&Dialog**. Unutar postavljene funkcije upisati poruku korisniku: **GREŠKA! Negativan broj**. Takođe, postaviti i numeričku konstantu koja će u numerički indikator upisati vrijednost **-999999**, kao znak da dobijeni rezultat nije validan



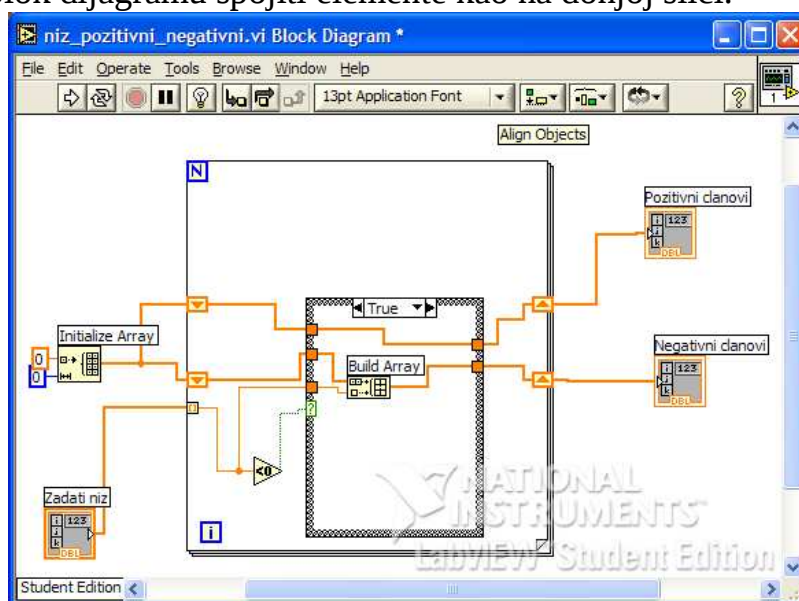
7. Kreirati virtualni instrument koji vrši razdvajanje elemenata niza na pozitivne i negativne, i formira dva nova niza: niz pozitivnih i niz negativnih brojeva.

Rješenje:

1. Izabrati element Array (niz) – **Controls** » **Array & Clusters** » **Array**
2. Odabrati element *Digital control* - **Controls** » **Numeric** » **Numeric control**
3. Postaviti element *Numeric control* u niz
4. Odabrati *Select tool* iz *Tools* palete i povećati niz tako da se prikažu 8 elemenata kao na donjoj slici.



5. Postaviti dva niza *Pozitivni* i *Negativni niz* analogno prethodnim koracima, samo umjesto numeričkih kontrola postaviti **numeričke indikatore**.
6. U blok dijagramu spojiti elemente kao na donjoj slici.



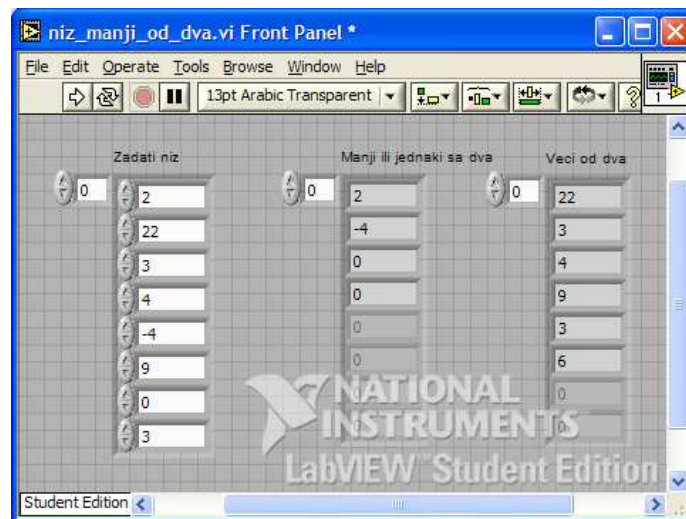
7. *Initialize Array.vi* – **Functions» Array »Initialize Array.vi** koristi se da bi se inicijalizovao novi niz.
8. *Build array* - **Functions» Array »Build array.vi** koristi se za kreiranje novog niza.
9. *Less Than 0?* – **Functions » Comparison » Less than 0?** koristi se za ispitivanje elemenata zadatog niza, da li su manji od nule.
10. Na ivicu *For* petlje postaviti pokazivač miša, kliknuti na desni taster i odabrati opciju **Add Shift Register**(dva puta). Zatim spojiti elemente kao na slici.

8. a) Izmijeniti program tako da se izdvoje elementi niza koji su veći od 2 u jedan niz, a ostali u drugi.
b) Izmijeniti program tako da se izdvoje elementi niza koji su veći od 2 i manji od 4 u jedan niz, a ostali u drugi, računajući i 2 i 4.

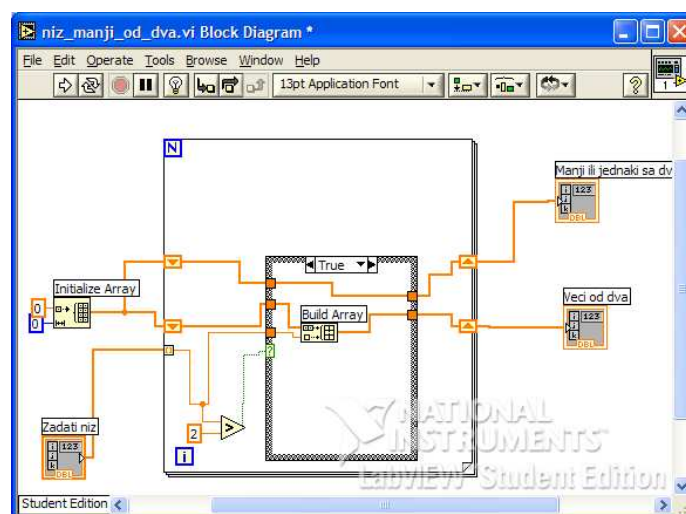
Rješenje:

a)

1. Na Front panelu promijeniti nazive nizova koji se dobijaju nakon izdvajanja, u skladu sa zahtjevima zadatka.

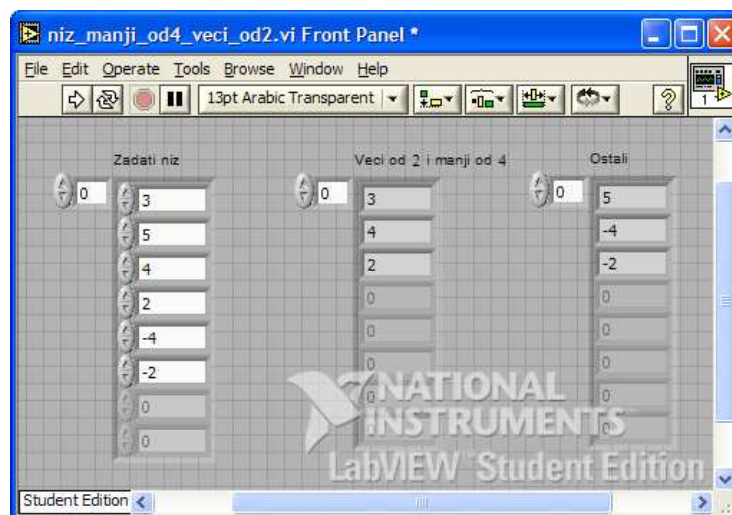


2. U Blok dijagramu je potrebno umjesto 0(nule) kao konstante za upoređivanje sa zadatim elementima niza postaviti broj 2. Umjesto funkcije **Less than 0?** koristiti funkciju **Functions->Comparison->Greater?** i povezati kao na donjoj slici.



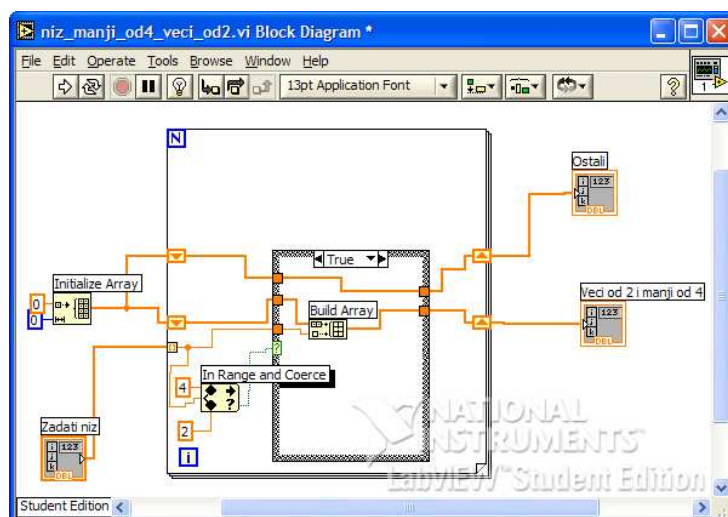
b)

1. Na Front panelu promijeniti nazive nizova koji se dobijaju nakon izdvajanja, u skladu sa zahtjevima zadatka.



2. U Blok dijagramu je potrebno je koristiti funkciju **Functions->Comparison->In Range and Coerce** da bi se izdvojili brojevi u traženom opsegu.

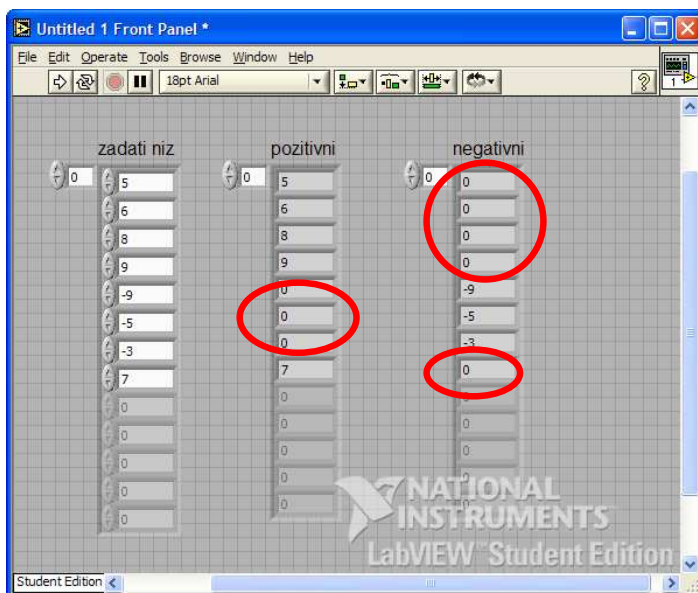
3. Desnim klikom da pomenutu funkciju potrebno je čekirati dvije opcije **Include upper limit i Include lower limit**, i postaviti dvije konstante: 2 i 4 kao granice zadanog opsega (slika)



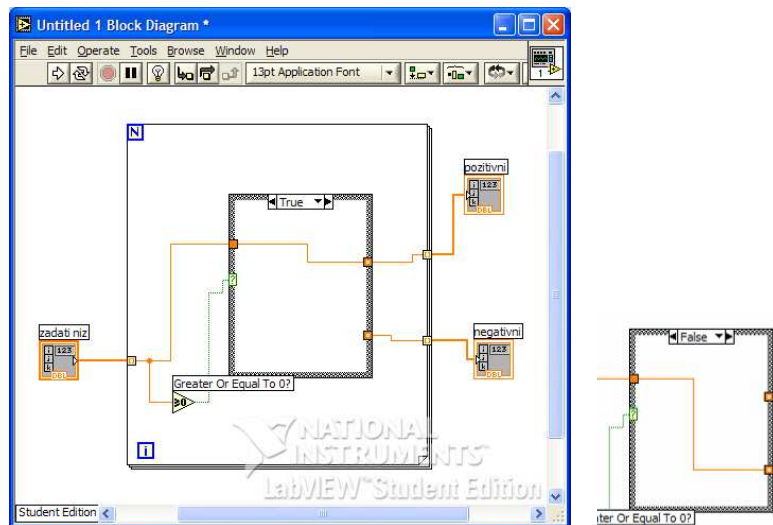
9. Kreirati virtualni instrument koji vrši razdvajanje elemenata niza na pozitivne i negativne, i formira dva nova niza: niz pozitivnih i niz negativnih brojeva.

Rješenje:

1. Izabrati element Array (niz) – **Controls** » **Array & Clusters** » **Array**
2. Odabrati element *Digital control* - **Controls** » **Numeric** » **Numeric control**
3. Postaviti element *Numeric control* u niz
4. Odabrati *Select tool* iz *Tools* palete i povećati niz tako da se prikaže desetak elemenata kao na donjoj slici.
5. Postaviti dva niza *Pozitivni* i *Negativni* niz analogno prethodnim koracima, samo umjesto numeričkih kontrola postaviti **numeričke indikatore**.



6. U blok dijagramu spojiti elemente kao na donjoj slici.



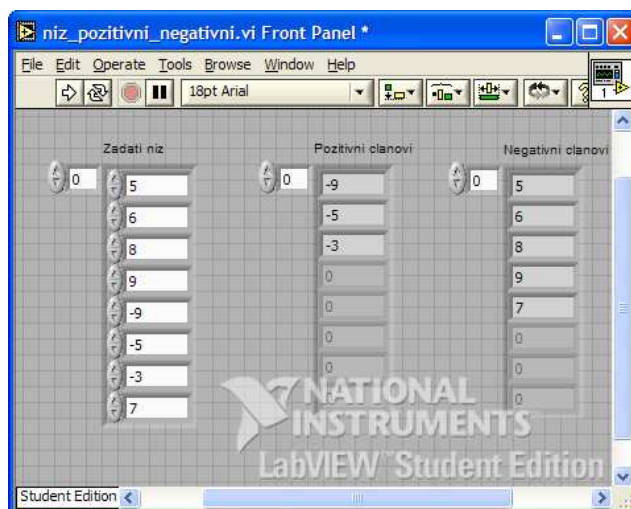
9. *Less Than 0?* – **Functions » Comparison » Greather Or Equal To 0?** koristi se za ispitivanje elemenata zadatog niza, da li su veći ili jednaki sa nulom.
10. Na desnu ivicu Case strukture potrebno je za oba tunela uraditi sledeće: desni klik na tunel -> Use Default If Unwired

Problem:

Na prvoj slici su uokvireni elementi pozitivnog i negativnog niza koji imaju vrijednost nula iako se ta vrijednost ne nalazi u zadatom nizu. Prva četiri elementa niza negativnih brojeva su nule, jer su prva četiri elementa zadatog niza veći od nule i oni su proslijeđeni u niz pozitivnih brojeva. Ovaj problem zauzimanja mjesta u oba niza potrebno je riješiti putem inicijalizacije, tj da svaki niz krene sa formiranjem elemenata od nulte pozicije. U narednom dijelu će biti urađen isti zadatak pomoću funkcija za inicijalizaciju i formiranje niza, da bi se izbjegao navedeni problem.

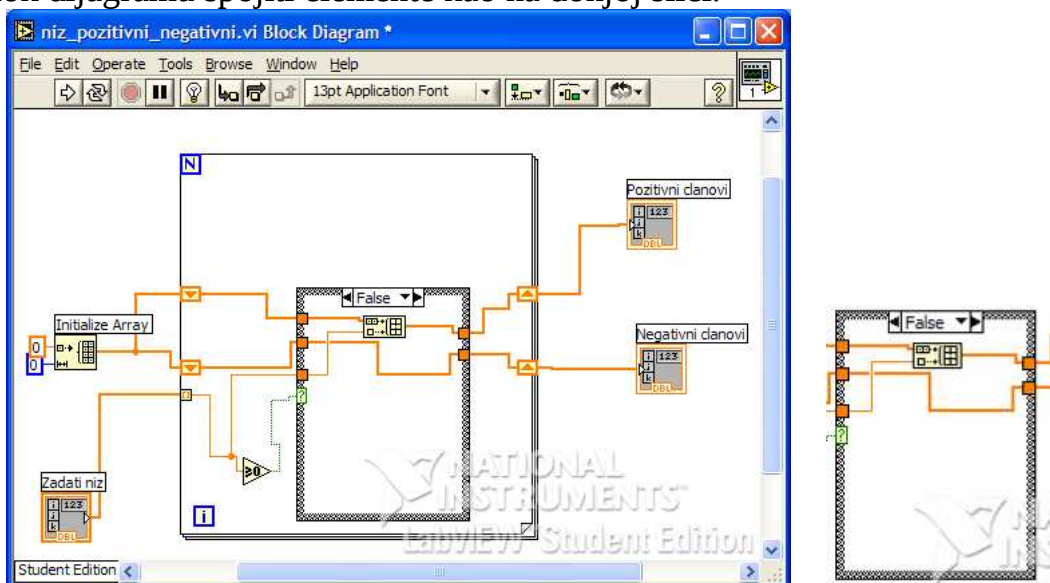
Rješenje 2 (korišćenjem funkcije za inicijalizaciju i pravljenje niza):

1. Izabrati element Array (niz) – **Controls » Array & Clusters » Array**
2. Odabrati element *Digital control* - **Controls » Numeric » Numeric control**
3. Postaviti element *Numeric control* u niz
4. Odabrati *Select tool* iz *Tools* palete i povećati niz tako da se prikažu 8 elemenata kao na donjoj slici.



5. Postaviti dva niza *Pozitivni* i *Negativni niz* analogno prethodnim koracima, samo umjesto numeričkih kontrola postaviti **numeričke indikatore**.

6. U blok dijagramu spojiti elemente kao na donjoj slici.



7. *Initialize Array.vi* – **Functions» Array »Initialize Array.vi** koristi se da bi se inicijalizovao novi niz.

8. *Build array - vi* – **Functions» Array »Build array.vi** koristi se za kreiranje novog niza.

9. *Less Than 0?* – **Functions » Comparison » Greather Or Equal To 0?** koristi se za ispitivanje elemenata zadatog niza, da li su veći ili jednaki sa nulom.

10. Na ivicu *For* petlje postaviti pokazivač miša, kliknuti na desni taster i odabrati opciju **Add Shift Register**(dva puta). Zatim spojiti elemente kao na slici.

Nakon pokretanja programa uočavaju se novodobijeni niz pozitivnih i negativnih brojeva, koji nemaju nulte elemente kao u prethodnom rešenju. Oba niza kreću od nultog elementa, čime je riješen nastali problem.

10. Napisati program u LabVIEW koji od 2 string kontrolera i jednog numeričkog kontrolera sa ulaza:

prvi string: "rezultat mjerenja iznosi"

broj: 5.00

drugi string: "volti"

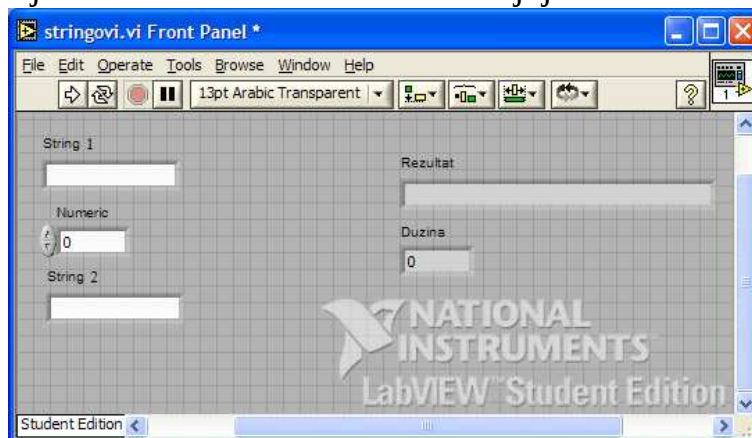
formira jedinstveni string:

"rezultat merenja iznosi 5.00 volti"

i prikazuje dužinu konačnog stringa.

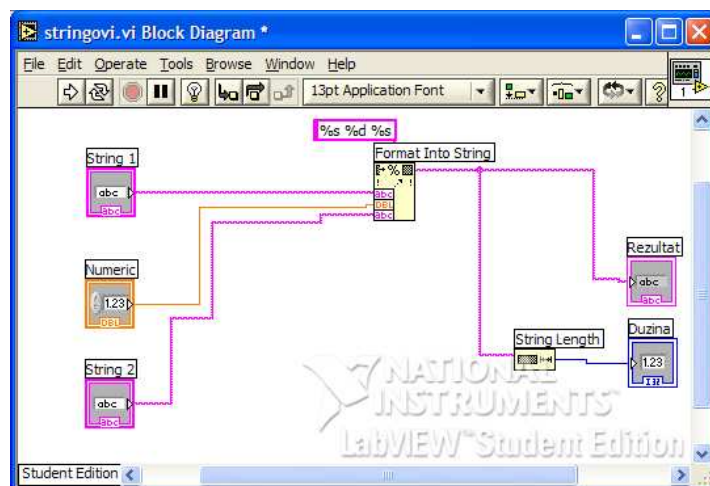
Rješenje:

1. Na Front panel postaviti dva string kontrolera i jedan numerički, i dva string indikatora. Postavljene elemente nazvati kao na donjoj slici.



2. Na Blok dijagramu postaviti sledeće dvije funkcije: **Format Into String** i **String Length**, koje se nalaze u paleti **Functions->String**.

3. Postavljene elemente povezati kao na donjoj slici.



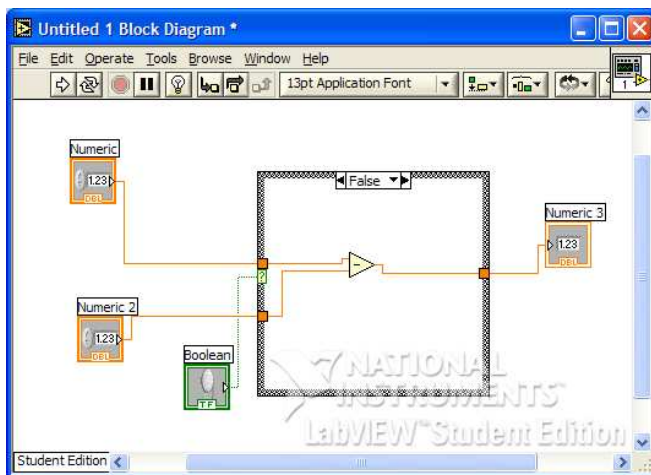
11. Kreirati program koji vrši operacije sabiranja i oduzimanja nad dva unesena broja, i rezultat ispisuje na front panelu. Jednim tasterom omogućiti izbor željene operacije (kao na slici). Snimiti program pod nazivom **prvi.vi**.



Rješenje:

Sličan (samo malo komplikovaniji) zadatak je rađen na vježbama. Tipovi kontrola gdje se unose brojevi su numeričke kontrole, a tip rezultata je numerički indikator. Dugme za biranje operacije može da bude Horizontal Toggle Switch, Vertical Toggle Switch, Slide Switch ili Vertical Slide Switch (iz palete kontrola BOOLEAN).

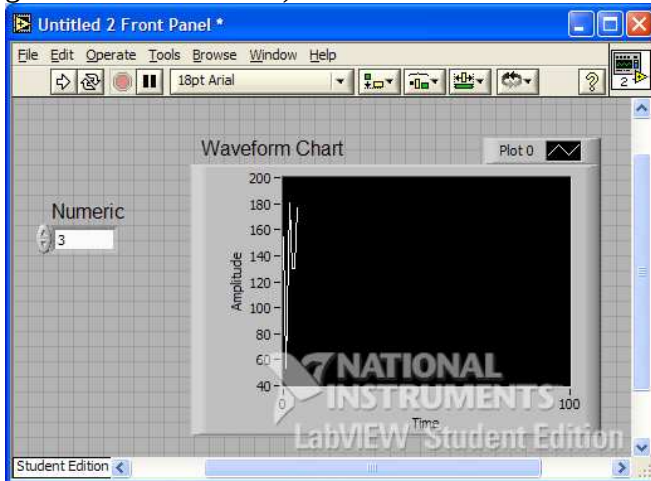
U blok dijagram se ubaci jedna CASE struktura i u jednom slučaju se postavi da vrši sabiranje (na pr. u slučaju TRUE), a u drugom da vrši oduzimanje. Zavisno od toga šta je odabrano pomoću postavljenog BOOLEAN-a program će raditi određenu operaciju.



12. Napraviti program koji generiše slučajne cijele brojeve (zaokružiti na bliži cijeli broj) u intervalu od 0 do 200, prikazuje ih na grafiku i upisuje u tekstualni fajl (slucajni.txt). Omogućiti sa front panela kontrolu broja generisanih elemenata. Snimiti program pod nazivom **drugi.vi**.

Rješenje:

Na Front panel se postavi jedan CHART, jedna numerička kontrola (za određivanje broja generisanih elemenata).



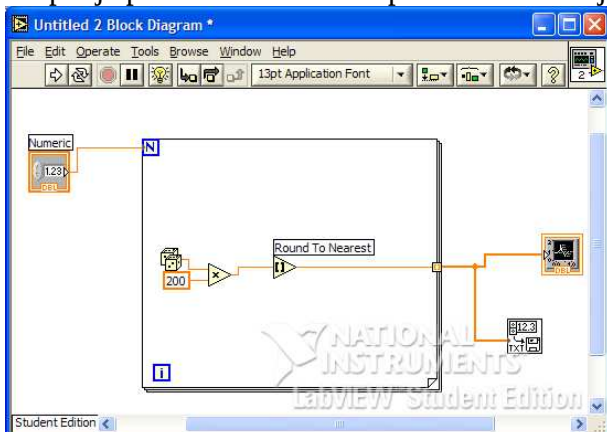
Na Blok dijagram se postavi jedna FOR petlja i poveže se ikonica numeričke kontrole na LOOP COUNT (čime smo odredili koliko puta će se izvršiti for petlja).

U for petlju se unese funkcija za generisanje slučajnih brojeva RANDOM NUMBER(0-1).

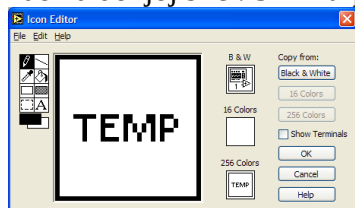
Potrebno je brojeve skalirati od 0 do 200, a to se postiže jednostavnim množenjem brojeva koje generiše Random funkcija.

Pošto je potrebno zaokružiti na bliži cijeli broj koristiti funkciju Round To Nearest.

Da bi upisali dobijene brojeve u tekstualni fajl koristiti funkciju Write to Spreadsheet File i izlaz iz for petlje povezati na terminal pomenute funkcije, 1D data.



13. Napraviti podprogram (subVI) koji simulira akviziciju temperature. Koristiti podprogram *Demo Voltage Read.vi* koji se nalazi u biblioteci *tutorial*. Podprogram treba da ima jedan izlaz i ikonicu kao na donjoj slici. Snimiti podprogram pod nazivom **treći.vi**.



Rješenje:

Da bi napravili podprogram potrebno je odraditi određene korake. Nakon pravilnog kreiranja podprograma postoji mogućnost njegovog korišćenja u drugim programima, kao bilo koje druge funkcije koja je već data u Function biblioteci.

Ovaj podprogram simulira akviziciju temperature što znači da će imati jedan izlaz.

Prvi korak je formiranje front panela. Na front panel postaviti Thermometer (iz palete Numeric).

Na blok dijagram postaviti funkciju Demo Voltage Read koja će da simulira mjerenje temperature.

Ova funkcija se nalazi All Functions-> Select a VI ->Program Files-> National instruments-> LabVIEW 7.0-> vi.lib->tutorial.llb

Dobijeni izlaz iz ove funkcije je potrebno skalirati do 100 i jednostavno ga povezati sa indikatorom Thermometer.

Sledeći korak jeste stvaranje ikonice koja će se koristiti pozivanjem ovog programa u nekom drugom programu.

Za kreiranje ikonice potrebno se vratiti u Front Panel. Duplim klikom otvoriti ikonicu iz gornjeg desnog ugla (ili desni klik pa Edit icon...). Sada je potrebno napraviti ikonicu kakva se traži u tekstu zadatka. Pomoću alata sa desne strane ukloniti sadržaj ikonice(ostaviti samo okvir) i otkucati tekst TEMP. Nakon završetka potvrditi sa OK.

Ovim je završeno kreiranje ikonice, a sad je potrebno formiranje terminala (u ovom slučaju samo jednog izlaza). Odabrati desni klik na novoformiranu ikonicu i odabrati Show Connector. Nakon toga kursor se promijenio u burence sa žicom i potrebno je lijevom klikom kliknuti na ikonicu a zatim na termometar. Nakon toga trebalo bi da je ikonica u gornjem desnom uglu dobila narandžastu boju.

Ovim je završeno kreiranje podprograma i on se sada može koristiti u bilo kom drugom programu.

