Osnovni matematički nizovi u programskom jeziku C++

Babić, Danijel

Undergraduate thesis / Završni rad

2014

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:614780

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
ELEKTROTEHNIČKI FAKULTET

Preddiplomski studij računarstva

OSNOVNI MATEMATIČKI NIZOVI U PROGRAMSKOM JEZIKU C++

Završni rad

Danijel Babić

Osijek, 2014.

ETFOS
ELEKTROTEHNIČKI FAKULTET OSIJEK
Sveučilište Josipa Jurja Strossmayera u Osijeku

Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom studiju

Osijek,

Odboru za završne i diplomske ispite

Prijedlog ocjene završnog rada

Ime i prezime studenta:	Danijel Babić
Studij, smjer:	Preddiplomski, računarstvo
Mat. br. studenta, godina upisa:	3364, 2011.
Mentor:	doc. dr. sc. Tomislav Rudec, dipl. ing.
Sumentor:	
Naslov završnog rada:	Osnovni matematički nizovi u programskom jeziku C++
Primarna znanstvena grana rada:	
Sekundarna znanstvena grana (ili polje) rada:	
Predložena ocjena završnog rada:	
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: Postignuti rezultati u odnosu na složenost zadatka: Jasnoća pismenog izražavanja: Razina samostalnosti:

Potpis sumentora:	Potpis mentora:

Dostaviti:

1. Studentska služba

Potpis predsjednika Odbora:

Dostaviti:

1. Studentska služba

ETFOS ELEKTROTEHNIČKI FAKULTET OSIJEK
Sveučilište Josipa Jurja Strossmayera u Osijeku

IZJAVA O ORIGINALNOSTI RADA

Osijek,
2014.

Ime i prezime studenta:	Danijel Babić
Studij :	Preddiplomski, računarstvo
Mat. br. studenta, godina upisa:	3364, 2011.

Ovom izjavom izjavljujem da je rad pod nazivom**:**

Osnovni matematički nizovi u programskom jeziku C++

izrađen pod vodstvom mentora

doc. dr. sc. Tomislav Rudec, dipl. ing.

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD

Područje nizova brojeva jedno je od najstarijih područja matematičke analize, korišteno još kod starih Grka za pojednostavljivanje i rješavanje raznih problema, poput onih geometrijskih o duljini kružnice ili površini kruga. Kako su svi nizovi promatrani u ovome radu beskonačni, kroz povijest je bilo otežano, ili nemoguće pratiti neka od njihovih svojstava, odrediti članove tih nizova s velikim indeksima i slično. Pojavom računarstva i povezivanjem računarstva s matematikom brojni neriješeni problemi rješavani su s lakoćom, a vrijeme izvršavanja većine operacija i zadataka s nizovima neizmjerno je smanjeno.

Ovaj rad kroz matematičke opise predstavlja neke od tih nizova, s ciljem pripreme za njihova programska opisa. Ponašanja i pravilnosti obrađenih nizova zanimljiva su za programsko promatranje i opisivanje, što je, kroz udruživanje matematičkih relacija i njihovih programskih izvedbi, konačan cilj ovog završnog rada.

Za dobru izvedbu i shvaćanje algoritama primjenjenih u programskom jeziku C++, potrebno je potpuno razumijevanje svih elemenata vezanih uz osnovne matematičke nizove. U prvom poglavlju ovoga rada dane su osnovne definicije niza realnih brojeva, kao i definicije popratnih pojmova, te svojstva takvih nizova. Zatim, u drugom poglavlju su predstavljeni i opisani osnovni nizovi kojima se bavi ovaj završni rad, ispraćeni svim formulama i relacijama nužnima za njihov programski opis. Naposljetku, u trećem poglavlju predstavljeni su osnovni nizovi, kao i dijelovi koda koji opisuju, računaju i ispisuju te nizove.

1.1. Zadatak završnog rada

Zadatak ovog završnog rada je definiranje i predstavljanje osnovnih matematičkih nizova, uz nužno definiranje svih važnih pojmova vezanih uz nizove brojeva. Primjena predstavljenih nizova značajna je u računarstvu, stoga je dio zadatka njihovo predstavljanje i implementacija u programskom jeziku C++.

2. NIZOVI BROJEVA

Niz (slijed) se dobije nizanjem ili svrstavanjem bilo brojeva bilo predmeta jednih za drugima. Brojevi ili predmeti u nizu zovu se članovi. Ako je njihov broj konačan, kaže se da je niz konačan, a ako je njih beskonačno mnogo, kaže se da je niz beskonačan.

2.1. Definicija niza realnih brojeva

Niz realnih brojeva je funkcija $a: A \to \mathbb{R}$, gdje je $A = \{1, 2, \dots, N\}$ ili je $A = \mathbb{N}$. Ako je $A = \mathbb{N}$, onda je riječ o beskonačnom nizu realnih brojeva, inače je niz konačan. Dakle, prirodnim brojevima $n \in A$ redom pridružujemo neke realne vrijednosti $a(n)$.

Općenito se članovi niza obilježavaju sa $a_1, a_2, \dots, a_n$, gdje je a_1 prvi član, a_2 drugi, a vrijednost $a(n)$ niza a na prirodnom broju n označava se sa a_n i naziva n-ti član ili opći član niza. Sam niz a označava se sa (a_n) ili jednostavno $a_1, a_2, \dots, a_n, \dots$. Svaki član niza ima točno određeno mjesto na kojem se nalazi, što niz razlikuje od skupa.

Osnovni niz brojeva je niz prirodnih brojeva: $1, 2, \dots, n, \dots$ do koga se došlo brojanjem. Da je taj niz beskonačan naznačeno je točkicama iza n.

Niz može biti zadan na tri načina:

1. Nabrajanjem prvih nekoliko članova (pri čemu prvih nekoliko članova ne određuje niz, jer pravilo ispisivanja članova niza ne mora biti jednoznačno).
   Primjer 1. 2, 4, 6, 8, 10, 12, ...
2. Općim članom.
   Primjer 2. $a_n = 2n$
3. Rekurzivnom formulom.
   Primjer 3. $a_1 = 2,\ a_n = a_{n-1} + 2$

2.2. Svojstva nizova realnih brojeva

2.2.1. Monotonost i stacionarnost niza

Niz (a_n) raste ako

$$a_n \leq a_{n+1}, \tag{2-1}$$

za svaki *n*.

Niz (a_n) pada ako

$$a_n \geq a_{n+1} \tag{2-2}$$

za svaki *n*.

Niz je monoton ako raste ili ako pada.

Ako je u (2-1), odnosno (2-2) znak stroge nejednakosti, kažemo da niz strogo raste, odnosno strogo pada.

Niz realnih brojeva (a_n) je stacionaran ako postoji prirodan broj n_0 takav da je $a_n=a_{n0}$ za svaki $n \geq n_0$.

2.2.2. Omeđenost niza

Niz (a_n) je omeđen odozdo ako postoji $m \in \mathbb{R}$ takav da je $a_n \geq m$, za svaki $n \in \mathbb{N}$.

Niz (a_n) je omeđen odozgo ako postoji $m \in \mathbb{R}$ takav da je $a_n \leq m$, za svaki $n \in \mathbb{N}$.

Niz je omeđen (ograničen) ako je omeđen odozdo i odozgo, odnosno ako postoje $m, M \in \mathbb{R}$ takvi da je $m \leq a_n \leq M$, za svaki $n \in \mathbb{N}$.

2.3. Gomilište, okolina, limes i konvergencija niza

Postoje tri vrste nizova:

1. Nizovi koji se približavaju jednoj realnoj graničnoj vrijednosti.

   Primjer 4. Članovi niza $a_n = \frac{n}{n+1}$ su: $\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}$, ... koji se sve više približavaju broju 1.
2. Nizovi čiji članovi rastu ili padaju neograničeno.

   Primjer 5. Članovi niza $a_n = n$ su 1, 2, 3, 4, ... koji rastu u $+\infty$.
3. Nizovi koji nisu monotoni niti se približavaju nekoj vrijednosti.

   Primjer 6. Članovi niza $a_n = (-1)^n$ su -1, 1, -1, 1, ... koji neprestano izmjenjuju vrijednosti -1 i 1.

2.3.1. Gomilište

Za realan broj a kažemo da je gomilište niza (a_n) ako se u svakoj ε-okolini broja a nalazi beskonačno mnogo članova tog niza.
Niz dan primjerom 6 ima dva gomilišta, to su -1 i 1.

2.3.2. Okolina

Okolina realnog broja je svaki otvoreni interval (a, b) u kojem taj broj leži. Neka je (a, b)okolina realnog broja r. Tada je $r \in (a, b)$, odnosno $a < r < b$. Kako r ne može biti jednak ni broju a ni broju b, kažemo da r leži u unutrašnjosti intervala (a, b). Neka je $\varepsilon \in \mathbb{R}$. Okolinu $(r - \varepsilon, r + \varepsilon)$ nazivamo simetričnom okolinom ili ε-okolinom realnog broja r.

Okolina realnog broja a je svaki otvoreni interval oko a $O(a, \varepsilon) = \{x \in \mathbb{R} | a - \varepsilon < x < a + \varepsilon\}$, odnosno $O(a, \varepsilon) = \{x \in \mathbb{R} | |a - x| < \varepsilon\}$.
U okolini broja -1 niza danog primjerom 6 nalaze se svi članovi a_{2n-1}, dok se u okolini broja 1 nalaze svi članovi a_{2n} tog niza.

2.3.3. Limes

Za realan broj a kažemo da je granična vrijednost ili limes niza (a_n) ako se u svakoj okolini od a nalazi beskonačno mnogo članova niza, a izvan te okoline samo konačno mnogo njih. Dakle, ako i samo ako za svaki $\varepsilon > 0$ postoji $n_0 \in \mathbb{N}$ takav da za svaki $n > n_0$ vrijedi $|a_n - a| < \varepsilon$.
Ako a_n prilazi broju a blizu kada n postaje sve veći i veći, koristimo oznaku danu izrazom (2-3).

$$\lim_{n \to \infty} a_n = a \tag{2-3}$$

Iako je limes gomilište, svako gomilište nije limes. Razlika između gomilišta i limesa je u tome što se unutar svake okoline gomilišta (koje nije ujedno i limes) nalazi beskonačno članova niza, ali se i izvan te okoline također nalazi beskonačno članova niza.

Ako niz ima limes, onda ima samo jedan limes.

2.3.4. Konvergencija i divergencija

Nizove koji imaju limes nazivamo konvergentnim nizovima. Za nizove koji nisu konvergentni kažemo da su divergentni. Dakle, kod konvergentnog niza svaka okolina od a sadrži beskonačno mnogo članova niza, dok se izvan te okoline nalazi konačno mnogo članova niza.

Za beskonačni niz (a_n) kaže se da je nul-niz ako za proizvoljni $\varepsilon \in \mathbb{R}$ postoji konačno mnogo članova a_n tog niza za koje je $|a_n| \geq \varepsilon$, tj. ako u svakoj okolini broja 0 leži beskonačno, a izvan nje samo konačno mnogo članova tog niza. Neka je $(-\varepsilon, \varepsilon)$ jedna okolina broja 0 i neka je m najveći među indeksima onih članova niza (a_n) kojileže izvan okoline $(-\varepsilon, \varepsilon)$ ili neka je m=0 ako takvih članova nema. Postavimo li n_0=m+1, svi će članovi a_n, za koje je $n \geq n_0$, sigurno ležati u okolini broja 0.

Umjesto navedene definicije nul-niza, može se uzeti i slijedeća, prvoj ekvivalentna definicija: Beskonačni niz (a_n) je nul-niz onda i samo onda ako se svakom realnom pozitivnom ε da pridružiti prirodni broj n_0 takav da bude $|a_n| < \varepsilon$ čim je $n \geq n_0$.

Ako je niz (a_n) nul-niz, kažemo da njegovi članovi teže ili konvergiraju prema nuli i pišemo: $a_n \to 0$ ili $\lim_{n\to\infty} a_n = 0$.

Za beskonačni niz (a_n) kažemo da je određeno divergentan ako:

1. Počevši od određenog indeksa dalje svi njegovi članovi imaju isti predznak.
2. Za po volji odabrani broj $G \in \mathbb{R}$ postoji samo konačno mnogo članova a_n za koje je $|a_n| \leq G$.

Uz navedene pretpostavke, neka je m najveći među indeksima onih članova za koje je $|a_n| \leq G$ ili nula, ako takvih članova nema. Postavimo li n_0=m+1, za sve će članove a_n, za koje je $n \geq n_0$, vrijediti: $|a_n| > G$.

Radi toga je valjana i slijedeća definicija: Niz (a_n) je određeno divergentan onda i samo onda ako su mu, počevši od određenog indeksa svi članovi istog predznaka i ako se svakom $G \in \mathbb{R}$ dade pridružiti prirodni broj n_0 takav da za sve $n \geq n_0$ bude $|a_n| > G$.

Ako je beskonačni niz (a_n) određeno divergentan, onda vrijedi jedno od slijedećeg:

Za niz (a_n) kažemo da teži prema -∞ ako za svaki m ∈ ℝ postoji n_0 ∈ ℕ takav da je $a_n < m$, za svaki $n > n_0$. Za takav niz koriste se oznake $(a_n) \rightarrow +\infty$ ili $\lim_{n\to\infty} a_n = +\infty$.
Takav niz nije omeđen odozgo.

Za niz (a_n) kažemo da teži prema +∞ ako za svaki M ∈ ℝ postoji n_0 ∈ ℕ takav da je $a_n > M$, za svaki $n > n_0$. Za takav niz koriste se oznake $(a_n) \rightarrow -\infty$ ili $\lim_{n\to\infty} a_n = -\infty$.
Takav niz nije omeđen odozdo.

2.4. Periodički nizovi

Niz a_1, a_2, ..., a_k, a_1, a_2, ..., a_k, ..., a_n, ... kod kojeg je a_n=a_{t+sk}, gdje je $t \in I_k, s \in \mathbb{N}$, nazivamo čisto periodičkim nizom kojemu je a_1, a_2, ..., a_k period, a k duljina perioda.
Niz a_1, a_2, ..., a_m, b_1, b_2, ..., b_k, b_1, b_2, ..., b_k, ...,kod kojeg je za n>m, a_n=a_{m+t+sk}, gdje je $t \in I_k$, $s \in \mathbb{N}$, nazivamo mješovito periodičkim nizom. Slog a_1, a_2, ..., a_m zovemo predperiodom dok slog b_1, b_2, ..., b_k zovemo periodom tog niza. Broj m je duljina predperioda, a broj k broj duljina perioda.

2.5. Rekurzije

Kako je zadatak ovog rada predstavljanje osnovnih matematičkih nizova, čije uređenosti i pravilnosti omogućuju i olakšavaju operacije nad njima, prije predstavljanja samih nizova potrebno je predstaviti jednu od metoda kojom je moguće opisati te nizove - metodu rekurzije.
Rekurzija je u matematici i računarstvu metoda definiranja funkcija u kojima se definirajuća funkcija primjenjuje unutar definicije.

Rekurziju je moguće definirati na dva načina:

1) Neka je (a_n) potpuno uređeni niz. Njegovi članovi mogu se određivati jedan za drugim ako su ispunjena ova dva uvjeta:
1. poznat je prvi član niza, a_1;
2. postoji relacija koja povezuje opći član niza a_n s prethodnim članovima.
Relacija se naziva rekurzivna relacija, a za članove kažemo da ih odredujemo rekurzivno.

2) Neka je definiran niz brojeva (a_n). Tada rekurzivnom relacijom (rekurzivnom formulom ili rekurzijom) zovemo svaku relaciju, pomoću koje se n-ti član niza a_n izražava pomoću nekoliko prethodnih članova a_k, $k < n$.

Za nizove određene rekurzijom $a_{n-1} = Aa_n + Ba_{n-1}$, pri čemu su elementi a_1 i a_2 arbitrarni, a A i B konstante, postoji jednostavan način pomoću kojeg se njihovi članovi daju izraziti kao funkcije pripadnih im indeksa. Taj se postupak sastoji u slijedećem: zamjenjujući svaki a_k s x^k (gdje je x za sada neodređen), jednadžba rekurzije prelazi u $x^{n+1} = Ax^n + Bx^{n-1}$. Kako se traže samo ona rješenja te jednadžbe koja nisu jednaka nuli, jednadžbu je moguće podijeliti s x^{n-1}. Dobiva se jednadžba $x^2 - Ax - B = 0$. Neka su x_1 i x_2 njena rješenja.

Postavlja se $a_n = C_1x_1^n + C_2x_2^{n-1}$ ako je $x_1 \neq x_2$, odnosno $a_n = C_1x^n + C_2x_2^{n-1}$ ako je $x_1 = x_2$. Pri tome C_1, C_2 poprimaju vrijednosti ovisno o početnim uvjetima, tj. odabranim vrijednostima za a_1 i a_2.

3. OSNOVNI NIZOVI BROJEVA

Poznavajući definicije i postavke dane u prošlom poglavlju, moguće je predstaviti neke od osnovnih matematičkih nizova.

3.1. Niz prirodnih brojeva

Niz prirodnih brojeva nastao je brojenjem. Niz prirodnih brojeva je neograničen. Rezultat brojenja ne zavisi od redoslijeda kojim se predmetima pridružuju brojevi. Imenovani brojevi izražavaju koliko je jedinica u pojedinim skupovima ili množinama i kakve su one. Neimenovani broj samo kaže koliko ih ima, a ne i kakve su. Broj 1, kao element u skupu, naziva se jedinicom. Svaki prirodni broj izražava od koliko je jedinica sastavljen.

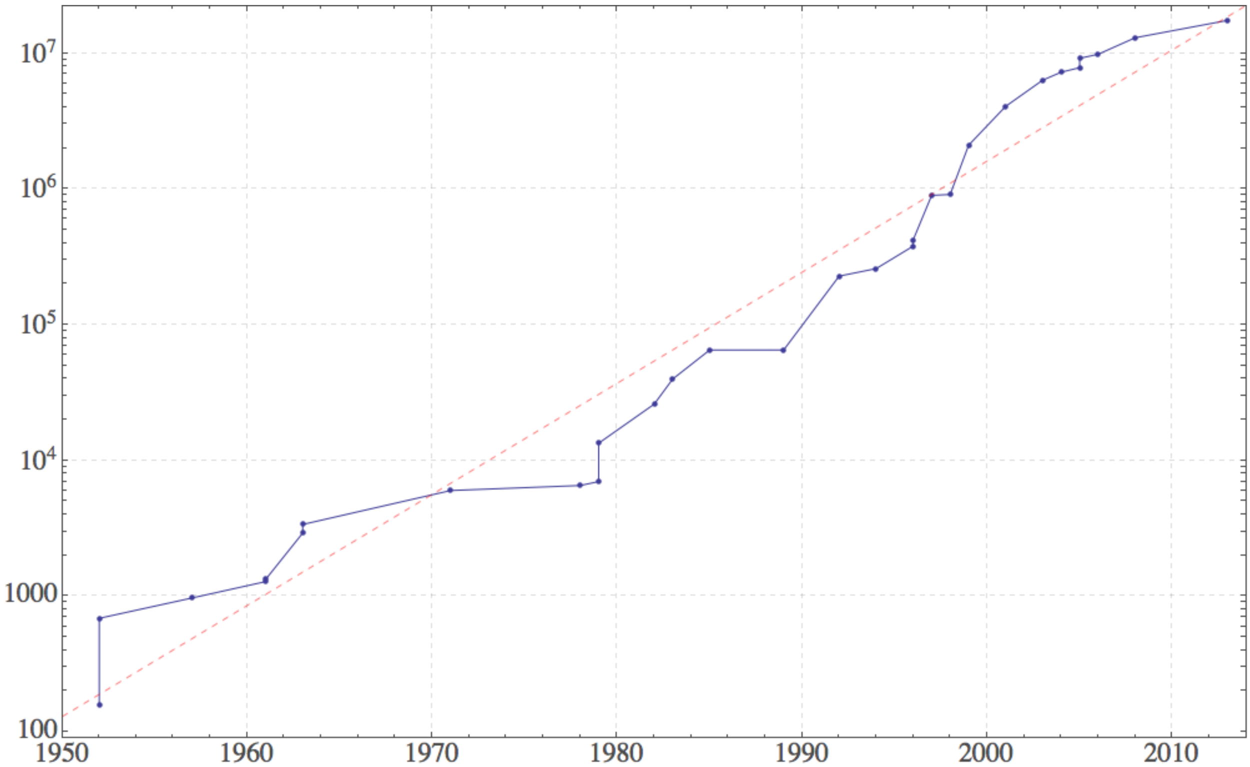
Niz prirodnih brojeva nužno je predstavljen jer se koristi za indeksiranje ostalih nizova. Ako u funkciji cjelobrojnog argumenta $y=f(n)$ argument n poprima samo vrijednosti niza prirodnih brojeva, onda se dobije niz ili slijed vrijednosti funkcije: $f(1), f(2), ..., f(n)$. Tada je svaki član toga niza funkcija svog rednog broja, tzv. indeksa. Zbog toga se u daljnjem tekstu podrazumijeva da je svaki broj n, koji označava redni broj nekog člana niza, iz skupa prirodnih brojeva.

3.2. Niz prostih brojeva

Za prirodni broj a kažemo da je djeljiv sa prirodnim brojem b onda i samo onda ako postoji prirodni broj c takav da je $a=b\cdot c$. Broj koji je djeljiv nekim brojem zove se višekratnik tog broja, a taj broj s kojim je djeljiv se zove mjera ili faktor prvog broja. Svaki produkt kao višekratnik je složen broj i kao takav djeljiv sa svakim svojim faktorom (mjerom). Ako je jedan broj mjera nekog drugog broja, onda je on mjera i višekratnika tog drugog broja. Ako su pribrojnici jedne sume djeljivi nekim brojem, onda je i suma djeljiva tim brojem, ili ako je neki broj mjera dvaju ili više brojeva, onda je on mjera i sume (razlike) tih brojeva.

S obzirom na djeljivost postoje dvije vrste prirodnih brojeva većih od 1: jedni su djeljivi sa samim sobom, to su prosti ili prim brojevi, dok su drugi osim sa 1 i samim sobom djeljivi još i s drugim brojevima, to su složeni brojevi.

Kako prostih brojeva po Euklidovom teoremu ima beskonačno mnogo, a za njihovo određivanje ne postoji matematička formula, otkrivanje prostih brojeva je kroz povijest matematičarima oduvijek predstavljalo izazov. Iako metode za otkrivanje prostih brojeva postoje već tisućljećima, od kojih je najpoznatija metoda Eratostenovog sita (3. st. pr. Kr.), one su izuzetno spore, te je posljednji prosti broj ručno određen, 2^{127}-1, određen 1876. godine, nakon 19 godina ispitivanja. Za dokazivanje idućeg većeg prostog broja bilo je potrebno čekati pojavu računala i 1951. godinu, kada je otkriven novi veliki prosti broj. Slika 3.1. prikazuje sve proste brojeve otkrivene uz pomoć računala, te ujedno prikazuje broj znamenki tih brojeva.



Slika 3.1. Poznati prosti brojevi otkriveni uz pomoć računala [9]

Iz prikazanog grafa može se zaključiti koliko je pojava računarstva utjecala na matematiku, te teoriju brojeva, čiji jedan dio se razmatra ovim radom.

3.3. Aritmetički niz

Aritmetički niz ili aritmetička progresija je niz brojeva sa stalnom razlikom između bilo kojeg člana i njegovog prethodnika. Ta se razlika obilježava sa *d* i zove se razlika ili diferencija aritmetičkog niza, a računa se prema formuli danoj izrazom (3-1). Ako je *d* pozitivan, niz raste, a ako je *d* negativan, niz pada.

$$d = a_n - a_{n-1} \tag{3-1}$$

Dakle, prema definiciji niz $a_1, a_1+d, a_1+2d, a_1+3d, ..., a_1+(n-1)d$ je aritmetički niz.
Opći član aritmetičkog niza je član koji se nalazi na *n*-tom rednom mjestu u nizu, a njegova se vrijednost računa formulom danom izrazom (3-2).

$$a_n = a_1 + (n-1)d \tag{3-2}$$

Za sumu prvih *n* članova aritmetičkog niza formule su dane izrazima (3-3) i (3-4), pri čemu se izraz (3-4) može koristiti ukoliko je poznat član a_n.

$$S_n = \frac{n}{2}(a_1 + a_n) \tag{3-3}$$

$$S_n = \frac{n}{2}(2a_1 + (n-1)d) \tag{3-4}$$

U ove dvije formule za a_n i S_n dolazi pet veličina: a_1, a_n, n, d, S_n, pa se uvijek dvije mogu izračunati ako su poznate ostale tri.

Ukoliko između dva zadana broja *a* i *b* želimo načiniti niz od *r* brojeva, koji zajedno sa *a* i *b* čine aritmetički niz, govorimo o interpolaciji aritmetičkog niza, u kome je *a* prvi član, a *b* posljednji. Ako je *d* diferencija traženog niza, onda se za izračun diferencije pri interpolaciji aritmetičkog niza koristi formula dana izrazom (3-5).

$$d = \frac{b-a}{r+1} \tag{3-5}$$

Aritmetički niz je dobio taj naziv zbog svojstva da je svaki član aritmetička sredina između dva susjedna člana ili između članova jednako udaljenih od njega. Aritmetička sredina više brojeva je suma tih brojeva podijeljena s brojem pribrojnika.

Ukoliko se definicija aritmetičke sredine primjeni na tri uzastopna člana aritmetičkog niza dolazimo do formule dane izrazom (3-6), koja se koristi za računanje drugog od tri uzastopna člana niza, ukoliko su prvi i treći član poznati.

$$a_{n+1} = \frac{a_n + a_{n+2}}{2} \tag{3-6}$$

Zapišemo li izraz (3-6) u obliku izraza (3-7), dobivamo rekurzivnu relaciju uz pomoć koje se svaki član aritmetičkog niza može pronaći uz pomoć dva prethodna člana.

$$a_{n+2} = 2a_{n+1} - a_n \tag{3-7}$$

U ovom slučaju je potrebno znati samo prva dva člana aritmetičkog niza, a_1 i a_2, kako bi aritmetički niz bio određen.

3.4. Geometrijski niz

Geometrijski niz ili geometrijska progresija je niz brojeva sa stalnim kvocijentom između bilo kojeg člana i njegovog prethodnika. Taj kvocijent se označuje sa q i zove se kvocijent geometrijskog niza, a računa se po formuli danoj izrazom (3-8).

$$q = \frac{a_n}{a_{n-1}} \tag{3-8}$$

Dakle, prema definiciji niz a_1, a_1q, a_1q^2, a_1q^3, ..., a_1q^{n-1}, je geometrijski niz.

Opći član geometrijskog niza je član koji se nalazi na n-tom rednom mjestu u nizu, a njegova se vrijednost računa formulom danom izrazom (3-9).

$$a_n = a_1 q^{n-1} \tag{3-9}$$

Za sumu prvih n članova geometrijskog niza formule su dane izrazima (3-10) i (3-11), pri čemu se izraz (3-11) može koristiti ukoliko je poznat član a_n.

$$S_n = a_1 \frac{q^n - 1}{q - 1} \tag{3-10}$$

$$S_n = \frac{a_n q - a_1}{q - 1} \tag{3-11}$$

Ako je $|q| < 1$ tada je moguće odrediti sumu beskonačno mnogo članova niza korištenjem formule dane izrazom (3-12).

$$S_\infty = \frac{a_1}{1 - q} \tag{3-12}$$

U ove dvije formule za a_n i S_n dolazi pet veličina: a_1, a_n, n, d, S_n, pa se uvijek dvije mogu izračunati ako su poznate ostale tri.

Ukoliko između dva zadana broja a i b želimo načiniti niz od r brojeva, koji zajedno sa a i b čine geometrijski niz, govorimo o interpolaciji geometrijskog niza, u kome je a prvi član, a b posljednji. Ako je q kvocijent traženog niza, onda se za izračun kvocijenta pri interpolaciji geometrijskog niza koristi formula dana izrazom (3-13).

$$q = \sqrt[r+1]{\frac{b}{a}} \tag{3-13}$$

Geometrijski niz je dobio taj naziv zbog svojstva što je apsolutna vrijednost bilo kojeg člana geometrijska sredina između apsolutnih vrijednosti susjednih članova ili između apsolutnih vrijednosti članova koji su jednako udaljeni od njega. Geometrijska sredina više pozitivnih brojeva je korijen produkta tih brojeva, pri čemu je stupanj korijena jednak broju faktora.

Promatranjem općeg zapisa geometrijskog niza (a_1, $a_2=a_1q$, $a_3=a_1q^2$, ..., $an=a_1q^{n-1}$) može se zaključiti kako se i geometrijski niz može rekurzivno predstaviti relacijom danom izrazom (3-14).

$$a_{n+1} = qa_n \tag{3-14}$$

3.5. Fibonaccijev niz

Fibonaccijev niz je niz brojeva koji počinje brojevima 0 i 1, a svaki sljedeći broj u nizu dobije se zbrajanjem prethodna dva, što se općenito može prikazati izrazom (3-15).

$$f(n) = f(n-1) + f(n-2) \tag{3-15}$$

Fibonnacijev niz je u ovome radu ubrojan među osnovne matematičke nizove radi svojeg specifičnog svojstva, koje je jednako važno u matematici, kao i računarstvu - rekurzije. Tako je Fibonnacijev niz uzet kao osnovni i najjednostavniji primjer rekurzivno zadanog niza. tj. niza čiji je *n*-ti član moguće odrediti poznavanjem nekoliko prethodnih članova.

4. OSNOVNI NIZOVI U PROGRAMSKOM JEZIKU C++

Programski jezik C++ je objektno orijentirani programski jezik opće namjene, koji spada među jezike više programske razine. Razvijan je 1980-ih od strane Bjarena Stroustrupa kao proširenje programskom jeziku C, s najvećom razlikom u odnosu na C u tome što je u C++ uveden rad s klasama.

Programski kodovi koji se nalaze u ovome poglavlju su ključni dijelovi koda za opisivanje određenih nizova, dok se potpuni kodovi nalaze u prilogu.

4.1. Niz prirodnih brojeva

Kao što je već spomenuto, niz prirodnih brojeva se u računarstvu, kao i u matematici, koristi za indeksiranje nizova s kojima se radi. Ono što razlikuje programski opisane nizove od matematičkih je prvi indeks, tj. indeks prvog člana niza, koji u programskom jeziku C++ ima vrijednost 0. Tako su indeksi svih članova niza opisanog jezikom C++ za jedan manji od indeksa članova niza koji su matematički opisani, odnosno, ukoliko neki niz ima *n* članova, indeksi tih članova u programskom jeziku C++ će redom imati slijedeće vrijednosti: *0, 1, 2, ..., n-2, n-1*.

Kako bi ta neujednačenost u ovome radu bila izbjegnuta, s članovima svih nizova koji su obrađeni u ovome poglavlju, a čiji indeksi su jednaki nuli, neće se raditi.

4.2. Niz prostih brojeva

Kao što je spomenuto u prethodnom poglavlju, ne postoji matematička formula koja bi mogla odrediti članove niza prostih brojeva, stoga je za njihovo otkrivanje potrebno korisiti neku od metoda koja ispituje jesu li pojedini članovi niza prirodnih brojeva prosti. Kako je zadatak ovog završnog rada predstaviti osnovne nizove, te nije potrebno da broj članova nizova prelazi 100, za ispitivanje je li određeni broj prost, korišten je osnovni algoritam, prikazan slikom 4.1.

```
p = true;
for (i = 2; i < j && p==true;i++)
       if (j%i == 0) p = false;
if (p == true) cout << j << "\n";
```

Slika 4.1. Ispitivanje prostog broja

Za ispitivanje i ispis niza prostih brojeva do broja *n* korišten je algoritam prikazan slikom 4.2.

```
cout << 2 << "\n";
for (j = 3; j <= n; j++)
{
	p = true;
	for (i = 2; i < j && p==true;i++)
		if (j%i == 0) p = false;
	if (p == true) cout << j << "\n";
}
```

Slika 4.2. Računanje i ispisivanje *n* članova niza prostih brojeva

4.3. Aritmetički niz

Kako je do člana a_n aritmetičkog niza, kao i do svih članova niza koji mu prethode moguće doći na dva načina, oba načina su prikazana programskim kodovima. Prvi način, koji je vezan uz izraz (3-2), za određivanje člana a_n aritmetičkog niza, te izraz (3-4), za određivanje sume aritmetičkog niza, prikazan je slikom 4.3. Računanje pojedinačnih članova niza, kao i njihov ispis vrši se uz pomoć petlje, koja se izvršava *n* puta, što je ujedno i broj izračunatih članova niza.

```
cout << a[1] << "\n";
for (i = 2; i <= n; i++)
{
	a[i] = a[1] + (i - 1)*d;
	cout << a[i] << "\n";
}
s = (n / 2)*(a[1] + (n - 1)*d);
cout << s;
```

Slika 4.3. Računanje i ispisivanje *n* članova aritmetičkog niza

Drugi način vezan je uz izraz (3-7), za rekurzivno određivanje člana a_n aritmetičkog niza, te je prikazan slikom 4.4.

```
cout << a[1] << "\n";
cout << a[2] << "\n";
for (i = 3; i <= n; i++)
{
	a[i] = 2*a[i-1] -a[i-2];
	cout << a[i] << "\n";
}
```

Slika 4.4. Računanje i ispisivanje *n* članova aritmetičkog niza rekurzivnom formulom

4.4. Geometrijski niz

Kao i kod aritmetičkog, i kod geometrijskog niza je do člana a_n moguće doći na dva načina, te su oba prikazana odgovarajućim programskim kodovima. Prvi dio koda odgovara izrazu (3-9), za određivanje člana a_n geometrijskog niza, te izrazu (3-10), za određivanje sume geometrijskog niza. Spomenuti kod je prikazan slikom 4.5. Računanje pojedinačnih članova niza, kao i njihov ispis vrši se uz pomoć petlje, koja se izvršava n puta, što je ujedno i broj izračunatih članova niza.

```
cout << a[1] << "\n";
for (i = 2; i <= n; i++)
{
	a[i] = a[1]*pow(q,(i-1));
	cout << a[i] << "\n";
}
s = a[1]*((pow(q,n)-1)/(q-1));
cout << s;
```

Slika 4.5. Računanje i ispisivanje n članova geometrijskog niza

Drugi dio koda odgovara izrazu (3-14), izrazu za rekurzivno određivanje člana a_n, te je prikazan slikom 4.6.

```
cout << a[1] << "\n";
for (i = 2; i <= n; i++)
{
	a[i] = q*a[i-1];
	cout << a[i] << "\n";
}
```

Slika 4.6. Računanje i ispisivanje n članova geometrijskog niza rekurzivnom formulom

4.5. Fibonaccijev niz

Zbog načina na koji je zadan, Fibonaccijev niz programski je moguće opisati na više načina, iako je ključni dio, rekurzivna formula koja ga opisuje nužno uvijek ista. Prateći izraz (3-15), algoritam koji računa i ispisuje članove Fibonaccijevog niza prikazan je slikom 4.7.

```
cout << a[1] << "\n";
cout << a[2] << "\n";
for (i = 3; i <= n; i++)
{
	a[i] = a[i-2]+a[i-1];
	cout << a[i] << "\n";
}
```

Slika 4.7. Računanje i ispisivanje *n* članova Fibonaccijevog niza

4.6. Ispitivanje nizova

Kao konačan zadatak rada koji objedinjuje obrađene nizove, napisan je program u kojeg se unose članovi nekog niza, te se potom, prema uvjetima i svojstvima nizova ispituje čine li dani članovi koji od definiranih nizova (aritmetički, geometrijski ili niz zadan prema Fibonaccijevom pravilu).

Kako je za promatranje aritmetičke, ali i geometrijske sredine brojeva potrebno imati najmanje tri broja, program podrazumijeva da je svaki ispitivani niz, koji ima jedan ili dva člana, i aritmetički i geometrijski niz. Jednako tako je i za ispitivanje niza zadanog prema Fibonaccijevom pravilu potrebno znati najmanje tri člana da bi se moglo utvrditi radi li se o takvome nizu, te se također podrazumijeva kako je svaki ispitivani niz, koji ima jedan ili dva člana, niz prema Fibonaccijevom pravilu.

Slikom 4.8. dana je funkcija PrepoznajAritmetickiNiz, koja ispituje jesu li uneseni članovi članovi koji zadovoljavaju svojstva i uvjete aritmetičkog niza. Funkcija PrepoznajAritmetickiNiz je funkcija logičkog tipa koja vraća vrijednost *istina* ili *laž* ovisno o rezultatu provjere niza. Funkcija uzima razlike drugog i prvog člana, pretposljednjeg i posljednjeg člana niza, te ispituje njihov odnos. Ukoliko su razlike jednake, radi se o aritmetičkom nizu, te funkcija vraća vrijednost *istina*. U protivnom se vraća vrijednost *laž*.

Kako bi niz, za kojeg je jednom dokazano kako nije aritmetički, nosio tu oznaku, korištena je varijabla logičkog tipa, naziva isAritmetic, koja više ne može poprimiti vrijednost *istina* nakon što se njena vrijednost prvi puta promijeni u *laž*.

```
bool PrepoznajAritmetickiNiz(double * nizP, int brojClanova)
{
	if (brojClanova < 3)
		return true;
	else
	{
		double Temp1, Temp2;
		Temp1 = nizP[1] - nizP[0];
		Temp2 = nizP[brojClanova - 1] - nizP[brojClanova - 2];
		if (Temp1 == Temp2)
			return true;
		else
		{
			isArtimetic = false;
			return false;
		}
	}
}
```

Slika 4.8. Funkcija za ispitivanje aritmetičkog niza

Slikom 4.9. dana je funkcija PrepoznajGeometrijskiNiz koja ispituje jesu li uneseni članovi članovi koji zadovoljavaju svojstva i uvjete geometrijskog niza. Funkcija PrepoznajGeometrijskiNiz je funkcija logičkog tipa koja vraća vrijednost *istina* ili *laž* ovisno o rezultatu provjere niza. Funkcija uzima kvocijente drugog i prvog člana, pretposljednjeg i posljednjeg člana niza, te ispituje njihov odnos. Ukoliko su kvocijenti jednaki, radi se o geometrijskom nizu, te funkcija vraća vrijednost *istina*. U protivnom se vraća vrijednost *laž*.

Kako bi dijeljenje s nulom bilo izbjegnuto, iako je moguć geometrijski niz čiji su svi članovi jednaki nuli, dio funkcije provjerava koji članovi niza su jednaki nuli, te ovisno o rezultatu vraća vrijednosti *istina* ili *laž*.

Kako bi niz, za kojeg je jednom dokazano kako nije geometrijski, nosio tu oznaku, korištena je varijabla logičkog tipa, naziva isGeometric, koja više ne može poprimiti vrijednost *istina* nakon što se njena vrijednost prvi puta promijeni u *laž*.

```
bool PrepoznajGeometrijskiNiz(double * nizP, int brojClanova)
{
	if (brojClanova < 3)
		return true;
	else
	{
		double Temp1, Temp2;
		if (!(nizP[0] == 0 || nizP[brojClanova - 2] == 0 ))
		{
			Temp1 = nizP[1] / nizP[0] ;
			Temp2 = nizP[brojClanova -1] / nizP[brojClanova -2];
		}
		else
		{

			if (nizP[0] != nizP[brojClanova - 1])
			{
				isGeometric = false;
				return false;
			}
			else
				return true;
		}
		if (Temp1 == Temp2)
			return true;
		else
		{
			isGeometric = false;
			return false;
		}
	}

}
```

Slika 4.9. Funkcija za ispitivanje geometrijskog niza

Slikom 4.10. dana je funkcija FibboNizTest koja ispituje jesu li uneseni članovi članovi koji zadovoljavaju svojstva i uvjete niza prema Fibonaccijevom pravilu. Funkcija FibboNizTest je funkcija logičkog tipa koja vraća vrijednost *istina* ili *laž* ovisno o rezultatu provjere niza. Funkcija uspoređuje razliku posljednjeg i pretposljednjeg člana, s članom ispred pretposljednjeg člana. Ukoliko su razlika i član ispred pretposljednjeg člana jednaki, radi se o nizu prema Fibonaccijevom pravilu, te funkcija vraća vrijednost *istina*. U protivnom se vraća vrijednost *laž*.

U glavnom dijelu programa nalazi se petlja koja šalje svaki novi uneseni član niza svim funkcijama na ispitivanje, a koja prestaje s radom kada niz ne odgovara niti jednome od onih za koje se ispituje, ili kada odgovara samo jednom takvom nizu.

```
bool FibboNizTest(double * nizP, int brojClanova)
{
	if (brojClanova < 3)
		return true;
	else
	{
		if (nizP[brojClanova-1]-nizP[brojClanova-2]==nizP[brojClanova -3])
			return true;
		else
		{
			isFibbo = false;
			return false;
		}
	}
}
```

Slika 4.10. Funkcija za ispitivanje niza prema Fibonaccijevom pravilu

5. ZAKLJUČAK

Ovim završnim radom potpuno su predstavljeni su neki od osnovnih matematičkih nizova, čiji su matematički opisi praćeni i programskim algoritmima. Navedene su i usvojene sve definicije potrebne za opisivanje, definiranje i postavljanje nizova brojeva, kao što i je učinjeno na primjeru pet različitih nizova.

Neki od nizova koje se može ubrojiti u osnovne nizove, poput niza koji daje bazu prirodnog logaritma, broj *e*, ili Cauchyjevog niza, nisu analizirani, jer su njihovo ponašanje i njihova svojstva slična analiziranim nizovima, te se njihovom analizom ne bi znatno proširila stečena znanja o nizovima.

Kako je cilj završnog rada u potpunosti ostvaren, tj. na temelju danih definicija, relacija i opisa moguće je razumjeti i opisati gore navedene nizove, ovaj rad može poslužiti kao dobar i kompletan uvod u područje matematike koje se bavi nizovima, kao i njihovu implementaciju u nekom od programskih jezika.

LITERATURA

[1] M. Degoricija, prof., Algebra 1 (Aritmetika i osnove apstraktne algebre), Pedagoška akademija Slavonski Brod, Slavonski Brod, 1970.

[2] S. Prvanović, Moderna matematika, Zavod za izdavanje udžbenika Socijalističke Republike Srbije, Beograd, 1971.

[3] M. Sevdić, prof., Matematika, Privreda, Zagreb, 1963.

[4] Z. Kurnik, 2014., *Metoda rekurzije*
http://web.math.pmf.unizg.hr/nastava/metodika/materijali/rekurzije.pdf [20. 8. 2014.]

[5] A. Jurasić, 2014., *Nizovi realnih brojeva*
http://www.math.uniri.hr/~ajurasic/predavanje4.pdf, [20. 8. 2014]

[6] I. Slapničar, 2002., *Niz realnih brojeva*
http://lavica.fesb.hr/mat1/predavanja/node126.html [20. 8. 2014.]

[7] I. Slapničar, 2002., *Gomilište i podniz*
http://lavica.fesb.hr/mat1/predavanja/node127.html [20. 8. 2014.]

[8] I. Slapničar, 2002., *Omeđenost, monotonost i konvergencija*
http://lavica.fesb.hr/mat1/predavanja/node128.html [20. 8. 2014.]

[9] 2014., *Largest known prime number*
http://en.wikipedia.org/wiki/Largest_known_prime_number [20. 8. 2014.]

SAŽETAK

OSNOVNI MATEMATIČKI NIZOVI U PROGRAMSKOM JEZIKU C++

U ovome radu definiranje niz realnih brojeva, načini zadavanja takvog niza, kao i pojmovi važni za njegovo razumijevanje, poput gomilišta, okoline, limesa i konvergencije. Definirana su svojstva nizova, svojstvo monotonosti i omeđenosti. Predstavljen je općeniti opis periodičkih nizova, kao i definicija rekurzije, nužna za razumijevanje jednog od mogućih načina zadavanja nizova. Formulama i opisima predstavljeni su neki od osnovnih matematičkih nizova: niz prirodnih brojeva, niz prostih brojeva, aritmetički niz, geometrijski niz i Fibonaccijev niz. Njihove pravilnosti su zanimljive za promatranje i ispitivanje u nekom od programskih jezika. Programski jezik C++ odabran je kao jezik u kojem će se napisati i izvršiti algoritmi napisani uz pomoć formula koje opisuju ponašanje pojedinih nizova. Kao konačan rezultat i zadatak rada, vidljiva je veza između svih spomenutih i korištenih matematičkih relacija i opisa nizova, s njihovom programskom primjenom.

Ključne riječi:

niz brojeva, rekurzija, prirodan broj, prost broj, aritmetički niz, geometrijski niz, Fibonaccijev niz, C++

ABSTRACT

BASIC MATHEMATICAL SEQUENCES IN C++ PROGRAMMING LANGUAGE

This paper defines sequence of real numbers, ways in which such sequence can be specified, and notions that are necessary for its understanding, such as limits and convergence. Basic properties like monotonicity and bound have been defined, as well. General description of periodic sequences is presented, as well as defition of recursion, which is necessary to understand, as it is one of possible methods to set a sequence. Formulas and descriptions have been used to present some of the basic mathematical sequences: sequence of natural numbers, sequence of prime numbers, arithmetic progression, geometric progression and Fibonacci sequence. Their regularities are interesting to observe and perform in some of the programming languages. C++ Programming language has been selected for a language in which the algorithms have been written and executed. As a final result and task of this paper, connection between all of the mentioned and used mathematical relations and description, with their software application, can be seen.

Keywords:

sequence, recursion, natural number, prime number, arithmetic progression, geometric progression, Fibonacci sequence

ŽIVOTOPIS

Danijel Babić

Prebivalište: 32275, Bošnjaci, V. Nazora 224
Boravište: 31000, Osijek, Cvjetkova 4A
Datum i mjesto rođenja: 07.07.1992. Vinkovci

Mobitel: 098/90 63 683
E-mail: danijel.babic1@gmail.com

Obrazovanje

Prirodoslovno-matematička gimnazija, Gimnazija Županja, Županja (2008..-2011.)

Državna matura položena 4.srpnja 2011.

Trenutno: student 3. godine, Elektrotehnički fakultet u Osijeku, sveučilišni studij, smjer Računarstvo (prosjek ocjena 4.8)

Natjecanja, nagrade i priznanja

Tijekom cijelog osnovnoškolskog i srednjoškolskog obrazovanja – natjecanja iz matematike i informatike:

1. razred Gimnazije – 2.mjesto na Županiji na natjecanju iz matematike

2. razred Gimnazije – 3.mjesto na Županiji na natjecanju iz matematike

3. razred Gimnazije – 1. mjesto na Županiji na natjecanju iz matematike

4. razred Gimnazije – 2.mjesto na Županiji na natjecanju iz matematike s pozivom na Državno natjecanje, 1.mjesto na Županiji na natjecanju iz informatike

Posebna znanja i vještine

Znanje jezika: engleski (aktivno govor, čitanje i pisanje), njemački (pasivno govor, čitanje i pisanje)

Poznavanje rada na računalu: MS Office, C/C++, HTML, OOP

Položen vozački ispit B kategorije.

PRILOZI

Prilog P.4.

Potpuni kodovi svih obrađenih matematičkih nizova, napisani u programskom jeziku C++