Rješavanje zadataka u pseudojeziku

poznavati i primijeniti standardne algoritme za:

– zamjenu sadržaja dviju varijabli

– prebrojavanje prema zadanome kriteriju

– zbrajanje prema zadanome kriteriju

– izračun srednje vrijednosti brojeva

– pretraživanje prema zadanome kriteriju

– traženje najmanjega i najvećega među (učitanim) brojevima

– rad s prirodnim brojevima

* 1. **Zamjena sadržaja dviju varijabli**

**koristi se treća pomoćna varijabla** u koju se spremi prva (z postaje x), prva varijabla postaje jednaka vrijednosti druge (x=y), a druga varijabla postaje pomoćna (y=z).

Primjer 1.

ulaz(x,y);

z:=x;

x:=y;

y:=z;

Izlaz('x=',x,'y=',y);

kraj

* 1. **Prebrojavanje prema zadanom kriteriju**

Primjer 2. Napisati program u pseudojeziku koji će ispisati koliko je brojeva od 1 do 100 djeljivo sa 4.

Kod prebrojavanja **koristimo varijablu koju nazivamo brojač**. Ta varijabla se na početku programa postavi na 0,a kada se god desi događaj koji brojimo ona se uveća za 1.

Primjer brojača b:=b+1

početak

b:=0;

za n:=1 do 100 činiti

 ako je n mod 4=0 onda b:=b+1;

izlaz(" brojeva od 1 do 100 koji su djeljivi sa 4 ima",b);

kraj

Primjer 3. Napisati program u pseudojeziku koji će ispisati koliko je brojeva od 1 do 100 djeljivo sa 2 i sa 3.

početak

b:=0;

za n:=1 do 100 činiti

 ako je n mod 2=0 and n mod 3=0 onda b:=b+1;

izlaz(" brojeva od 1 do 100 koji su djeljivi sa 2 i 3 ima",b);

kraj

Koji će brojevi biti ispisani nakon izvođenja sljedećeg programskog odsječka?

m := 5;

dok je m<10 činiti

{

m := m - 2;

izlaz(m);

}

* 1. **zbrajanje prema zadanom kriteriju**

Primjer: Izračunati sumu svih dvoznamenkastih brojeva koji su parni.

Kod zbrajanja prema zadanom kriteriju uvijek uzmemo varijablu u kojoj će se čuvati trenutna suma. ta varijabla se isto na početku postavi na 0 (s=0). Kada se desi događaj koji brojimo tj. našli smo parni dvoznamenkast broj onda se suma uveća za taj broj (s=s+n). Na kraju programa s je suma svih tkvih brojeva jer se u svakom koraku uvijek prethodni iznos sume uveća za novi pribrojnik.

početak

s:=0;

za n:=10 do 99 činiti

 ako je n mod 2 =0 onda s:=s+n;

izlaz(' suma parnih dvoznamenkastih brojeva iznosi',s);

kraj

* 1. **pretraživanjem prema zadanom kriteriju**

 Primjer: unosi se 20 brojeva sa tipkovnice. Napisati program u pseudojeziku koji će ispisati da li se pojavio broj 5.

početak

b:=0;

za n:=1 do 20 činiti

 {

 ulaz(br);

 ako je br =5 onda b:=1;

 }

ako je b=1 onda

izlaz('u nizu je bio broj 5')

inače

izlaz('u nizu nije bilo broja 5');

Kraj

Što bi u programu trebalo promijeniti da se ispiše koliko je bilo petica.

* 1. **za izračun srednje vrijednosti brojeva**

Primjer: Izračunati srednju vrijednost od 30 učitanih brojeva

Kod računanja srednje vrijednosti (prosjeka) potrebno je sumirati sve brojeve i podijeliti ih sa brojem komada

početak

s:=0

za n:=1 do 30 činiti

 {

 ulaz(br);

 s:=s+br;

 }

pr:=s/30;

izlaz ('prosjek unesenih brojeva je', pr);

kraj

Primjer: Na ulazu se unose brojevi sve dok se ne unese broj 0. Ispisati srednju vrijednost unesenih brojeva. Ovdje je veći problem jer ne znamo koliko se brojeva unese pa ih je uz sumiranje potrebno i brojiti.

s:=0;

b:=0;

ponavljati

 {

 ulaz(broj);

 s:=s+broj;

 b:=b+1;

do broj=0; }

izlaz(s/b);

Kod petlji kod kojih ne znamo koliko puta će se ponoviti korisno je upotrijebiti petlju ponavljati- do uvjet

* 1. **za traženje najmanjeg i najvećeg broja među učitanima**

 Primjer: Unijeti 20 brojeva i ispisati najmanji

 Kod algoritma za najmanji broj uvijek se koristi neka varijabla u kojoj se čuva trenutni najmanji broj. Na početku programa postavimo ju uvijek na prvi broj koji se unese. Pri tome se svaki sljedeći uneseni broj uspoređuje sa tim minimumom. Ako je kojim slučajem novi broj manji od trenutnog minimuma onda on postaje minimum

početak

ulaz (broj);

min:=broj; ….........(unijeli smo prvi broj i min postaje jednak njemu(to je trenutni najmanji broj)

za n:=2 do 20 činiti

 { ulaz(broj);

 ako je broj<min onda min:=broj;}

izlaz(' najmanji broj je',min);

kraj

Za traženje najvećeg broja sve je isto samo se znak nejednakosti < mijenja u veće i naravno tu varijablu možemo nazvati npr max.

* 1. **za rad sa prirodnim brojevima**

to može biti bilo kakav problem pa su neki primjeri dani u zadacima ispod kao npr okretanje broja naopako, provjera znamenaka desetice, jedinice, i sl.

* 1. Napiši pseudokod za program koji upisuje prirodan broj pa računa i ispisuje sumu svih dvoznamenkastih brojeva većih od upisanog broja. (Napomena: upisani broj ne mora biti dvoznamenkast)

ulaz(a);

s:=0;

ako je a<10 onda

 za i:=10 do 99 činiti s:=s+i;

 inače

 ako je (a>9)i(a<=99) onda

 za i:= a do 99 činiti s:=s+i;

 inače s:=0;

izlaz('suma dvoznamenkastih brojeva većih od',a,'iznosi',s);

* 1. Potrebno je unijeti troznamenkasti broj a ispisati troznamenkasti broj koji ima sve tri iste znamenke koje su jednake znamenki desetica učitanog broja. Npr za uneseni 123 treba ispisati 222

znamenku jedinice uvijek dobijemo broj mod 10

znamenku desetice uvijek dobijemo broj div10mod10 itd

{

ulaz (a);

d:=adiv10mod10;

b:=d\*100+d\*10+d; ……………………….drugi način b:=d\*111;

izlaz(b);

}

* 1. Napiši algoritam u pseudojeziku koji učitava realan broj na 3 decimale, zaokružuje taj broj na

 dvije decimale i ispisuje ga

 (ako broj pomnožimo sa 1000 onda je cijeli broj , nađemo zadnju znamenku i ako je> 5 znamenku desetice uvećamo za 1. Pr: 123.458…..izlaz je 123.46

{

ulaz(b);

b1:=b\*1000;

d3:=b1 mod 10; ……………………zadnja decimala

d2:=b1div10mod10; …………………..predzadnja decimala

d1:=b1div100mod10;

ako je d3>5 onda d2:=d2+1;

b2:=b1div1000+d1\*0.1+d2\*0.01;

izlaz(b2);

}

* 1. Napisati program koji učitava prirodni broj. Ako učitani broj nije troznamenkasti, treba ispisati odgovarajuću poruku i završiti izvođenje programa. Ako je učitani broj troznamenkasti, program treba ispisati broj koji čine znamenke učitanog broja i njemu zrcalno simetričnog broja. Npr. ako je zadan broj 674, program treba ispisati broj 674476.

ulaz(a);

ako je (a<100)ili(a>999) onda izlaz('broj nije troznamenkast');

 inače

 {

 b:=a\*1000+(amod10)\*100+(adiv 10mod10)\*10+ a div 100;

 izlaz (b);

 }

* 1. Napiši program koji će učitati tri broja i ispisati najmanji

{

ulaz(a,b,c);

min:=a;

ako je b<min onda min:=b;

ako je c< min onda min:=c;

izlaz(min);

}

* 1. Napiši program koji će učitati tri broja i ispisati ih poredane po veličini od najmanjeg do najvećeg.

{

ulaz(a,b,c);

min:=a;

ako je b<min onda min:=a;

ako je c<min onda min:=c; …………….nakon ovoga u min je najmanji broj

max:=a;

ako je b>max onda max:=b;

ako je c>max onda max:=c; …………………nakon ovoga u max je najveći broj

ako je (a<>min)i(a<>max) onda mid:=a; ……ako a nije ni min ni max onda je on srednji broj

ako je (b<>min)i(b<>max) onda mid:=b;

ako je (c<>min)i(c<>max) onda mid:=c;…………….zadnje 3 naredbe traže srednji broj

izlaz(min,mid,max);

}

* 1. Učitati tri prirodna broja. Ako su sva tri učitana broja parni brojevi, program treba ispisati učitane brojeve poredane po veličini od najmanjeg prema najvećem. Ako bilo koji od učitanih brojeva nije paran broj, program treba ispisati učitane brojeve poredane po veličini od najvećeg prema najmanjem

Ovaj zadatak je nadogradnja prethodnog jer treba samo na početku testirati uvjet tj. ispitati da li su brojevi parni

ulaz(a,b,c);

min:=a;

ako je b<min onda min:=a;

ako je c<min onda min:=c;

max:=a;

ako je b>max onda max:=b;

ako je c>max onda max:=c;

ako je (a<>min)i(a<>max) onda mid:=a;

ako je (b<>min)i(b<>max) onda mid:=b;

ako je (c<>min)i(c<>max) onda mid:=c;

ako je (amod 2=0)i (bmod 2=0)i(cmod2=0) onda izlaz(min,mid,max)

 inače izlaz( max,mid,min);

* 1. Napiši program za računanje potencije . Ulazni podaci su brojevi a i b. (b - prirodan broj).

57 = 5\*5\*5\*5\*5\*5\*5…………….b puta se broj množi sam sa sobom

ulaz(a,b);

p:=a;

za i:=2 do b činiti

 p:=p\*a;

izlaz(p);

* 1. Napiši algoritam u pseudojeziku koji učitava dekadski cijeli broj i pretvara ga u binarni

Uputa: binarne znamenke su ostaci kod dijeljenja sa 2 , a novi broj je broj div 2. Ideja je da dobiveni binarni broj napravimo dekadskim koji onda okrenemo. Npr 11 će dati najprije 1101 a onda ćemo okrenuti i dobiti 1011

ulaz(broj);

x:=0;

dok je broj>0 činiti

 { x:=x\*10+broj mod 2;

 broj:=broj div 2;}

y:=0;

dok je x>0 činiti

 { y:=y\*10+x mod 10;

 x:=x div 10;} ……….. ovaj blok naredbi je algoritam za okretanje broja

izlaz(x);

* 1. Napiši pseudokod za program koji unosi prirodni broj u oktalnom brojevnom sustavu i pretvara ga u broj u dekadskom brojevnom sustavu. Program ispisuje broj u dekadskom brojevnom sustavu (u programu se mogu koristiti operatori div za rezultat cjelobrojnog dijeljenja i mod za ostatak cjelobrojnog dijeljenja)

Pr: 23458= 5\*80+4\*81+3\*82+2\*83

Prvo ćemo okrenuti broj a onda znamenku po znamenku množiti sa 8 tj od broja 2345 dobit ćemo 5432 i onda ((2\*8+3)\*8+4)\*8+5

ulaz(b);

m:=0;

dok je b>0 činiti

 { m:=m\*10+b mod 10;

 b:=b div 10;}

y:=0;

dok je m>0 činiti

 { y:= y\*8+ m mod 10

 m:=m div 10;}

izlaz (y);

* 1. Napišite program koji učitava ukupni broj vozila te ukupni broj guma na parkiralištu. Vozila mogu biti automobili ili motori, pri čemu automobil ima četiri gume, a motor dvije gume. Program treba ispisati koliko na parkiralištu ima automobila, a koliko guma.

Pr : n ..broj vozila od toga je x automobila i y motora x + y = n

 m… broj guma x\*4+y\*2=m

 Moramo riješiti pješice taj sustav

y=(-4n+m)/(-2)

x=(6n-m)/2

{

ulaz(n,m);

x:=(-4\*n+m)/(-2);

y:=(6\*n+m)/2;

izlaz(' Na parkirialištu ima',x,'automobila i ',y,'motora')

}

Jakov ima p prijatelja koje želi počastiti. Počastit će ih s ukupno k kolača. Jakov svoje prijatelje želi počastiti tako da svi dobiju podjednak broj kolača pa ih je poslagao u red. Prvom u redu dao je prvi kolač, drugom u redu dao je drugi kolač i tako redom do posljednjeg (p-tog) prijatelja. Nakon toga vratio se na početak reda i nastavio dijeliti kolače istim redom sve dok nije podijelio sve kolače. Očito je da se na ovaj način moglo dogoditi da su neki prijatelji dobili po jedan kolač više od ostalih prijatelja.

Jakova na kraju zanima koliko je najmanje kolača b dobio svaki prijatelj te koliko je prijatelja m dobilo jedan kolač manje od drugih prijatelja. Napišite program u pseudojeziku koji učitava broj prijatelja p i broj kolača k te ispisuje podatke koji zanimaju Jakova.

ulaz(p);

ulaz(k);

b := k div p;

m := p – k mod p;

ako je m = p onda m := 0;

izlaz (b);

izlaz (m);