Matematički programski paketi u saobraćaju i transportu

Programiranje u Matlabu

Drugi deo - petlje

Izbegavanje petlji vektorizacijom

- Napraviti 500 x 500 matricu ciji su svi elementi jednaki pi.

```
for i =1:500
    for j =1:500
        A(i,j)=pi;
    end
end
```

Isti zadatak bez petlji:

A = pi*ones(500,500);

- Uporediti vremena potrebna za resenja sa i bez petlji, koristeci naredbu etime(t0,t1)

ili tic toc naredbu. Pogledati help za više informacija o ovim naredbama.

Zadaci:

- Izračunati sumu$\sum_{k=0}^{n} \frac{(-1)\hat{}k}{2^k} k$ sa for petljom i bez petlje, gde n unosi korisnik.

- Napisati program u skriptu koji odredjuje i ispisuje n-ti Fibonačijev broj, gde n unosi korisnik. Fibonačijevi brojevi su 1,1,2=1+1,3=1+2,5+2+3,...f(n) =f(n-2)+f(n-1)

- Koristeći for petlju napisati program u skriptu koji iscrtava grafike funkcije f(x)=cos(nx) za n = 1,2,3,4,5,6 i 0 <=x<=2*Pi.

- Korisnik unosi dva niza x i y. Koristeći for petlju kreirati matricu A čiji elementi su definisani kao $A_{ij} = x_i y_j$.

While petlja

```
while  uslovni izraz
     grupa Matlabovih komandi
end
```

- Koristi se kada nije unapred poznat broj prolazaka kroz petlju
- Prestaje da prolazi kroz petlju kada uslovni izraz postane netačan
- Pažnja: beskonačna petlja

Primer while petlje

- Izračunati sumu prvih n prirodnih brojeva

manju od unapred date granice limit.

```
suma = 0;
n = 0;
while suma <= limit
n = n + 1;
suma = suma + n;
end
fprintf('suma : %i broj: %i \n', suma, n);
```

Zadaci:

1. Koristeći Tejlorov polinom funkcije sin(x)

$\sum_{k=0}^{\infty} \frac{(-1)^k x^{2k+1}}{(2k+1)!}$ izračunati sa tačnošću od 0.0001 vrednost sin(x) gde je x vrednost ugla u stepenima koji unosi korisnik.

2. Napisati program koji odredjuje kusur koji kasa za samousluživanje treba da vrati korisniku. Program generiše cenu kao slučajanu celobrojnu veličinu manju od 2000. Zatim mašina pita korisnika kojom novčanicom plaća – i to mora biti jedna novčanica iznosa 2000, 1000, 500, 200, 100,50, 20, 10, 5, 2 ili 1 dinar. Ako je uneta novčanica manja od cene ispisuje se odgovarajuća poruka. Ako je novčanica dovoljna, program izračunava kusur i ispisuje kojim novčanicama se kusur isplaćuje, tako da se upotrebi najmanji mogući broj novčanica.

zadatak 3.2.2 iz Zbirke iz Planiranja saobraćaja

MODELI PROSTORNE RASPODELE KRETANJA

Data je matrica razmene kretanja u postojecem stanju (Matrica t). Na osnovu modela prosecnog faktora rasta prognozirati buducu razmenu kretanja izmedju zona. Kriterijum 0.95<F< 1.05 neka bude merodavan za broj iteracija

t=[0 20 40; 50 0 15; 40 15 0];

t_ij – postojeći broj kretanja iz i u j

t_i, t_j – ukupan postojeći broj kretanja iz zone i i u zonu j

F_i - faktor porasta zone i

T_i - ukupan broj budućih putovanja (dobijem modelom nastajanja kretanja)

U modelu prosecnog faktora porasta je

T_ij=t_ij*(F_i+F_j)/2 – buduci broj kretanja iz i u j

F_i=T_i/t_i

T_vrste = [90 80 85];T_kolone = [120 55 75];

Na naredna dva slajda su slike zadatka i rešenja iz Zbirke. Mi ćemo ga rešiti na dva načina: koristeći petlje i bez petlji.

gde je:

T_{ij} – budući broj kretanja iz i u j
t_{ij} – postojeći broj kretanja iz i u j
F_i i F_j – faktori porasta zone i i zone j
T_i i T_j – ukupan prognozirani broj kretanja iz zone i i u zonu j
t_i i t_j – ukupan postojeći broj kretanja iz zone i i u zonu j

U opštem slučaju se dobijene vrednosti za ukupan broj ciljnih i izvornih kretanja po zoni neće poklapati sa vrednostima dobijenim modelom nastajanja kretanja, pa je potrebno model kalibrisati, tj. korigovati faktor porasta.

Ako imamo da je :

$$T_i \neq T_{in} \qquad \text{i} \qquad T_j \neq T_{jn}$$

prelazimo na sledeći korak, i korigujemo faktore otpora.

T_{in} i T_{jn} – vrednosti ukupnog broja kretanja iz zone i i u zonu j

$$F_{in(k+1)} = \frac{T_i}{T_{in}} \qquad F_{jn(k+1)} = \frac{T_j}{T_{jn}}$$

Sa novim vrednostima faktora otpora ulazimo u sledeću iteraciju. Postupak se ponavlja dok se ne postigne:

$$T_i = T_{in} \qquad T_j = T_{jn}$$

Zadatak 2: Data je matrica razmene kretanja u postojećem stanju (Tabela 3.11). Na osnovu metoda prosečnog faktora rasta prognozirati buduću razmenu kretanja između zona. Kriterijum $0.95 < F < 1.05$, neka bude merodavan za broj iteracija.

$$T_{ij} = t_{ij} \frac{(F_i + F_j)}{2}$$

$$F_i = \frac{T_i}{t_i} \qquad F_j = \frac{T_j}{t_j}$$

Tabela 3.11 Matrica razmene kretanja u postojećem stanju

i\j	1	2	3	t_i	T_i	F_i
1	0	20	40	60	90	1.5
2	50	0	15	65	80	1.23
3	40	15	0	55	85	1.55
t_j	90	35	55	180	255	
T_j	120	55	75			
F_j	1.33	1.57	1.36			

✓ Rešenje

Prva iteracija:

$$F_{in(k+1)} = \frac{T_i}{T_{in}}, \qquad F_{jn(k+1)} = \frac{T_j}{T_{jn}}$$

$\boxed{20(1.5+1.57)/2}$

Tabela 3.12 Vrednosti dobijene posle prve iteracije

i\j	1	2	3	t_i	T_{in}	F_{in}
1	0	31	57	88	90	1.02
2	64	0	19	83	80	0.96
3	58	23	0	81	85	1.05
t_j	122	54	76	252	255	
T_{jn}	120	55	75			
F_{jn}	0.98	1.02	0.99			

Imamo vrednost faktora porasta od 1.05, tako da ulazimo u novu iteraciju

Druga iteracija:

Tabela 3.13 Vrednosti dobijene posle druge iteracije

i\j	1	2	3	t_i	T_{in}	F_{in}
1	0	31	57	88	90	1.02
2	62	0	18	80	80	1
3	59	24	0	83	85	1.02
t_j	121	55	75	251	255	
T_{jn}	120	55	75			
F_{jn}	0.99	1	1			

$$\mathbf{0.95 < F < 1.05}$$

3.2.1.3 Fratar model

Osnovne dve pretpostavke na kojima se bazira ovaj model su:

Domaći zadatak

- Rešiti po izboru jedan od zadataka 3,4,5,6 iz Poglavlja 3 iz Zbirke zadataka iz Planiranja saobraćaja i poslati mi rešenje na mejl. Zadatke podelite po dogovoru, nemojte svi raditi isti zadatak.