



IV. RADIONICA - INFOKUP NATJECANJE

ALGORITMI ZA SREDNJE ŠKOLE

Tomo Sjekavica, Informatički klub FUTURA
Dubrovnik, 18. siječanj 2014.



Zaštićeno licencom <http://creativecommons.org/licenses/by-nc-sa/3.0/hr/>



Creative Commons



☐ **slobodno smijete:**

- **dijeliti** — umnožavati, distribuirati i javnosti priopćavati djelo
- **remiksirati** — prerađivati djelo

☐ **pod slijedećim uvjetima:**

- **imenovanje.** Morate priznati i označiti autorstvo djela na način kako je specificirao autor ili davatelj licence (ali ne način koji bi sugerirao da Vi ili Vaše korištenje njegova djela imate njegovu izravnu podršku).
- **nekomercijalno.** Ovo djelo ne smijete koristiti u komercijalne svrhe.
- **dijeli pod istim uvjetima.** Ako ovo djelo izmijenite, preoblikujete ili stvarate koristeći ga, preradu možete distribuirati samo pod licencom koja je ista ili slična ovoj.

U slučaju daljnjeg korištenja ili distribuiranja morate drugima jasno dati do znanja licencne uvjete ovog djela. Najbolji način da to učinite je linkom na ovu internetsku stranicu.

Od svakog od gornjih uvjeta moguće je odstupiti, ako dobijete dopuštenje nositelja autorskog prava.

Ništa u ovoj licenci ne narušava ili ograničava autorova moralna prava.

Tekst licence preuzet je s <http://creativecommons.org/>.

Maja 1/3



Županijsko
natjecanje
HSIN – 2007

- Maja je bila tužna jer se nitko nije želio igrati s njom, te je naposljetku odlučila igrati pojednostavljenu verziju popularne igre „Čovječe, ne ljuti se“ sama protiv sebe. Cilj njene igre je što prije doći s prvog na posljednjeg polja ploče koja se sastoji od 100 polja u nizu. Jedinu prepreku do sretnog završetka igre joj predstavlja određeni broj rupa koje se nalaze na ploči i u koje ne smije stati svojom figuricom. Da joj igra ne bi bila preteška Maja je odlučila da neće bacati kockicu već će svaki put kockicu okrenuti na broj koji njoj odgovara.

Maja 2/3



Županijsko
natjecanje
HSIN – 2007

- Tvoj zadatak je pomoći Maji i odrediti koliko najmanje poteza ona mora odigrati da dođe do kraja ploče, tako da ni u jednom trenutku ne stane figuricom u rupu na ploči.
- Napomene:
 - Prvo i posljednje polje na ploči neće biti rupe.
 - Maja igra standardnom igraćom kockicom na kojoj se nalaze brojevi od 1 do 6.
 - Maja će uvijek moći završiti igru, tj. nikada neće biti šest ili više rupa zaredom.

1	2	3	4	5	6	7	8	9	10
20	19	18	17	16	15	14	13	12	11
21	22	23	24	25	26	27	28	29	30
40	39	38	37	36	35	34	33	32	31
41	42	43	44	45	46	47	48	49	50
60	59	58	57	56	55	54	53	52	51
61	62	63	64	65	66	67	68	69	70
80	79	78	77	76	75	74	73	72	71
81	82	83	84	85	86	87	88	89	90
100	99	98	97	96	95	94	93	92	91

Maja 3/3



Županijsko
natjecanje
HSIN – 2007



□ Ulazni podaci:

- cijeli broj N ($0 \leq N \leq 15$), broj rupa na ploči,
- N različitih prirodnih brojeva R ($2 \leq R \leq 99$), koji označavaju da je polje pod rednim brojem R rupa u koju se ne smije stati figuricom.

□ Izlazni podatak:

- prirodan broj X – najmanji broj poteza koje Maja treba odigrati da dođe s prvog na posljednje polje.

Primjeri testnih podataka

ULAZ	ULAZ	ULAZ	ULAZ	ULAZ	ULAZ
0	8	3	4	5	10
	3	11	11	63	49
	4	12	12	64	48
	5	7	10	65	47
	7		7	66	46
	8			67	45
	10				42
	11				41
	12				40
					39
					38
IZLAZ	IZLAZ	IZLAZ	IZLAZ	IZLAZ	IZLAZ
17	18	17	18	18	18

Vrijeme





Maja – rješenje 1/2

```
#include<stdio.h>
```

```
int main(){  
    int i, n, r, polje[100]={0};  
    int pozicija=1, rezultat=0, pom;
```

```
    scanf("%d", &n);  
    for(i=0; i<n; i++){  
        scanf("%d", &r);  
        polje[r] = 1;  
    }
```

1	2	3	4	5	6	7	8	9	10
20	19	18	17	16	15	14	13	12	11
21	22	23	24	25	26	27	28	29	30
40	39	38	37	36	35	34	33	32	31
41	42	43	44	45	46	47	48	49	50
60	59	58	57	56	55	54	53	52	51
61	62	63	64	65	66	67	68	69	70
80	79	78	77	76	75	74	73	72	71
81	82	83	84	85	86	87	88	89	90
100	99	98	97	96	95	94	93	92	91



maja.c



Maja – rješenje 2/2

```
do{
    for(i=1; i<=6; i++){
        if(polje[pozicija+i] == 0)
            pom = pozicija + i;
    }
    pozicija = pom;
    rezultat++;
}while(pozicija < 94);

printf("%d\n", rezultat + 1);
system("pause");
return 0;
}
```

```
0
17
Press any key to continue . . .
```

```
8
3
4
5
7
8
10
11
12
18
Press any key to continue . . .
```

```
3
11
12
7
17
Press any key to continue . . .
```



Mirko 1/3



Županijsko
natjecanje
HSIN – 2008

- U popularnu pizzeriju u centra grada vole dolaziti i stari i mladi. Šef kuhinje je primijetio jednu čudnu pravilnost. Šef kuhinje želi znati kada će mu **doći prvi gost**, te kada će mu **zadnji gost otići**. Također želi znati koliko će mu **gostiju u jednom trenutku najviše biti u restoranu** da bi znao koliko stolova mora postaviti. Naravno on je svjestan da sam ne može odrediti trenutke kada će mu gosti doći te koliko će dugo oni ostati u restoranu, stoga je odlučio nazvati Mirka Miočića za kojeg svi znamo da odgovor na svako pitanje ima u malom prstu.

Mirko 2/3



Županijsko
natjecanje
HSIN – 2008

- Mirko mu je napisao za svakog gosta vrijeme od početka dana u minutama kad će mu određeni gost doći, ali nažalost ne onim redom kojim gosti dolaze. Također mu je rekao da će prvi gost na njegovoj listi ostati u restoranu pola sata, drugi cijeli sat, treći gost 1,5 sati itd.
- **Napomene:**
 - smatramo da čovjek dolazi **na početku minute** za koje je Mirko rekao da će doći, a tek nakon što svi novi gosti dođu **tek onda odlaze oni koji u toj minuti trebaju otići**.
 - zadnji gost će otići do kraja dana, tj. minuta odlaska će biti manja od 1440 ($24 * 60 = 1440$ min).

Mirko 3/3



Županijsko
natjecanje
HSIN – 2008



□ Ulazni podaci:

- prirodan broj **N** ($1 \leq N \leq 10$) – broj gostiju koji će doći u restoran;
- **N** prirodnih brojeva **A1, A2, A3, ... , An** ($1 \leq A_i \leq 1440$), svaki u svom redu – minuta dolaska gosta, računajući od početka dana.

□ Izlazni podatak:

- vrijeme u minutama, računajući od početka dana, kada je došao prvi gost u pizzeriju;
- vrijeme u minutama, računajući od početka dana, kada je zadnji gost izašao iz pizzerije;
- najveći broj ljudi u jednom trenutku koji se nalazi u pizzeriji.

Primjeri testnih podataka

ULAZ	ULAZ	ULAZ	ULAZ	ULAZ	ULAZ
3	5	5	7	8	10
100	200	200	123	500	100
110	220	300	321	450	110
200	230	150	31	515	120
	600	140	23	520	130
	500	100	56	510	140
			47	515	150
			98	700	160
				100	150
					140
					130
IZLAZ	IZLAZ	IZLAZ	IZLAZ	IZLAZ	IZLAZ
100	200	100	23	100	100
290	720	360	381	910	430
2	3	4	5	5	9

Vrijeme





Mirko – rješenje 1/2

```
#include<stdio.h>

int main(){
    int i, n, a[10];
    int prvi=1440, zadnji=0;
    int dan[1442]={0}, max=0, ima=0;
    scanf("%d", &n);
    for(i=1; i<=n; i++){
        scanf("%d", &a[i]);
        if(a[i] < prvi)
            prvi = a[i];
        if(a[i] + i*30 > zadnji)
            zadnji = a[i] + i*30;
        dan[a[i]]++;
        dan[a[i] + i*30 +1]--;
    }
```





Mirko – rješenje 1/2

```
for(i=1; i<1442; i++){  
    ima = ima + dan[i];  
    if(ima > max)  
        max = ima;  
}
```

```
printf("\n%d\n", prvi);  
printf("%d\n", zadnji);  
printf("%d\n", max);
```

```
system("pause");  
return 0;
```

```
}
```

```
3  
100  
110  
200  
  
100  
290  
2  
Press any key to continue . . .
```

```
5  
200  
220  
230  
600  
500  
  
200  
720  
3  
Press any key to continue . . .
```

```
10  
100  
110  
120  
130  
140  
150  
160  
150  
140  
130  
  
100  
430  
9  
Press any key to continue . . .
```



Stari 1/2



Školsko
natjecanje
Algoritmi - 2012

- Marini je dosadno. Napisala je zadaću, prošetala psa, pospremila sobu, a zbog tehničkog kvara nije bilo interneta u cijelom gradu. Zato je odlučila riješiti jedan tipičan numerički zadatak s nekog starog natjecanja koji je još davno bila pronašla. Zadatak kaže:
- „Zadano je **N** prirodnih brojeva. Koristeći zadane brojeve treba **napraviti novi N-teroznamenasti broj** tako da je njegova **krajnja lijeva** znamenka **najmanja** znamenka u prvom zadanom broju, njegova druga znamenka s lijeva je najmanja znamenka u drugom zadanom broju, itd. Postupak se ponavlja sve do znamenke jedinica novog broja koja mora biti najmanja znamenka u zadnjem zadanom broju. Napiši program koji za zadane brojeve kreira i ispisuje traženi novi broj“.

Stari 2/2



Školsko
natjecanje
Algoritmi – 2012



□ Ulazni podaci:

- prirodan broj **N** ($1 \leq N \leq 6$), broj zadanih prirodnih brojeva;
- u sljedećih **N** redaka se nalazi po jedan prirodan broj **X** ($1 \leq X \leq 9999$), u poretku od prvog do zadnjeg zadanog broja iz zadatka.

□ Izlazni podaci:

- prirodan broj koji je kreiran na način opisan u zadatku.

□ Napomena:

- u **50%** test primjera vrijedit će da je **N=3** i da su svi zadani brojevi troznamenkasti

Vrijeme



Primjeri testnih podataka

ULAZ	ULAZ	ULAZ	ULAZ	ULAZ	ULAZ
3	4	3	2	3	4
325	4327	418	8643	447	200
461	89	639	23	733	300
596	528	257		575	400
	6				473
IZLAZ	IZLAZ	IZLAZ	IZLAZ	IZLAZ	IZLAZ
215	2826	132	32	435	3



Stari – rješenje 1/2

```
#include<stdio.h>
```

```
int main(){  
    int i, n, x, min;  
    int znamenka, broj=0;  
  
    scanf("%d", &n);  
  
    for(i=0; i<n; i++){  
        scanf("%d", &x);  
        min = 9;
```





Stari – rješenje 2/2

```
while(x > 0){
    znamenka = x % 10;
    if(znamenka < min)
        min = znamenka;
    x /= 10;
}
broj = broj*10 + min;
}
printf("%d\n", broj);

system("PAUSE");
return 0;
}
```

```
3
325
461
596
215
Press any key to continue . . .
```

```
4
4327
89
528
6
2826
Press any key to continue . . .
```

```
3
418
639
257
132
Press any key to continue . . .
```

```
4
200
300
400
473
3
Press any key to continue . . .
```





Benford 1/3

- Slavko voli zemljopis, a Mirku se tijekom zajedničkog trčanja pohvalio kako je zapamtio broj stanovnika svakog naselja svih zemalja svijeta. Mirko je na to predložio izazov: odabrati će jednu zemlju, a ako Slavko uspije na papiru ispisati broj stanovnika svih naselja te zemlje, Mirko će ga nagraditi velikom Mikado čokoladom od riže. Mirko ne zna zemljopis, pa teško može provjeriti jesu li svi Slavkovi brojevi točni, ali Mirko voli matematiku! Nedavno ga je oduševio Benfordov zakon, koji u svom pojednostavljenom obliku kaže:
 - „Ako imamo puno brojeva koji na neki način dolaze iz prirode i društva (na primjer, broj stanovnika svakog naselja jedne zemlje) te ako pogledamo prve znamenke svih tih brojeva, među njima će najviše biti jedinica, zatim dvojki, a zatim trojki i tako dalje, dok će najmanje brojeva počinjati sa znamenkom 9.“

Benford 2/3



- Mirko sumnja da će ga Slavko pokušati prevariti pa mu treba vaša pomoć. Napišite program koji će provjeriti zadovoljava li niz od **N** zadanih brojeva stanovnika ovaj zakon. Točnije, vaš program treba odrediti **vrijedi li $C_1 \geq C_2 \geq C_3 \geq \dots \geq C_9$** , gdje je **$C_1$** broj zadanih brojeva koji počinju znamenkom 1, **C_2** broj zadanih brojeva koji počinju znamenkom 2 i tako dalje, te, u slučaju da to ne vrijedi, **naći najmanje znamenke K i $K+1$ koje narušavaju ovo svojstvo**, te odgovarajuće brojeve **C_K i C_{K+1}** .

Benford 3/3



Školsko
natjecanje
Algoritmi – 2013



□ Ulazni podaci:

- Prvi red ulaza sadrži prirodni broj N ($1 \leq N \leq 1000$) - broj naselja odabrane zemlje. Svaki od sljedećih N redova sadrži jedan prirodni broj manji od 100,000,000 - broj stanovnika jednog naselja odabrane zemlje.

□ Izlazni podaci:

- Ako niz zadovoljava svojstvo opisano u zadatku tada prvi i jedini red ulaza treba sadržavati broj 0. U suprotnom, potrebno je u prvi red ispisati tri prirodna broja K , C_K i C_{K+1} gdje je K najmanja znamenka takva da je C_K manje od C_{K+1} .

Primjeri testnih podataka

ULAZ	ULAZ	ULAZ
6	6	10
456	456	6
1110	1110	4
1053	1053	84
303001	303001	85
22	3000	28
2005	2005	77
		62
		72
		62
		25
IZLAZ	IZLAZ	IZLAZ
0	2 1 2	1 0 2

Vrijeme





Benford – rješenje 1/2

```
#include<stdio.h>
```

```
int main(){  
    int i, broj, n, pom = 0;  
    int polje[10] = {0};  
  
    scanf("%d", &n);  
  
    for(i=0; i<n; i++){  
        scanf("%d", &broj);  
        for(; broj>=10; broj/=10);  
        polje[broj]++;  
    }  
}
```



benford.c



Benford – rješenje 2/2

```
for(i=1; i<9 && pom==0; i++)  
    if(polje[i]<polje[i+1]){  
        printf("%d %d ", i, polje[i]);  
        printf("%d\n", polje[i+1]);  
        pom = 1;  
    }
```

```
if(pom == 0)  
    printf("0\n");
```

```
system("PAUSE");  
return 0;
```

```
}
```

```
6  
456  
1110  
1053  
303001  
22  
2005  
0  
Press any key to continue . . .
```

```
6  
456  
1110  
1053  
303001  
3000  
2005  
2 1 2  
Press any key to continue . . .
```

```
10  
6  
4  
84  
85  
28  
77  
62  
72  
62  
25  
1 0 2  
Press any key to continue . . .
```



benford.c