

RADIONICE PROGRAMIRANJA U C-U ZA UČENIKE SREDNJIH ŠKOLA STUDENI/PROSINAC 2013.

ORGANIZATOR



U SURADNJI SA



UZ POTPORU





II. RADIONICA PROGRAMIRANJA ZA SREDNJE ŠKOLE - NAPREDNI

Tomo Sjekavica, Informatički klub FUTURA
Dubrovnik, prosinac 2013.



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 sljedećim 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/>.

Realni brojevi

Hmm, kako ću
izračunati apsolutnu
vrijednost???



- Ivo voli u slobodno vrijeme nabrajati sve realne brojeve osim nule. Ana voli razne izračune s brojevima, pa je zanima je li veći zbroj apsolutnih vrijednosti pozitivnih ili negativnih brojeva koje je Ivo nabrojio.
- Pomogni Ani i napiši program koji prima broj N i niz od N realnih brojeva, te izračunava veću apsolutnu vrijednost
- Ulazni podaci: svaki broj se unosi u svom retku
 - Cijeli broj N ($1 \leq N \leq 10$), veličina niza realnih brojeva
 - N realnih brojeva
- Izlazni podaci: u dva retka
 - Realni broj – apsolutna vrijednost na 3 decimale
 - Niz znakova
 - „pozitivni”, ako je zbroj apsolutnih vrijednosti pozitivnih brojeva veći
 - „negativni”, ako je zbroj apsolutnih vrijednosti negativnih brojeva veći

Primjeri testnih podataka

ULAZ	ULAZ	ULAZ	ULAZ
5	3	4	5
-1.12	-55.78	3.5	-13.13
2	30.4	-5.6666	2.34
3.5456	12.4444	-1.234	1.9999
-4.4		5.02	0.123
1.22			-2.589
IZLAZ	IZLAZ	IZLAZ	IZLAZ
6.766	55.780	8.520	15.719
pozitivni	negativni	pozitivni	negativni

Što sve od C-a moramo znati da bismo riješili ovaj zadatak?



- Deklarirati varijable cijelih i realnih brojeva
- Deklarirati jednodimenzionalno polje
- U petlji unositi brojeve u polje
- Ispitivati logičke uvjete
- Računati apsolutnu vrijednost broja (uz pomoć funkcije iz `math.h` zaglavne datoteke)
`fabs(x)`
- Zbrajati brojeve
- Ispisati podatak na standardni izlaz

Realni brojevi

Vrijeme



- Ivo voli u slobodno vrijeme nabrajati sve realne brojeve osim nule. Ana voli razne izračune s brojevima, pa je zanima je li veći zbroj apsolutnih vrijednosti pozitivnih ili negativnih brojeva koje je Ivo nabrojio.
- Pomogni Ani i napiši program koji prima broj N i niz od N realnih brojeva, te izračunava veću apsolutnu vrijednost
- Ulazni podaci: svaki broj se unosi u svom retku
 - Cijeli broj N ($1 \leq N \leq 10$), veličina niza realnih brojeva
 - N realnih brojeva
- Izlazni podaci: u dva retka
 - Realni broj – apsolutna vrijednost na 3 decimale
 - Niz znakova
 - „pozitivni”, ako je zbroj apsolutnih vrijednosti pozitivnih brojeva veći
 - „negativni”, ako je zbroj apsolutnih vrijednosti negativnih brojeva veći

Primjeri testnih podataka

ULAZ	ULAZ	ULAZ	ULAZ
5	3	4	5
-1.12	-55.78	3.5	-13.13
2	30.4	-5.6666	2.34
3.5456	12.4444	-1.234	1.9999
-4.4		5.02	0.123
1.22			-2.589
IZLAZ	IZLAZ	IZLAZ	IZLAZ
6.766	55.780	8.520	15.719
pozitivni	negativni	pozitivni	negativni

VAŽNO UPOZORENJE

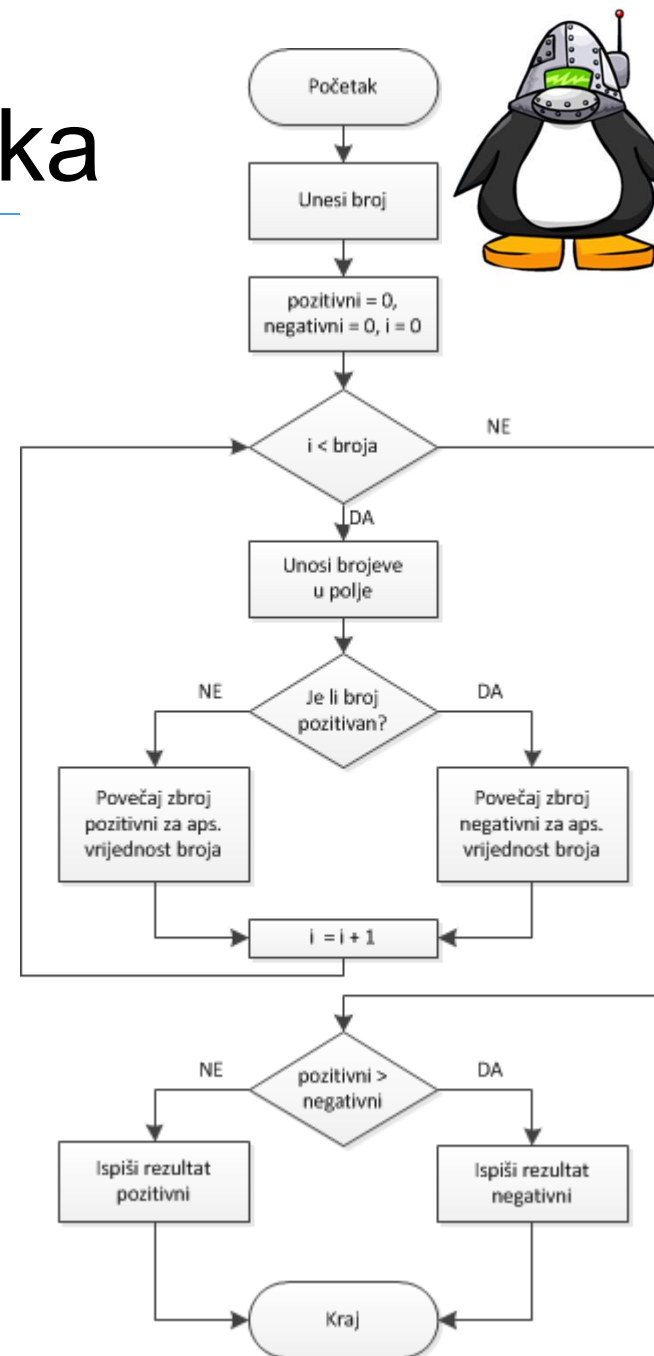


- Sada STANI i nemoj ići dalje.**
- Razmisli o zadatku** i kako bi ga riješio!
- Ako misliš da si došao do rješenja ili nemaš pojma kako bi ga riješio 😊 prijeđi na sljedeći slajd i pogledaj/provjeri.

Primjer rješavanje zadatka

□ Koraci algoritma rješenja:

- Pročitaj cijeli broj sa standardnog ulaza
- U petlji unosi realne brojeve u jednodimenzionalno polje
- Provjeri je li broj pozitivan ili negativan
 - Ako je broj pozitivan uvećaj zbroj apsolutnih vrijednosti pozitivnih brojeva
 - Ako je broj negativan uvećaj zbroj apsolutnih vrijednosti negativnih brojeva
- Ispiši na standardni izlaz veću apsolutnu vrijednost i tekst





Realni brojevi - rješenje

```
#include<stdio.h>
#include<math.h>
#define MAX 10
int main(){
    int i, n;
    float poz=0, neg=0;
    float niz[MAX] = {0};

    scanf("%d", &n);
    for(i=0; i<n; i++){
        scanf("%f", &niz[i]);    }
        if(niz[i] > 0){
            poz += fabs(niz[i]);
        }else{
            neg += fabs(niz[i]);
        }
    }
}
```



```
if(poz>neg){
    printf("%.3f\n", poz);
    printf("pozitivni\n");
}else{
    printf("%.3f\n", neg);
    printf("negativni\n");
}

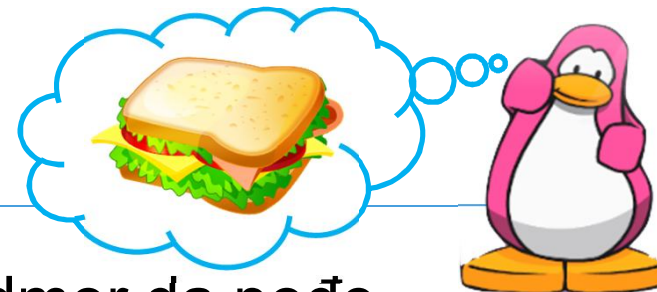
system("PAUSE");
return 0;
```



```
3
-55.78
30.4
12.4444
55.780
negativni
Press any key to continue . . .
```



Sendvič 1/2



- Ana je u školi i ne može dočekati odmor da pođe doma po sendvič. Ana živi u blizini škole, a vrijeme potrebno da pođe doma, pojede sendvič i vrati se u školu je točno koliko joj traje mali odmor. Ana gleda na sat i zanima je koliko će još trebati čekati da može poći po sendvič, a da se vrati do početka sljedećeg školskog sata.
- Ani prvi školski sat počinje u 09:00 (da se stigne naspavati 😊). Svaki školski sat traje **N** minuta, a u jednom danu Ana ima 7 školskih sati. Između svaka dva školska sata je mali odmor koji traje **M** minuta. Jedino je nakon trećeg sata veliki odmor koji traje **V** minuta. Ana može poći po sendvič preko malog i preko velikog odmora, ali i nakon nastave.
- Ani problem predstavlja često mijenjanje trajanja školskog sata, te trajanja velikog i malog odmora.

Sendvič 2/2



- Pomogni Ani i napiši program koji će joj reći koliko treba čekati prije nego može poći po sendvič.
- Ulazni podaci:
 - Tri cijela broja: školski sat, mali i veliki odmor
 - N – trajanje školskog sata u minutama ($1 \leq N \leq 60$)
 - M – trajanje malog odmora u minutama ($1 \leq N \leq 15$)
 - V – trajanje velikog odmora u minutama ($1 \leq N \leq 30$)
 - Dva cijela broja: vrijeme koje je Ana pročitala sa sata
 - HH – sati ($0 \leq HH \leq 23$)
 - MM – minute ($0 \leq MM \leq 59$)
- Izlazni podaci:
 - Niz znakova
 - “Mozes poci odmah”, ako Ana može odmah krenuti po sendvič
 - “Mozes poci za X minuta”, ako Ana treba čekati X minuta da krene po sendvič

Vrijeme



Primjeri testnih podataka

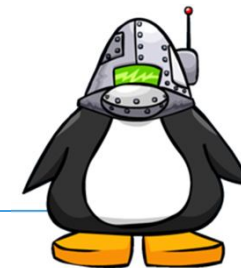
ULAZ	ULAZ	ULAZ	ULAZ
45	25	30	35
5	6	5	4
15	12	10	12
12	9	10	11
17	25	15	00
IZLAZ	IZLAZ	IZLAZ	IZLAZ
za 8	odmah	za 25	odmah

VAŽNO UPOZORENJE



- Sada STANI i nemoj ići dalje.**
- Razmisli o zadatku** i kako bi ga riješio!
- Ako misliš da si došao do rješenja ili nemaš pojma kako bi ga riješio 😊 prijeđi na sljedeći slajd i pogledaj/provjeri.

Primjer rješavanje zadatka



- Pročitaj cijele brojeve sa standardnog ulaza
- Odredi razliku od očitanoog vremena i početka nastave
- U petlji pronađi kada Ana može krenuti po sendvič
 - Provjeri je li razlika vremena manja od trajanja sata i ispiši poruku
 - Provjeri je li razlika vremena jednaka trajanju sata i ispiši poruku
 - Provjeri je li treći sat
 - Provjeri je li razlika vremena manja od trajanja sata i velikog odmora
 - Provjeri je li razlika vremena manja ili jednaka trajanju sata i velikog odmora umanjeno za mali odmor i ispiši odgovarajuću poruku ako jest i ako nije
 - Umanji razliku vremena za trajanje sata i trajanje velikog odmora
 - Ako nije treći sat
 - Provjeri je li razlika vremena manja od trajanja sata i malog odmora i ispiši odgovarajuću poruku
 - Umanji razliku vremena za trajanje sata i trajanje malog odmora



Sendvič – rješenje 1/3

```
#include<stdio.h>
int main(){
    int n, m, v, hh, mm, pom, i;
    scanf("%d %d %d", &n, &m, &v);
    scanf("%d %d", &hh, &mm);

    pom = hh*60 + mm - 9*60;
    for(i=0; i<7; i++){
        if(pom < n){
            printf("Mozes poci za %d minuta!\n", n-pom);
            break;
        }
        if(pom == n){
            printf("Mozes poci odmah!\n");
            break;
        }
    }
}
```



sendvic.c



Sendvič – rješenje 2/3

```
if(i == 2){
    if(pom < n + v){
        if(pom <= n + v - m){
            printf("Mozes poci odmah!\n");
            break;
        }else{
            printf("Mozes poci za ");
            printf("%d minuta!\n", n+v-pom+n);
            break;
        }
    }
    pom -= n + v;
}
```





Sendvič – rješenje 3/3

```
else{
    if(pom < n + m){
        printf("Mozes poci za ");
        printf("%d minuta!\n", n+m-pom+n);
        break;
    }
    pom -= n + m;
}
}

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

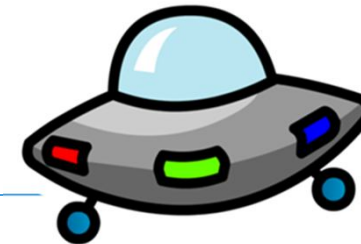
```
45
5
15
12
17
Mozes poci za 8 minuta!
Press any key to continue . . . _
```

```
25
6
12
9
25
Mozes poci odmah!
Press any key to continue . . . _
```



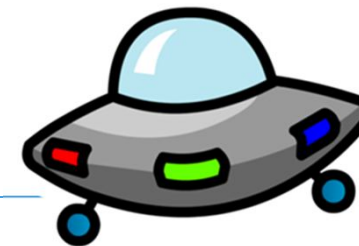
sendvic.c

Međuplanetarni turizam 1/3



- Dobro je poznata činjenica da Zemlju svakodnevno posjećuju NLO-i i na turistička razgledavanja svemira odvođe grupe putnika.
- Nažalost kapaciteti su mali pa pri svakom dolasku mogu povesti samo jednu grupu. Njihova turistička agencija je smislila shemu pomoću koje grupe unaprijed znaju hoće li otići na određeno putovanje.
- Shema omogućuje da se na temelju naziva putovanja i naziva grupe odredi hoće li određena grupa otputovati pri tom posjetu.
- Detalji sheme su opisani u nastavku, a vaš zadatak je da napišete program koji će za zadani naziv putovanja i grupe provjeriti hoće li ta grupa ići na to putovanje.

Međuplanetarni turizam 2/3



- Nazivi putovanja i grupe sastavljeni su od velikih slova međunarodne abecede (A-Z) i znaka praznine.
 - Maksimalna duljina oba naziva je 30 znakova.
- Svaki naziv se pretvara u broj koji ga predstavlja na sljedeći način:
 - Svaki znak naziva se pretvara u broj na sljedeći način: praznina=1, A=2, B=3, ... , Z=27
 - Dobivene vrijednosti se zbrajaju
 - Primjerice ako je naziv grupe „GIMNAZIJALCI S PLOCA” vrijednost tog naziva je 200

G	I	M	N	A	Z	I	J	A	L	C	I	S	P	L	O	C	A
8	10	14	15	2	27	10	11	2	13	4	10	1	17	13	16	4	2

- Ako je vrijednost naziva grupe modulo 7 jednaka vrijednosti naziva putovanja po istom modulu, grupa se može spremi za putovanje

Međuplanetarni turizam 3/3



- Napišite program koji će za zadane nazive putovanja i grupe odrediti hoće li grupa ići na zadano putovanje.

ULAZNI PODACI

- U 1. redu nalazi se naziv putovanja
- U 2. redu nalazi se naziv grupe

IZLAZNI PODACI

- Niz znakova.
 - Ako grupa ide na putovanje ispišite „IDU”
 - Ako grupa ne ide na putovanje ispišite „NE IDU”

PRIMJERI TESTNIH PODATAKA

ULAZ	ULAZ	ULAZ
MJESEC	MARS	RAZGLEDAVANJE SATURNA
PROGRAMERI	SREDNJOSKOLCI	GIMNAZIJALCI S PLOCA
IZLAZ	IZLAZ	IZLAZ
NE IDU	IDU	NE IDU

Vrijeme



Što sve od C-a moramo znati da bismo riješili ovaj zadatak?



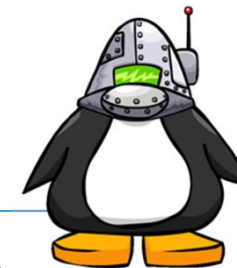
- Deklarirati varijable (cjelobrojnu i niz znakova)
- Unijeti niz znakova sa standardnog ulaza
 - PAZI! Niz znakova sadrži prazninu
- U petlji pristupati znakovima niza
 - PAZI! Kako znaš da si došao do kraja niza?
- Ispitivati logički uvjet
- Zbrajati brojeve
 - PAZI! Kako pretvoriti znak u određeni broj?
- Koristiti modulo operator
 - $z = x \% y;$**
- Ispisati podatak na standardni izlaz

VAŽNO UPOZORENJE

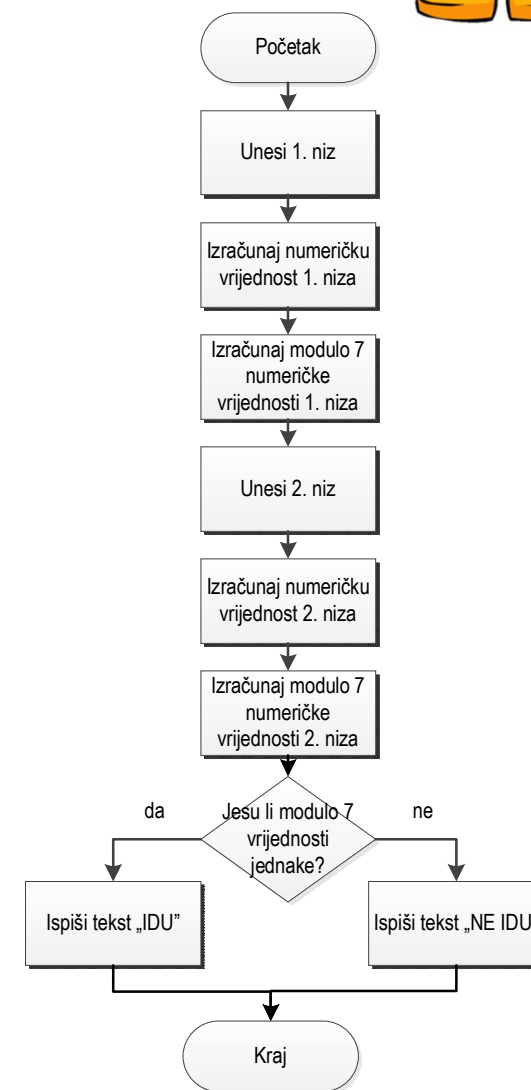


- Sada STANI i nemoj ići dalje.**
- Razmisli o zadatku** i kako bi ga riješio!
- Ako misliš da si došao do rješenja ili nemaš pojma kako bi ga riješio 😊 prijeđi na sljedeći slajd i pogledaj/provjeri.

Primjer rješavanja zadatka



- Koraci algoritma rješenja:
 - Dohvati niz znakova zadane duljine sa standardnog ulaza
 - Poveži svaki znak dohvaćenog niza s cijelim brojevima prema zadanom algoritmu
 - Zbroji brojčane vrijednosti pojedinačnih znakova
 - Izračunaj modulo 7 rezultat dobivenog broja
 - Ponovi prethodna 4 koraka za drugi niz znakova
 - Ako je izračunata modulo 7 vrijednost oba niza jednaka, ispiši na standardni izlaz „IDU”, inače ispiši tekst „NE IDU”





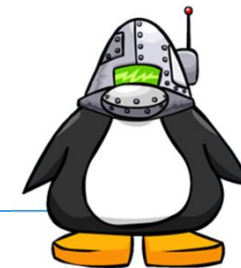
Prva verzija (*nepotpuna!*)

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int i, zbroj=0, rez=0; // deklaracija
    char polje[30+1]; // deklaracija polja
    scanf("%[^\\n]", polje); // unos sa stand. ulaza
    for (i=0;polje!="\\0"){ // polje
        zbroj = zbroj + polje[i]; // zbroj num. vr.
    }
    rez = zbroj % 7; // modulo 7
    printf("%d\\n", rez); // ispis rezultata
    return 0;
}
```

```
GIMNAZIJALCI S PLOCA
3
Press any key to continue . . .
```



Nedostaci prve verzije



- Kako ćemo slovo pretvoriti u zadani broj?
- Svako slovo ima svoj brojčani kod

- ASCII kodovi
- A=65, B=66,...

- **A** treba pretvoriti u **2**, **B** u **3**,...

x = kod - 63; ili

x = kod % 63;

- Što s razmakom?

- Izravno pridruživanje **if (kod==' ') x = 1**

Decimal	Octal	Hex	Character	Decimal	Octal	Hex	Character
0	0	00	NUL	64	100	40	@
1	1	01	SOH	65	101	41	A
2	2	02	STX	66	102	42	B
3	3	03	ETX	67	103	43	C
4	4	04	EOT	68	104	44	D
5	5	05	ENQ	69	105	45	E
6	6	06	ACK	70	106	46	F
7	7	07	BEL	71	107	47	G
8	10	08	BS	72	110	48	H
9	11	09	HT	73	111	49	I



Druga verzija

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int i, zbroj=0, rez=0; // deklaracija
    char polje[30+1]; // deklaracija polja
    scanf("%[^\n]", polje); // unos sa stand. ulaza
    for (i=0;polje!="\0"){ // polje
        if (polje[i]==' ')
            zbroj = zbroj + 1;
        else
            zbroj = zbroj + polje[i]-63;
    }
    rez = zbroj % 7; // modulo 7
    printf("%d\n", rez); // ispis rezultata
    return 0;
}
```

```
GIMNAZIJALCI S PLOCA
4
Press any key to continue . . .
```





Kompletan program – 1. verzija

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int i, zbroj1=0, zbroj2=0; // deklaracija
    char grupa[30+1], putovanje[30+1]; // deklaracija polja
    scanf("%s", putovanje); // unos naziva putovanja
    for (i=0;putovanje[i]!='\0';i++){ // polje
        if (putovanje[i]==' ')
            zbroj1 = zbroj1 + 1;
        else
            zbroj1 = zbroj1 + putovanje[i]-63;
    }
    scanf("%s", grupa); // unos naziva grupe
    for (i=0;grupa[i]!='\0';i++){
        if (grupa[i]==' ')
            zbroj2 = zbroj2 + 1;
        else
            zbroj2 = zbroj2 + grupa[i]-63;
    }
    if ((zbroj1%7)==(zbroj2%7))
        printf("IDU\n"); // ispis rezultata
    else
        printf("NE IDU\n"); // ispis rezultata
    system("PAUSE");
    return 0;
}
```

```
RAZGLEDAVANJE SATURNA
GIMNAZIJALCI S PLOCA
NE IDU
Press any key to continue . . .
```

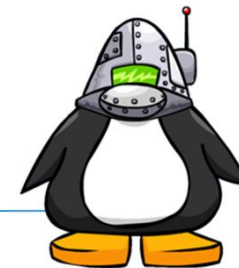
Unos naziva putovanja

Unos naziva grupe

Obrada i ispis rezultata



Moguće dorade



- Program korektno odrađuje zadatak
- Sva 3 zadana testna slučaja su ispravno obrađena
- Može li se program poboljšati?
- Uočite ponavljajuće dijelove koda koji se mogu izdvojiti u posebnu funkciju
 - Obrada naziva putovanja i grupe je identična
 - Može se izdvojiti u posebnu funkciju koja će se pozivati dva puta – jednom za naziv putovanja, a drugi put za naziv grupe



Kompletan program – 2. verzija

Funkcija koja obrađuje nazive

```
#include <stdio.h>
#include <stdlib.h>
int obrada(char naziv[]){
    int i, zbroj=0;
    for (i=0;naziv[i]!='\0';i++){
        if (naziv[i]==' ')
            zbroj = zbroj + 1;
        else
            zbroj = zbroj + naziv[i] - 63;
    }
    return zbroj % 7;
}
...
```





Kompletan program – 2. verzija

Glavni dio programa

```
int main(){
    char grupa[30+1], putovanje[30+1];
    scanf(" %[^\\n]", putovanje);
    scanf(" %[^\\n]", grupa);
    if (obrada(putovanje)==obrada(grupa)) // poziv funkcije
        printf("IDU\\n");
    else
        printf("NE IDU\\n"); // ispis rezultata
    system("PAUSE");
    return 0;
}
```

```
MJESEC
PROGRAMERI
NE IDU

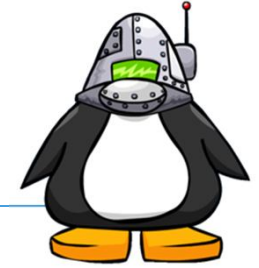
MARS
SREDNJOSKOLCI
IDU

RAZGLEDAVANJE SATURNA
GIMNAZIJALCI S PLOCA
NE IDU
```



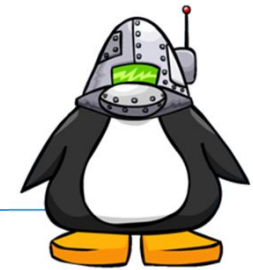
verzija1d.c

Dokle ići s doradama programa?



- Najvažnije je da program bude korektan (djelotvoran)
 - **Da uz zadane ulaze daje očekivane izlaze**
 - Da troši "podnošljivu" količinu resursa (vrijeme izvođenja, memorija, disk,...)
- U tom slučaju **nemojte doradivati** program – sigurno ćete napraviti neku grešku
- Ako program troši previše resursa (primjerice predugo vrijeme izvođenja) – treba ga optimizirati

Za one koji žele znati više



- Kako brzo istestirati više testnih primjera?
- Napravite tekstualnu datoteku s testnim ulazima
- Obradu u programu staviti unutar petlje koja se vrti onoliko puta koliko ima testnih slučajeva
- U komandnoj liniji napravite preusmjeravanje standardnog ulaza iz datoteke

primjer1.exe <ulaz.txt

- Možete napraviti i preusmjeravanje standardnog izlaza u datoteku

primjer1.exe <ulaz.txt >izlaz.txt



Testiranje - dorada u programu

Glavni dio programa

```
int main(){
    int i;
    char grupa[30+1], putovanje[30+1];
    for (i=0;i<3;i++){
        scanf("%[^\n]", putovanje);
        scanf("%[^\n]", grupa);
        if (obrada(putovanje)==obrada(grupa))
            printf("IDU\n");
        else
            printf("NE IDU\n"); // ispis rezultata
    }
    system("P>Primjer1e.exe <ulaz.txt");
    return 0;
}
```

Za svaki testni primjer, program se treba jednom izvesti

Testni primjeri: ulaz.txt

```
MJESEC
PROGRAMERI

MARS
SREDNJOSKOLCI

RAZGLEDAVANJE SATURNA
GIMNAZIJALCI S PLOCA
```

```
P>Primjer1e.exe <ulaz.txt
NE IDU

IDU

NE IDU

Press any key to continue . . . .
```



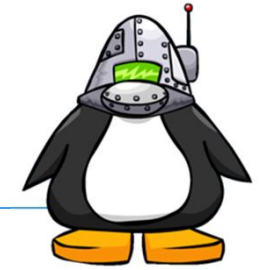
verzija1e.c

Umjesto zaključka



<http://www.infokup.hr/>

O Infokup natjecanju 1/4



- 3 razine natjecanja

Školska razina

21. siječnja 2014.

prijave od 15. prosinca...

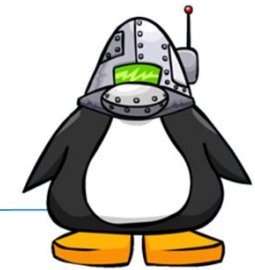
Županijska razina

21. veljače 2014.

Državna razina

...okršaj najboljih od 17. do 20. ožujka 2014.

O Infokup natjecanju 2/4



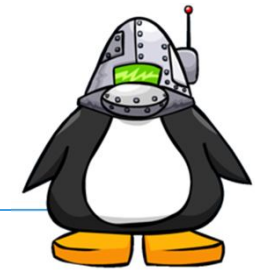
- Vrste natjecanja
 - Osnove informatike (teorija)
 - Algoritmi (to je ono za što se ovdje spremamo)
 - Razvoj softvera
- Zainteresirani za natjecanje **neka se jave svojim nastavnicima informatike do 15. prosinca**
- Školska razina natjecanja je **21. siječnja 2014**

O Infokup natjecanju 3/4



- Formalni mentor učenicima na natjecanju mora biti nastavnik iz iste škole
 - Trebate se svojim nastavnicima informatike javiti što prije jer se škole koje imaju kandidate za natjecanje moraju prijaviti na natjecanje do 15. prosinca
 - Ako još niste sigurni – prijavite se (uvijek stignete prespavati ili odustati 😊)
 - Znači šifra je "**Infokup – Algoritmi**"
- Web stranica natjecanja <http://www.infokup.hr/>
 - Detaljne informacije, propozicije natjecanja i prijava
 - Zadaci, rješenja i testni primjeri s prethodnih natjecanja

○ Infokup natjecanju 4/4



- Za natjecanje u kategoriji razvoj softvera je za ovu godinu vjerojatno prekasno...
 - ... osim ako već nemate gotovo isprogramirano rješenje!
 - Za iduću godinu (2014/2015. ima dovoljno vremena za kvalitetnu pripremu)
- Ako je netko zainteresiran za sudjelovanje u ovoj vrsti natjecanja neka nam se javi osobno na kraju radionice (ili naknadno e-mailom (unidu.prog@gmail.com) ili putem *Facebooka*)
 - Kod ove vrste natjecanja imate **potpunu slobodu** izbora aplikacije koju želite napraviti te programskog jezika i razvojnih alata koje želite koristiti