

Sastavni dio kompleta zadataka su ove upute te uvodna stranica na kojoj se nalaze mnogi bitni podaci o zadacima i molimo vas da i jedno i drugo vrlo pažljivo pročitate. Na ostalim stranicama se nalaze tri zadatka. Prilikom rješavanja zadataka preporučuje se korištenje olovke i papira za skiciranje i razradu algoritma.

PROVEDBA ŠKOLSKOG NATJECANJA

Prikupljanje i evaluacija rješenja, objavljivanje liste rezultata te rješavanje eventualnih žalbi je dužnost članova školskog povjerenstva te ste dužni slijediti njihove upute prije, tijekom te nakon završetka natjecanja. Nakon završetka natjecanja, članovi školskih povjerenstava će evaluirati vaša rješenja koristeći unaprijed pripremljene test podatke. Za svaki zadatak trebate predati izvorni kod rješenja te odgovarajuću izvršnu (exe) datoteku. Radi lakše i brže evaluacije, imena datoteka moraju odgovarati onima na uvodnoj stranici kompleta zadataka.

ULAZNI I IZLAZNI PODACI

Kod svakog pojedinog zadatka obratite pozornost na poglavlja **ULAZNI PODACI** i **IZLAZNI PODACI**. Tu su definirana pravila vezana za format ulaznih i izlaznih podataka koji mora biti poštovan kako bi vaša rješenja bila točno evaluirana. Ulaz i izlaz treba se odvijati preko standardnog ulaza i standardnog izlaza. Vaš program sa standardnog ulaza mora očekivati samo ulazne podatke, a na standardni izlaz mora ispisivati samo izlazne podatke. Vaši programi ne smiju pristupati nikakvim datotekama niti ih kreirati. Za ilustraciju i bolje razumijevanje pogledajte poglavlje *primjer pravilno napisanog programa*.

Prilikom rješavanja nekog zadatka i testiranja njegovog rješenja preporučuje se korištenje **operatora redirekcije ulaza** kako ne biste više puta nepotrebno unosili podatke preko tipkovnice. Npr., od ulaznih podataka za neki od test primjera iz teksta zadatka možete napraviti tekstualnu datoteku i testirati vaš program tako da ga pokrećete na sljedeći način (pretpostavimo da se zadatak zove *janje*):

```
janje < janje.txt
```

Znak < je operator redirekcije ulaza i sve što se nalazi u datoteci *janje.txt* bit će proslijeđeno vašem programu na isti način kao da je bilo uneseno preko tipkovnice.

PRIMJERI PRAVILNO NAPISANIH PROGRAMA

Zadatak: Napišite program koji će zbrojiti i oduzeti dva cijela broja.

Ulaz: U prvom retku se nalaze dva cijela broja A i B, međusobno odvojena jednim razmakom.

Izlaz: U prvi redak ispišite zbroj, a u drugi redak razliku brojeva A i B.

PASCAL

```
program
p(input,output);
var
  a,b : integer;
begin
  readln(a,b);
  writeln(a+b);
  writeln(a-b);
end.
```

C

```
#include <stdio.h>
int main(void)
{
  int a,b;
  scanf("%d%d",&a,&b);
  printf("%d\n",a+b);
  printf("%d\n",a-b);
  return 0;
}
```

C++

```
#include <iostream>
using namespace std;
int main(void)
{
  int a,b;
  cin >> a >> b;
  cout << a+b << endl;
  cout << a-b << endl;
  return 0;
}
```

PODRŠKA ZA PROGRAMSKI JEZIK PYTHON

Od ove godine zadatke možete rješavati koristeći programski jezik Python, a podržane su i verzija 2 i verzija 3 jezika. Imajte na umu da je podrška za ovaj programski jezik eksperimentalna te nije garantirano da će biti moguće sve zadatke dovoljno efikasno riješiti u Python-u.

Za zadatke riješene u Python-u potrebno je predati samo izvorni kod. Molimo da prva linija u kodu identificira verziju Python-a koju treba koristiti kao u primjerima dolje.

Za testiranje iz komandne linije možete također koristiti operator redirekcije ulaza, međutim potrebno je eksplicitno pozvati odgovarajući prevoditelj. Na primjer:

```
C:\Python27\python janje.py < janje.txt
```

U sljedećoj tablici dani su primjeri pravilno napisanih programa u Python'u za isti zadatak kao gore.

PYTHON2

```
#!/usr/bin/python2
a, b = map(int, raw_input().split())
print a+b
print a-b
```

PYTHON3

```
#!/usr/bin/python3
a, b = map(int, input().split())
print(a+b)
print(a-b)
```

Želimo vam puno znanja, koncentracije i sreće na ovom i daljnjim natjecanjima.