**Olimpijada znanja** **2025.**

**Takmičenje iz Programiranja za o**snovnu** školu**

**16. 5. 2024.**

**Uputstva takmičarima**

Ovo takmičenje sastoji se od rješavanja **3 problemska zadatka** u vremenu od **2 sata** (180 minuta). Zadatke je potrebno rješavati u jednom od sljedećih programskih jezika: C, C++ ili Java. Takmičari u C-u i C++-u moraju koristiti programski alat **CodeBlocks**. Za programski jezik Java predviđena je upotreba platforme **Eclipse**. Dozvoljeno je koristiti editor po izboru i pomoću navedenih alata prevoditi izvorni kod u izvršnu datoteku.

Tokom takmičenja **ne smijete komunicirati** ni sa jednom osobom, osim dežurne osobe takmičenja. To znači da morate **raditi samostalno** i **ne smijete koristiti Internet**. Takođe, zabranjena je upotreba bilo kakvih ranije napisanih programa ili dijelova programa.

Po isteku vremena predviđenog za takmičenje, na desktopu u folderu sa imenom ***Takmicenje2024*** moraju se nalaziti datoteke sa snimljenim izvornim kôdovima rješenja. Vaša rješenja možete upload-ovati na sistem za takmičenje, ako je dostupan, i provjeriti koliko bodova ste dobili. Nakon takmičenja, komisija će testirati vaša rješenja još jednom. Na kraju svakog zadatka dati su primjeri test podataka. Ti primjeri služe da bi vam tekst zadataka bio što je moguće jasniji i za provjeru formata ulaza i izlaza, a ne služe za provjeru ispravnosti vašeg programa. Ako vaš program radi na tim primjerima, to **nije garancija** da će raditi na službenim podacima za testiranje. Takmičari mogu svaki zadatak

Ukupan broj bodova na nekom zadatku jednak je zbiru bodova test podataka koji se poklapaju sa službenim rješenjem. Ukupan broj bodova jednak je zbiru bodova na svim zadacima.

Sve informacije o zadacima (ime zadatka, vremensko i memorijsko ograničenje, način bodovanja) možete naći na uvodnoj stranici s naslovom Zadaci. Ako vam nije jasno nešto u vezi načina organizacije ovog takmičenja, odmah postavite pitanje dežurnom da vam to razjasni. Tokom cijelog takmičenja možete postavljati pitanja dežurnom u vezi zadataka. Dozvoljena su pitanja **koja razjašnjavaju nejasnoće u tekstu zadatka**. Ne smijete postavljati pitanja u vezi rješavanja zadataka. Prije nego postavite pitanje, pročitajte još jednom zadatak, jer je moguće da ste u prethodnom čitanju preskočili dio teksta zadatka.

**Nemojte štampati ništa što se u zadatku ne traži**, kao npr. poruke tipa ‘Rjesenje je:’ ili ‘Unesite brojeve’ i slično!

**Nepoštovanje ovih pravila ili nepridržavanje formata izlaznih podataka rezultiraće nepovratnim gubitkom bodova.**

Srećno i uspješno takmičenje!

**VAŽNO za jezik Java!**

Ne kreirajte pakete za vaše zadatke, već koristite podrazumijevani (default) paket.

**Zadaci**

|  |  |  |  |
| --- | --- | --- | --- |
| **Zadatak** | **Zadatak1 (A)** | **Zadatak2 (B)** | **Zadatak3 (C)** |
| **Izvorni kôd** | A.java  A.pas  A.c  A.cpp | B.java  B.pas  B.c  B.cpp | C.java  C.pas  C.c  C.cpp |
| **Memorijsko ograničenje** | 256MB | 256MB | 256MB |
| **Vremensko ograničenje**  **(po test podatku)** | 1 sekunde | 1 sekunda | 1 sekunda |
| **Ukupno bodova** | **35** | **35** | **30** |

**A: Ljestvica**

C-dur ljestvica se sastoji od 8 nota: c d e f g a h C. Za potrebe ovog zadatka označimo note brojevima od 1 do 8. Ljestvica može biti odsvirana uzlazno, od 1 do 8, silazno, od 8 do 1, ili razloženo, u bilo kom redosljedu. Zadat je neki odsvirani redosljed. Napravite program koji će odrediti da li je ljestvica odsvirana uzlazno, silazno ili razloženo.

**Ulaz:** U prvom i jedinom redu unosi se 8 brojeva od 1 do 8, uključujući 1 i 8. Svaki broj će se u ulazu pojaviti tačno jednom.

**Izlaz:** U prvi i jedini red izlaza potrebno je ispisati riječ “silazno” ako je odsvirana ljestvica silazna, “uzlazno” ako je odsvirana ljestvica uzlazna i “razlozeno” ako je odsvirana ljestvica razložena.

**Primjeri:**

|  |  |
| --- | --- |
| Ulaz | Izlaz |
| 1 2 3 4 5 6 7 8 | uzlazno |
| 8 7 6 5 4 3 2 1 | silazno |
| 8 1 7 2 6 3 5 4 | razlozeno |

**Napomena:** Potrebno je poslati jedan fajl sa imenom A.c/A.cpp/A.java.

**Napomena: Naziv klase u java fajlu mora takođe biti A.**

**B: Šifra**

Sofija je pronašla staru knjigu u kojoj piše da su drevni Rimljani koristili poseban način da sakriju tajne poruke. Taj način se zove Cezarova šifra – svaki put kada napišu slovo, oni ga zamijene slovom koje je nekoliko mjesta dalje u abecedi.

Na primjer, ako pomjerimo svako slovo za 2 mjesta unaprijed, "a" postaje "c", "b" postaje "d", i tako dalje. Ako dođemo do kraja abecede, nastavljamo od početka ("y" → "a", "z" → "b").

Sofija želi da napiše tajnu poruku svom drugu, ali joj treba tvoja pomoć da napravi program koji će poruku šifrovati.

Ulaz:

Prvi red standardng ulaza sadrži riječ koju je potrebno šifrovati. Riječ će sadržati samo mala slova englese abecede i dužina riječi će biti manja od 100 karaktera. Drugi red standardnog ulaza sadrži cio broj koji predstavlja za koliko mjesta se pomjera svako slovo u abecedi.

Izlaz:

Potrebno je štampati poruku šiftovanu Cezarovim algoritmom.

|  |  |
| --- | --- |
| Ulaz | Izlaz |
| cezar  1 | dfabs |
| aca  2 | cec |

**C. Invazija vanzemaljaca**