

Pravilnik srednjoškolskih takmičenja iz programiranja

Poslednja izmena: 12.02.2008.

Komisija za organizaciju srednjoškolskih takmičenja iz programiranja
e-mail: tak.prog@gmail.com

Sadržaj

- Sadržaj
- Uvod
- Takmičari
- Opšta pravila
- Nivoi takmičenja
 - Okružno takmičenje
 - Državno takmičenje
 - Srpska informatička olimpijada
 - Izorno takmičenje
- Zadaci
 - Programsko okruženje
 - Tipovi zadataka
 - Standardni zadaci
 - Interaktivni zadaci
 - Zadaci sa poznatim ulazom
 - Program takmičenja

Uvod

Ovaj pravilnik donosi Komisija za organizaciju srednjoškolskih takmičenja iz programiranja (u daljem tekstu Komisija) radi uređenja organizacije takmičenja iz programiranja.

U slučaju nesuglasica uzrokovanih različitim tumačenjima ovog pravilnika, biće prihvaćeno ono tumačenje koje da Komisija, i po njemu će dalje biti postupano. Sve mogućnosti koje nisu predviđene ovim pravilnikom, biće razrešene na način koji Komisija odluči.

Cilj srednjoškolskih takmičenja iz programiranja je popularizacija programiranja među učenicima srednjih škola, kao i izbor državnog tima za učešće na međunarodnim takmičenjima iz programiranja.

Takmičenje se održava svake godine u vidu ciklusa sastavljenog od više nivoa, od kojih se svaki održava na mestima i u vreme koje utvrdi Komisija.

Radi lakšeg čitanja i tumačenja pravilnika u ovakvim žutim pravougaonicima biće data pojašnjenja određenih odredbi pravilnika, a takođe i saveti i preporuke takmičarima i njihovim mentorima.

Takmičari

Takmičari mogu biti samo učenici koji su u vreme održavanja takmičenja upisani u neku od srednjih škola sa područja Srbije. Izbor učenika za učešće na prvom nivou takmičenja (okružno takmičenje) obavlja lokalni odbor, dok se izbor učenika za učešće na višim nivoima takmičenja obavlja po pravilima opisanim u delu nivoi takmičenja.

Takmičari su podeljeni u dve kategorije: "A" i "B". Razlika između kategorija je u zadacima koje takmičari rešavaju i u granici prolaznosti na sledeći nivo takmičenja. Svi takmičari koji su učenici odeljenja koja rade po programu matematičke gimnazije takmiče se u "A" kategoriji, dok takmičari koji su učenici ostalih odeljenja, mogu da biraju u kojoj će se kategoriji takmičiti. Kada takmičar jednom izabere da se takmiči u "A" kategoriji mora ostati u toj kategoriji do kraja godišnjeg ciklusa takmičenja. Ovaj izbor takmičari mogu učiniti u proizvoljnom trenutku (čak i za vreme takmičenja). Ako je takmičar dao na pregledanje bar jedan zadatak koji je zadat takmičarima iz "A" kategorije, a nije zadat takmičarima iz "B" kategorije, smatraće se da je takmičar izabrao da pređe u "A" kategoriju.

Takmičari iz "A" kategorije rešavaju teže zadatke, ali je broj takmičara koji pređe na sledeći nivo takmičenja uglavnom veći u "A" kategoriji nego u "B" kategoriji.

Opšta pravila

Za vreme takmičenja svakom takmičaru mora biti obezbeđen jedan računar sa instaliranim operativnim sistemom, pomoćnim programima, kompjuterima, fajlovima, itd. potrebnim za takmičenje. Svi vidovi komunikacije takmičarevih računara sa drugim računarima ili uređajima moraju biti onemogućeni. Ispravnost celog sistema mora biti proverena pre početka takmičenja. Takođe, takmičarima se moraju obezbediti nesmetani uslovi za rad u toku takmičenja.

Ako za vreme takmičenja dođe do tehničkih problema, mora se pokušati da se ti problemi otklone što brže i sa što manjom štetom po takmičare. Ako je takmičar značajno oštećen, a on sam nije uzrokovao tu štetu, takmičaru šteta mora biti nadoknađena. Takmičaru će se šteta nadoknaditi produžavanjem vremena za izradu zadataka. Izuzetno retko, u opravdanim slučajevima, Komisija može doneti odluku da se šteta takmičaru nadoknadi na neki drugi način (npr. pozivanjem na sledeći nivo takmičenja).

Takmičar na takmičenju ne sme pokušati da koristi bilo kakav pisani ili štampani materijal, tekstove u elektronskom obliku, postojeće kodove na dodeljenom mu računaru (osim onih koji su delovi standardnih paketa zvaničnih programskih jezika), niti bilo kakve uređaje (osim dodeljenog računara) sposobne za komunikaciju ili skladištenje podataka. Takmičar sme komunicirati samo sa dežurnim i nadzornim osobljem.

Neposredno nakon završetka takmičenja, takmičarima će biti omogućeno da svoj rad sačuvaju na svoje memorijske medije. Takmičar ne sme pokušati ovo da uradi pre završetka takmičenja, bez obzira na njegovo eventualno ranije odustajanje od takmičenja.

Svaki takmičar koji prekrši ili pokuša da prekrši bilo koju odredbu pravilnika, može biti diskvalifikovan ili sankcionisan na način koji odredi Komisija.

Na početku takmičenja, takmičarima će biti dati papiri sa postavkama zadataka i svim potrebnim informacijama. Vreme trajanja takmičenja će jasno biti naznačeno.

Nakon završetka takmičenja, rešenja takmičara će biti ocenjena i lista sa nezvaničnim rezultatima će biti objavljena u što kraćem roku. Takođe će biti objavljeni svi primeri koji su korišćeni za bodovanje takmičarskih rešenja, kao i ispravna rešenja zadataka sa komentarima.

Po završetku takmičenja sva rešenja takmičara moraju biti čuvana u neizmenjenom obliku, do objave zvaničnih rezultata.

Komisija ima pravo da javno objavi takmičarska rešenja, nakon završetka takmičenja.

Nakon objave nezvaničnih rezultata, takmičari imaju pravo uložiti žalbu Komisiji sve do isteka za to predviđenog roka, koji će biti jasno određen. Ukoliko je reč o okružnom takmičenju, žalbe se najpre moraju uložiti lokalnom odboru, a tek u slučaju odbijanja, takmičar ima pravo uložiti žalbu direktno Komisiji. Lokalni odbori i Komisija su dužni da odgovore na svaku žalbu. Rezultati postaju zvanični (konačni) kada se odobre od strane Komisije, nakon isteka roka za žalbe.

Žalbe koje su zasnovane na izvornom kodu ili ispisu u izlazni fajl se po strogom pravilu odbijaju. Takmičar nema pravo da traži da mu se ispravi greška u izvornom kodu, bez obzira koliko mala ona bila.

Primeri žalbi koje će se razmatrati su žalbe o hardverskom ili softverskom nefunkcionisanju sistema, pogrešna informacija data od ovlašćenog lica, sumnja na grešku u procesu pregledanja (pri čemu molimo takmičare da najpre sami provere ispravnost svog rešenja), itd.

U skladu sa zvaničnom rang listom, takmičarima će biti dodeljene nagrade. Postoje prva, druga i treća nagrada, i pohvala, pri čemu uglavnom više takmičara dobija svaku od nagrada. Nagrade (kao nominalan pojam) su zvanična mera uspeha na takmičenju.

Nivoi takmičenja

Godišnji ciklus takmičenja se sastoji od četiri nivoa: okružnog takmičenja, državnog takmičenja, srpske informatičke olimpijade i izbornog takmičenja. Za svaki nivo takmičenja važe opšta pravila, a specifičnosti svakog nivoa su posebno navedene.

Izbor učenika koji će se takmičiti na višem nivou takmičenja (državno, srpska informatička olimpijada i izborna) pravi Komisija u skladu sa zvaničnim rezultatima takmičenja prethodnog nivoa. Rezultati sa svih

mesta na kojima je prethodni nivo takmičenja održan se sjedinjuju, i formira se jedinstvena lista rezultata svih takmičara. Za svaku kategoriju Komisija određuje broj bodova potreban da takmičar iz te kategorije prođe na sledeći nivo takmičenja.

U posebnim slučajevima, Komisija može pozvati takmičara da učestvuje na nekom nivou takmičenja, ako je taj učenik iz opravdanih razloga bio sprečen da učestvuje na prethodnom nivou takmičenja (što mora da potvrdi takmičarev mentor), i Komisija smatra da je takmičar dovoljno dobar da učestvuje na tom nivou takmičenja (što se utvrđuje na osnovu rezultata koje je taj takmičar postigao ranijih godina na takmičenjima iz programiranja, i preporuke mentora).

Okružno takmičenje

Za sprovođenje okružnog takmičenja zaduženi su lokalni odbori, formirani u okruzima u kojima se odvija takmičenje. Lokalni odbor čine istaknuti profesori, nastavnici, saradnici, stručnjaci, itd. iz oblasti programiranja, računarstva i informatike.

Izbor učenika koji će se takmičiti na okružnom takmičenju pravi lokalni odbor. Oni taj izbor mogu učiniti na proizvoljan, ali nepristrasan način.

Preporučuje se da se svakom učeniku koji želi da se takmiči, to i omogući. Ukoliko za to ne postoje mogućnosti, pogodno je da lokalni odbor organizuje malo pred-takmičenje ili sličan način provere sposobnosti učenika, kako bi izbor bio nepristrasan.

Najkasnije dva dana pre okružnog takmičenja Komisija će poslati lokalnom odboru sav potreban materijal i uputstva za održavanje takmičenja. Lokalni odbor je dužan da postupi prema tim uputstvima.

Elektronska pošta (e-mail) je zvaničan način komunikacije između Komisije i lokalnih odbora. Lokalni odbor mora obavestiti Komisiju da su dobili informacije, uputstva ili materijal koji im je poslat. U slučaju da lokalni odbor nije u mogućnosti da koristi elektronsku poštu, o tome mora što ranije obavestiti Komisiju kako bi se pronašao drugi način za dostavljanje informacija, uputstava i materijala lokalnom odboru. Lokalni odbor je dužan da obezbedi sigurnost komunikacije, kako bi pre takmičenja pristup materijalu za takmičenje imale samo autorizovane osobe.

Ocenjivanje takmičarskih rešenja vrši lokalni odbor u skladu sa uputstvima dobijenim od Komisije. Tako dobijeni rezultati su nezvanični. Oni što pre moraju biti objavljeni takmičarima, čime počinje rok za žalbe. Žalbe se podnose lokalnom odboru. Lokalni odbor mora jasno naznačiti kome, do kada i na koji način se mogu podnositi žalbe. Lokalni odbor je dužan da detaljno razmotri svaku žalbu i izvrši ispravku rezultata ukoliko se žalba pokaže osnovanom.

Lokalni odbor će biti obavešten od strane Komisije do kada mora obustaviti prihvatanje i obradu žalbi i poslati rezultate (sa svim isprankama) Komisiji. Nakon toga žalbe se mogu upućivati Komisiji, ali samo one žalbe koje su pre toga bile upućene lokalnom odboru i na njih dobijen negativan odgovor. Krajnji rok za žalbe upućene komisiji će biti istaknut na web prezentaciji takmičenja iz programiranja. Nakon završetka roka za žalbe, Komisija proglašava zvanične rezultate

Zadaci za okružno takmičenje će biti takvi da se mogu rešiti pomoću 16-bitnih kompajler (Borland Pascal i Borland C/C++), ali se preporučuje da lokalni odbor omogući takmičarima zvanične kompajlere i okruženja.

Lokalni odbor je dužan da što pre po završetku takmičenja, prosledi komisiji radove svih takmičara, i to samo njihove izvorne kodove za sve standardne ili interaktivne zadatke, i sve izlazne fajlove za sve zadatke sa poznatim ulazom.

Državno takmičenje

Za organizovanje državnog takmičenja u potpunosti je zadužena Komisija.

Državno takmičenje može biti održano na više mesta. Određivanje na kom mestu će se takmičiti koji takmičar vršiće se na osnovu blizine mesta odakle je takmičar i na osnovu raspoloživih takmičarskih mesta. Ukoliko takmičar ima posebnih želja u vezi sa mestom na kom bi želeo da se takmiči, to treba da saopšti Komisiji što pre nakon objavljivanja rezultata okružnog takmičenja, kako bi se pokušalo naći zadovoljavajuće rešenje.

Srpska informatička olimpijada

Za organizovanje srpske informatičke olimpijade u potpunosti je zadužena Komisija.

Srpska informatička olimpijada se održava na jednom mestu, gde učestvuju svi pozvani takmičari, i odvija se u dva takmičarska dana.

Na srpskoj informatičkoj olimpijadi ne postoje kategorije, već su svi takmičari ravnopravni (rešavaju iste zadatke i rangiraju se u jedinstvenoj listi rezultata).

Izorno takmičenje

Za organizovanje izbornog takmičenja u potpunosti je zadužena Komisija. Izorno takmičenje se organizuje radi odabira nacionalnog tima, koji će učestvovati na međunarodnim takmičenjima.

Ovaj nivo takmičenja je vanredan nivo, koji ne mora uvek biti održan. O njegovom održavanju odlučuje Komisija u skladu sa trenutnim mogućnostima i potrebama. Odluka o tome da li će se izorno takmičenje održavati ili ne biće objavljena pre početka srpske informatičke olimpijade.

Kao i na srpskoj informatičkoj olimpijadi, na izbornom takmičenju ne postoje kategorije takmičara.

Konačna rang lista se formira tako što se za svakog takmičara sabere broj bodova koji je takmičar osvojio na srpskoj informatičkoj olimpijadi i broj bodova koji je takmičar osvojio na izbornom takmičenju. U konačnoj listi ne smeju biti dva takmičara koji su jednako rangirani, pa u slučaju da više takmičara imaju jednak ukupan broj bodova prednost se daje onima koji su osvojili više bodova na višim nivoima takmičenja tog ciklusa. Ako je i nakon ovoga poredak neodređen, prihvata se način za utvrđivanje poretka koji izabere Komisija.

Na svako međunarodno takmičenje na kome naša država učestvuje, šalju se takmičari sa vrha ukupne rang liste sa poslednjeg završenog ciklusa takmičenja, tako da za svakog takmičara koji učestvuje na međunarodnom takmičenju važi da svaki takmičar koji je ispred njega u ukupnoj rang listi ili učestvuje na tom istom takmičenju ili je sprečen ili ne želi da učestvuje.

Zadaci

Svi zadaci koji se zadaju na takmičenjima su napravljeni da budu algoritamski po prirodi.

Programsko okruženje

Za rešavanje zadataka, takmičari mogu da koriste programske jezike Pascal, C i C++. Takmičarima moraju biti obezbeđena zvanična okruženja (editor-i, debugger-i, ...) i zvanični kompajleri. Pored toga, takmičarima mogu biti obezbeđena dodatna okruženja i kompajleri.

Zvanični kompajleri su:

- Free Pascal 2.0.0
- MinGW (GCC 3.4.2)

Zvanična okruženja su:

- Free Pascal IDE 1.0.6
- Dev-C++ 4.9.9.2

Dodatna okruženja i kompajleri su:

- Borland Pascal 7.0
- Borland C 3.1
- RHIDE (DJGPP) 1.5

Ako organizatori takmičenja nisu u mogućnosti obezbediti navedene verzije zvaničnih kompajlera i okruženja, mogu se koristiti neke od približnih verzija koje su u potpunosti kompatibilne sa zvaničnim verzijama.

Zadaci koji se zadaju na okružnom takmičenju su koncipirani tako da se mogu rešiti uz pomoć 16-bitnih kompajlera (Borland Pascal 7.0 i Borland C 3.1). Ukoliko lokalni odbor nema mogućnosti, nije u obavezi da takmičarima obezbedi zvanične kompajlere i okruženja, već umesto njih, može obezbediti samo

navedene 16-bitne kompajlere i okruženja. U tom slučaju se samo ti isti kompajleri mogu koristiti prilikom pregledanja.

Ukoliko se kao rešenje zadatka traži izvorni kod, on mora biti napisan za neki od zvaničnih kompajlera. Prilikom testiranja, kodovi će biti kompajlirani isključivo sa zvaničnim kompajlerima (Od ove odredbe se može odstupiti samo na okružnim takmičenjima kao što je opisano u prethodnom paragrafu).

Često se objavljuju nove verzije okruženja i kompajlera, pa je zato pogodno pred svako takmičenje proveriti koje verzije su aktuelne kao zvanične.

Navedeni kompajleri i okruženja, kao i detaljne informacije o njima se mogu naći na sajtovima:

Free Pascal: <http://www.freepascal.org/>

MinGW (GCC) kompajler: <http://www.mingw.org/>

Dev-C++ okruženje: <http://www.bloodshed.net/>

RHIDE okruženje: <http://www.rhide.com/>

DJGPP kompajler: <http://www.delorie.com/djgpp/>

Tipovi zadataka

Standardni zadaci

U zadacima ovog tipa, od takmičara se traži da napiše program koji rešava opisani problem. Ulazni podaci se prosleđuju takmičarevom programu, koji na osnovu njih treba da izračuna rezultat i prezentuje ga na traženi način u vidu izlaznih podataka.

Format ulaznih i izlaznih podataka su precizno zadati. Ulazni podaci se učitavaju iz fajla ili sa standardnog ulaza, i oni će uvek biti u opisanom obliku. Korektnost ulaznih podataka nije potrebno proveravati u programu. Takmičarev program mora ispisivati izlazne podatke isključivo u traženom formatu.

Efikasnost igra važnu ulogu u zadacima ovog tipa, pri čemu se pod efikasnošću smatra optimizovanost programa pri korišćenju određenih resursa (trajanje izvršavanja programa i količina korišćene memorije).

Ovakvi zadaci najčešće imaju više rešenja različitih efikasnosti. Bodovanje zadataka će biti koncipirano tako da efikasnija rešenja donose više bodova, ali da svako ispravno rešenje donosi određeni broj bodova. Najčešće će samo najefikasnija ispravna rešenja biti bodovana sa maksimalnim brojem bodova.

Takmičar na pregledanje predaje samo izvorni kod svog programa. Taj program će biti proveravan na unapred pripremljenom skupu test primera koji će služiti kao ulazni podaci za program. Svaki korektno rešen test primer donosiće takmičaru broj bodova određen za taj test primer. Mogući su i zadaci sa parcijalnim bodovanjem test primera, u kom slučaju će kriterijum za bodovanje biti saopšten takmičarima zajedno sa zadacima. Broj bodova koji je takmičar osvojio na zadatku jednak je zbiru bodova koje je osvojio na svim test primerima tog zadatka. Test primeri nisu poznati takmičarima i smeju biti objavljeni tek nakon završetka takmičenja.

Jedan vid zadataka sa parcijalnim bodovanjem su zadaci u kojima se traži što je bolje rešenje (u odnosu na neki dobro definisani kriterijum). Tada je broj osvojenih bodova neka funkcija izlaza takmičarevog programa, takva da bolja rešenja dobijaju više bodova.

Zajedno sa formulacijom zadatka, takmičarima će biti saopštena vremenska i memorijska ograničenja koja program treba da ispoštuje. Na test primerima na kojima takmičarev program ne ispoštuje zadata ograničenja takmičar neće dobiti bodove. Zadato vremensko ograničenje se odnosi na određenu računarsku konfiguraciju (referentna konfiguracija), koja će biti navedena uz samo ograničenje. Dakle, vremensko ograničenje je vreme za koje bi program trebao da završi sa radom ako bi se izvršavao na računaru navedene konfiguracije. Takmičaru će biti omogućen uvid u konfiguraciju računara koji mu je dodeljen.

Da bi se vreme izvršavanja programa lakše preračunavalo, takmičarima će biti obezbeđen program koji je napravljen tako da se na referentnoj konfiguraciji izvršava tačno jednu sekundu. Taj program će ispisivati koliko je vremena radio na konfiguraciji na kojoj je pokrenut.

S obzirom da takmičarev računar najverovatnije nema konfiguraciju jednaku referentnoj, takmičar

treba da bude sposoban da preračunava vremena izvršavanja programa za različite računarske konfiguracije. Takođe, za potrebe testiranja mora se odrediti vremensko ograničenje za računar na kome se testira, tako da ono bude u skladu sa zadatim ograničenjem na referentnoj konfiguraciji (Na okružnom takmičenju za ovo je zadužen lokalni odbor). Vremensko ograničenje za neki računar dobija se množenjem vremenskog ograničenja za referentnu konfiguraciju i broja dobijenog pomoću pomenutog programa.

Garantuje se da postoji program koji korektno rešava postavljen problem i koji osvaja maksimalan broj bodova na svim test primerima. Međutim, ne garantuje se da postoji program koji korektno rešava problem i koji ujedno za sve moguće validne ulazne podatke (koji zadovoljavaju postavljena ograničenja) ispunjava sve uslove efikasnosti postavljene u zadatku (vreme izvršavanja, memorijsko ograničenje, ...).

Ovo znači da takmičar treba da se potruži da napiše što bolje (efikasnije) rešenje, bez obzira što to rešenje možda ne zadovoljava zadate uslove efikasnosti u svim slučajevima.

Ako program koji takmičar preda na pregledanje na nekom test primeru ne ispoštuje bilo koju od sledećih obaveza, na tom test primeru takmičar će dobiti nula bodova.

- Program mora pravilno završiti sa radom (exit code mora biti 0).
- Program ne sme prekoračiti zadato vremensko ili memorijsko ograničenje.

Takmičarev program ne sme prekršiti sledeće odredbe. Ukoliko to učini takmičar može na tom zadatku biti bodovan sa 0 bodova, diskvalifikovan sa takmičenja ili sankcionisan na način koji odredi Komisija (ili lokalni odbor, u slučaju okružnog takmičenja).

- Program sme da koristi isključivo module i biblioteke koji su priloženi uz zvanične kompajlere, kao i one koji su navedeni u formulaciji zadatka.
- Program ne sme pokušati da koristi ni jedan drugi fajl osim fajlova navedenih u formulaciji zadatka, piše na ekran, koristi standardan ulaz ili izlaz osim ukoliko se to ne traži u formulaciji zadatka, šalje podatke na bilo koji eksterni uređaj, pristupa mreži, pokreće druge programe, direktno pristupa memoriji, koristi sistemske usluge, čita ili menja sistemska podešavanja, (ovo uključuje zabranu očitavanja sistemskog vremena). Jedini slučaj kada je nešto od ovoga dozvoljeno je kada je to jasno navedeno u formulaciji zadatka.
- Program mora da bude deterministički, tj. svaki put kada se pozove sa istim ulaznim podacima program mora dati iste izlazne rezultate.

Korišćenje generatora slučajnih brojeva je dozvoljeno, ali svaki put kada se program pokrene mora biti generisan isti niz slučajnih brojeva. To se postiže tako što se generator inicijalizuje postavljanjem semena (seed) generisanja na tačno određenu vrednost. Konkretno, u programskim jezicima C i C++, može se koristiti `srand(c)` (pri čemu je `c` neka konstanta) i potom pozivati `rand()`, a u Pascal-u se može postaviti `RandSeed := c`, i potom koristiti funkcija `Random()`.

Kao što je naglašeno, zabranjeno je čitanje sistemskog vremena. Očitati vreme će biti jedino moguće ako je na raspolaganje data biblioteka sa funkcijom koja to radi.

Interaktivni zadaci

Interaktivni zadaci su podvrsta standardnih zadataka, i zato za interaktivan tip zadataka važe sve odredbe koje važe za standardan tip zadataka.

U interaktivnim zadacima, takmičarski program ima mogućnost (a često i obavezu) da komunicira sa datim programskim modulom, koji je implementiran kao korisnička biblioteka za programski jezik, koji takmičar koristi. Takmičaru je precizno naveden samo interfejs tog modula, dok njegova implementacija najčešće ostaje skrivena.

Broj poziva neke funkcije ili procedure datog modula može biti ograničen, ili se može tražiti da se taj broj minimizira, pa takmičarevo rešenje može biti bodovano i u zavisnosti od toga koliko efikasno koristi ponuđeni modul.

S obzirom na to da je implementacija datog modula najčešće skrivena, takmičarima može biti dat modul sa istim interfejsom kome je implementacija vidljiva i slobodna za menjanje od strane takmičara. On je dat isključivo da takmičarima bude lakše da pišu i proveravaju svoj program. Taj modul neće biti korišćen prilikom testiranja i nema isto ponašanje i istu funkcionalnost kao modul

koji će se koristiti za testiranje.

Zadaci sa poznatim ulazom

U ovom tipu zadataka, takmičaru je data formulacija problema i određen broj primera ulaznih podataka. Takmičaru je cilj da nađe izlazne podatke koji se dobijaju kada se reši problem sa zadatim ulaznim podacima. Ulazni podaci su najčešće dati u fajlovima (jedan fajl po primeru) u formatu opisanom u tekstu zadatka. Takmičar treba da formatira i sačuva izlazne podatke na način koji je naveden u tekstu zadatka.

Takmičar na pregledanje šalje izlazne podatke (najčešće u vidu fajlova). Bodovanje se vrši proveravanjem tačnosti tih izlaznih podataka. Za svaki korektno rešen primer, takmičar dobija broj bodova predviđen za taj primer. Mogući su i zadaci sa parcijalnim bodovanjem, u kom slučaju će kriterijum za bodovanje biti saopšten takmičarima zajedno sa zadacima. Broj bodova koji je takmičar osvojio na zadatku jednak je zbiru bodova koje je osvojio na svim primerima tog zadatka.

Jedino što je bitno da bi se osvojili bodovi je da izlazni podaci budu tačni. Način na koji su oni dobijeni nije bitan. Takmičar ih može ručno računati, pisati programe ili bilo kako kombinovati svoje sposobnosti u cilju dobijanja konačnog rezultata. Naravno, zadaci će biti takvi da je pomoć računara neophodna da bi se zadatak u potpunosti rešio.

Program takmičenja

Godišnji ciklus takmičenja koncipiran je tako da se težina zadataka povećava sa porastom nivoa takmičenja. Na težinu zadatka utiče složenost ideje, količina znanja i nivo programerskog umeća koji su potrebni da bi se zadatak rešio.

Znanja potrebna za određene nivoe takmičenja su navedena u sledećoj listi, pri čemu svaki nivo podrazumeva i sve što je potrebno i za prethodne nivoe. Međutim ova lista nije stroga, već je data orijentaciono, pa su moguća odstupanja od nje.

- Okružno takmičenje
 - Jednostavne strukture podataka (slogovi, nizovi, stringovi, matrice, liste, skupovi)
 - Jednostavni matematički postupci (sumiranje, brojni sistemi, prosti brojevi, Euklidov algoritam)
 - Rekurzija, bektrek, kombinatorna prebrajanja
 - Operacije sa velikim brojevima
 - Elementarni algoritmi za sortiranje (selection sort, insertion sort, bubble sort, counting sort)
 - Predstavljanje osnovnih geometrijskih objekta (tačke, duži, prave, kruznice), jednostavni postupci nad njima (nalaženje preseka, udaljenosti, uglova), analitička geometrija
- Državno takmičenje
 - Brzi algoritmi za sortiranje (quick sort, merge sort), brzo pretraživanje (binarna pretraga, nalaženje k-tog elementa)
 - Predstavljanje stabala i grafova, jednostavni grafovski algoritmi (pretraživanje u dubinu i u širinu, povezane komponente, nalaženje petlji, topološko sortiranje), binarno stablo pretraživanja
 - Dinamičko programiranje
 - Jednostavni geometrijski algoritmi (određivanje površine poligona, pripadnosti tačke poligonu)
- Srpska informatička olimpijada
 - Grafovski algoritmi (najkraći putevi, minimalno stablo razapinjanja, jaka povezanost, dvostruka povezanost, mostovi, artikulacioni čvorovi, Ojlerov put, mečinci, protok)
 - Strukture podataka (prioritetni redovi, heš tabele, disjunktni skupovi, segmentna stabla)
 - Geometrijski algoritmi (konveksni omotač, triangulacija, presek poligona, najbliže tačke)
 - Napredne tehnike traženja uzorka u tekstu

Data lista se jednako odnosi na obe kategorije takmičara. Razlika između kategorija nije u znanjima koja su takmičarima potrebna, već u složenosti ideja i nivou programerskih veština potrebnih da bi se zadaci rešili.

Zadaci su koncipirani tako da je dobra ideja najbitniji faktor za njihovo rešavanje. Ako takmičar zna sve potrebne algoritme i strukture podataka, ne znači da će on umeti da reši sve zadatke. Količina znanja jeste bitan, ali ne i presudan faktor.