

# Програм такмичења из информатике за основне школе (2021/22)

Напомене:

- Ознака (+) значи да је тема обавезна, а ознака ( $\leq 1$ ) да се тема појављује у највише једном задатку на датом нивоу такмичења; У једном задатку може бити заступљено више тема.
- Теме по нивоима су за пети и шести разред исте, али ученици шестог разреда могу (а не морају) из исте теме да добију нешто тежи задатак. Исто важи за седми и осми разред.

## Теме по разредима и нивоима

### Пети и шести разред

квалификације: једноставан улаз и излаз, целобројна и реална аритметика (+), позициони запис бројева са датим бројем цифара и мешовите јединице мере, рад са текстуалним подацима (нискама), једноставни програми разгранате структуре (+), основни алгоритми над малим серијама бројева, основни итеративни алгоритми I ( $\leq 1$ );

околно такмичење: све претходне теме, плус основни итеративни алгоритми I (+), сложен улаз (+), форматирани излази, вишеструко и угнежђено гранање, примена гранања у геометрији праве ( $\leq 1$ ), основни итеративни алгоритми II ( $\leq 1$ ), алгоритми са угнежђеним петљама ( $\leq 1$ ), позициони запис са непознатим бројем цифара ( $\leq 1$ ), основна употреба низова ( $\leq 1$ );

државно такмичење: све теме за околни ниво, плус основна употреба низова (+), примена гранања у геометрији равни, елементи теорије бројева I, анализа подсерија, трансформисање и генерисање низова, матрице ( $\leq 1$ ), елементи ефикасности алгоритама I (+), репрезентација основних типова ( $\leq 1$ ), рекурзија I ( $\leq 1$ );

### Седми и осми разред

квалификације: све теме за квалификације и околно такмичење за 5. и 6. разред, плус алгоритми са угнежђеним петљама ( $\leq 1$ ), примена гранања у геометрији равни;

околно такмичење: све теме за квалификације, плус елементи теорије бројева I, анализа подсерија, трансформисање и генерисање низова, матрице ( $\leq 1$ ), елементи ефикасности алгоритама I (+), репрезентација основних типова ( $\leq 1$ );

државно такмичење: све теме за околни ниво, плус рекурзија I, рекурзија II ( $\leq 1$ ), елементи теорије бројева II ( $\leq 1$ ), елементи ефикасности алгоритама II ( $\leq 1$ );

по новом (2022/23):

- елементи ефикасности алг. II (сада 3) на државном само за осми р.
- рекурзија II иде тек на СИО,

### **Сви разреди**

Српска информатичка олимпијада: све претходно набројане теме, плус графови;

Изборно такмичење: све теме заступљене на међународним јуниорским такмичењима из програмирања, на којима учествује екипа Србије.

## **Детаљнији опис тема из програма такмичења**

-- За неке теме су као илустрација наведени карактеристични задаци које та тема обухвата, али се тема не ограничава на набројане примере. Уместо примерима задатака, тема је у неким случајевима описана и очекиваним знањем и способностима.

-- Под малом серијом бројева се подразумева серија која садржи фиксиран и у тексту задат број елемената (обично три до пет елемената).

-- Реч низ се овде користи као опште име за линеарну (једнодимензионалну) изменљиву колекцију података. У различитим језицима такав тип података има различита имена, а у неким језицима постоји и више таквих типова (низ, вектор, листа и сл.) Слично томе, реч матрица се овде користи као опште име за дводимензионални низ.

- Једноставан улаз и излаз
  - Учитавање сваког податка (броја, ниске, карактера) задатог у посебном реду
  - Исписивање без задавања изгледа податка (формата исписивања)
- Целобројна и реална аритметика

- Формирање аритметичког израза на основу текста математичког проблема из области обрађених у оквиру редовног школског градива математике
- Запис аритметичког израза у програму
- Апсолутна вредност броја и примене
- Целобројни количник и остатак при дељењу целих бројева и примене
- Заокруживање реалног броја на већи, мањи или најближи цео број
- Позициони запис бројева са датим бројем цифара и мешовите јединице мере
  - Формирање природног броја од датих цифара или група цифара
  - Издвајање цифара природног броја са унапред познатим бројем цифара
  - Превођење јединица за мерење дужине, масе, запремине (нпр.  $2354\text{мл} = 2\text{л } 3\text{дл } 5\text{цл } 4\text{мл}$ )
  - Превођење времена задатог у сатима, минутима и секундама у секунде и обратно. Поређење, сабирање и одузимање времена датих у сатима, минутима и секундама.
  - Превођење углова задатих у степенима, минутима и секундама у секунде и обратно. Поређење, сабирање и одузимање углова датих у степенима, минутима и секундама.
- Рад са текстуалним подацима (нискама)
  - Проналажење подниске (секвенце) у ниски, замена једне подниске другом; овде се типично користе библиотечке функције
  - Налажење датог знака у ниски, растављање ниске на два дела – до и од датог знака (на пример, издвајање имена и презимена)
  - Растављање ниске на више делова раздвојених датим знаком или знаковима (на пример, читање времена задатог у формату hh:mm:ss)
  - Издвајање карактера или задатих сегмената ниске, обртање делова ниске
  - Спајање мале серије ниски у већу ниску, са или без уметања посебних знакова на спојевима

По новом,

- и конверзија ASCII карактера у број (код) и обрнуто
- "Спајање мале серије" постаје "Спајање серије"
- Једноставни програми разгранате структуре
  - Условно извршавање делова програма
  - Релацијски оператори над целим бројевима
  - Испитивање дељивости

- Логички оператори и изрази (на пример, испитивање да ли је година преступна)
- Променљиве које садрже логичке вредности
- По новом у гранање 1 спада и
  - Гранање на основу припадности надовезаним бројевним интервалима (на пример, одређивање агрегатног стања воде на основу дате температуре)
  - Гранање на основу дискретног скупа вредности (на пример, одређивање броја дана на основу редног броја месеца)
  - Лексикографско поређење две торке (нпр. поређење два датума)
- Основни алгоритми над малим серијама бројева
  - Одређивање најмањег и највећег елемента мале серије бројева, као и његовог редног броја
  - Филтрирање мале серије бројева (на пример, одређивање на колико од 4 писмена задатка је ученик добио оцену 5)
  - Уређивање (сортирање) мале серије бројева и примене ради поједностављивања решења задатка (на пример: формирати најмањи троцифрен број од 4 дате цифре; проверити да ли се од неке 3 од 4 дате дужи може формирати једнакокраки троугао)
- Основни итеративни алгоритми I (са неугнежђеним петљама без гранања)
  - Генерисање правилне серије бројева (на пример: аритметичке или геометријске прогресије; равномерно размакнутих тачака неког реалног интервала, ...)
  - Израчунавање броја елемената, збира, производа серије бројева. Израчунавање просека (аритметичке средине).
  - Одређивање минимума или максимума серије бројева
  - Пресликавање серије бројева (на пример, табелирање функције)
- Сложен улаз
  - читавање унапред познатог броја података (бројева, ниски), задатих у једном реду
  - читавање података из више редова, где се прво читава број редова и где сваки ред садржи један податак
  - читавање више података из једног реда, где се претходно читава број података
  - читавање података из више редова, где се прво читава број редова и где сваки ред садржи унапред познат број података
- Форматиран излаз
  - Исписивање реалних бројева са задатим бројем децимала
  - Исписивање података у пољу дате ширине, са потребним бројем водећих нула или других карактера. На пример исписивање датума у формату 01.09.2100. или времена у формату 23:05:00 и слично.

- Вишеструко и угнежђено гранање
  - Гранање на основу припадности надовезаним бројевним интервалима (на пример, одређивање агрегатног стања воде на основу дате температуре)
  - Гранање на основу дискретног скупа вредности (на пример, одређивање броја дана на основу редног броја месеца)
  - Хијерархија услова (на пример, одређивање квадранта којем припада тачка задата својим целобројним координатама)
  - Лексикографско поређење две торке (нпр. поређење два датума)

Савет: код задатака разгранате структуре, тежити да се решење реализује кроз анализу што мањег броја случајева. Број случајева које треба анализирати може да се смањи применом одређивања минимума и максимума два броја (на пример: одређивање пресека два временска интервала; одређивање површине уније два правоугаоника паралелних осама), применом апсолутне вредности (симетрија) итд.

- Примена гранања у геометрији праве (по новом део гранања 2)
  - Проблеми у којима је потребно одредити односе тачака и дужи на правој или на временској оси (на пример: да ли тачка припада дужи; растојање тачке и дужи на правој; пресек две дужи; минималан интервал који садржи две дужи; редослед тачака, тј. која од три тачке је између остале две)
- Основни итеративни алгоритми II (неугнежђене петље са гранањем)
  - Тражење првог или последњег елемента који испуњава услов
  - Филтрирање серије бројева (одређивање елемената који задовољавају дати услов)
  - Провера да ли бар један елемент серије испуњава услов и да ли сви елементи испуњавају услов
  - Однос суседних елемената серије (на пример, провера да ли је серија уређена)
  - Комбиновање основних итеративних алгоритама (на пример, одредити збир квадрата свих негативних међу учитаним бројевима)
- Алгоритми са угнежђеним петљама
  - Генерисање комбинација и варијација  $k$ -торки од  $n$  бројева ( $k$  дато у тексту,  $n$  се читава)
  - Набрајање подниски дате ниске у различитим редоследима (на пример, сви префикси ниске или све подниске у растућем редоследу дужине)
  - Табеларни прикази (на пример, исписати таблицу множења, исписати Паскалов троугао као троугаону табелу)

- Цртање троуглова, ромбова и сличних облика помоћу звездица, цртица и других ASCII карактера
- Обрада више задатих серија елемената (на пример, ако се за сваког ученика учитавају оцене, који ученик има највећу просечну оцену)
- Позициони запис броја са непознатим бројем цифара
  - Одређивање цифара датог броја (здесна налево)
  - Формирање броја на основу цифара слева надесно (на пример, Хорнеровом шемом)
  - Формирање броја на основу цифара здесна налево
  - Обрада серије цифара броја основним итеративним алгоритмима (на пример, одредити збир кубова непарних цифара датог броја)
- Основна употреба низова
  - Низови фиксне дужине. Низови (библиотечке колекције) који се могу допуњавати
  - Секвенцијално приступање елементима низа унапред или уназад
  - Вишеструка примена основних итеративних алгорита на елементе низа (израчунавање збира, производа, минимума, максимума, линеарна претрага итд.), са или без коришћења библиотечких функција, по сопственом избору
  - Пресликавање и филтрирање елемената низа, са или без коришћења библиотечких функција по сопственом избору
- Примена гранања у геометрији равни (по новом део гранања 2)
  - Проблеми у којима је потребно одредити односе тачака и правоугаоника којима су ивице паралелне осама координатног система у равни (на пример: да ли је тачка у правоугаонику; пресек два правоугаоника; најмањи правоугаоник који садржи дате тачке; одређивање тзв. „такси“ тј. Менхетн растојања између тачака у равни, дефинисаног са  $d = \text{abs}(x_1 - x_2) + \text{abs}(y_1 - y_2)$ ).
- Елементи теорије бројева I
  - Број и збир делилаца природног броја
  - Утврђивање да ли је број прост, налажење простих бројева
  - Растављање броја на прости чиниоце и примене (на пример, налажење најмањег природног броја, којим треба помножити дати број да се добије потпун квадрат или куб)
  - Напомена: очекивана сложеност алгоритама из претходне три ставке је корен из ен.
  - Израчунавање НЗД и НЗС два или више бројева помоћу Еуклидовог алгорита и примене (на пример, скраћивање разломака)
- Анализа подсерија
  - Разлагање улазне серије на подсерије: подсерије фиксне дужине (на пример, седмице у једној години), подсерије одређене неким

- сепаратором унутар оригиналне серије (на пример, подсерије бројева раздвојене нулама), подсерије елемената од краја претходне до испуњења услова (на пример, пуњење једнаких врећа редом, предметима датих величина), подсерије одређене односом сваког елемента у односу на претходни (на пример, растуће подсерије)
  - Рачунање и комбиновање статистика на подсеријама (на пример, просечна температура у свакој седмици током једне године, најдужа серија узастопних победа кошаркаког тима)
- Трансформисање и генерисање низова
  - Премештање елемената у оквиру низа (на пример, циклично померање за одређени број места у једном или другом смеру, обртање делова низа, ...)
  - Формирање новог низа на основу постојећег, или постојећих низова или њихових делова. Резултујући низ не мора бити исте дужине као полазни.
  - Представљање великих целих бројева помоћу низова и имплементација рачунских операција сабирања, одузимања и множења великих бројева представљених низом цифара.
- Матрице
  - Учитавање (врсту по врсту или колону по колону) и исписивање матрица
  - Секвенцијално приступање елементима матрице у различитим редоследима (по врстама одозго или одоздо, а у оквиру сваке врсте слева или здесна, или по колонама слева/здесна, а у колони одозго/одоздо, по дијагоналама у разним смеровима)
  - Израчунавање статистика (збир, производ, минимум, максимум) над матрицом, линеарно претраживање матрице, пресликавање (у матрицу), филтрирање (у низ).
  - Израчунавање статистика над врстама, колонама и дијагоналама матрице, комбинације (нпр. најмањи од максимума по врстама и сл.)
  - Приступање елементима правоугаоних и троугаоних делова матрице
  - Премештање правоугаоних делова у оквиру матрице
  - Провера да ли матрица или њени правоугаони или троугаони делови испуњавају дати услов (на пример, да ли је квадратна матрица горње-троугаона тј. да ли су сви елементи испод главне дијагонале нуле, да ли је квадратна матрица симетрична, да ли логичка матрица представља рефлексивну релацију)
  - Формирање нове матрице на основу постојеће, или постојећих матрица, по датом правилу (на пример, дате су релације отац, мајка

помоћу логичких или бинарних матрица, израчунати матрицу која представља релацију бака; у бинарној матрици  $A$  је дат распоред мина, а израчунава се матрица  $B$ , тако да је  $b[i][j]$  број мина на пољима суседним пољу  $a[i][j]$ )

- Елементи ефикасности алгоритама I (сада 2)
  - Такмичар треба да уме да оцени временску сложеност алгорита који примењује (видети мали увод у ефикасност на порталу [retlja.org](http://retlja.org)).
  - Проблеми који се могу ефикасније решити (у односу на наивно решење) применом једне или више наредних техника (видети одговарајућа поглавља у другом делу збирке на порталу [retlja.org](http://retlja.org)).
  - примена математичких формула (типично константно време уместо линеарног, на пример: збир првих  $n$  чланова аритметичког низа; број елемената неког интервала дељивих датим бројем итд.)
  - инкрементално рачунање (типично линеарно уместо квадратног времена, на пример: израчунавање свих префиксних сума; израчунавање свих факторијела бројева од 1 до  $n$ )
  - употреба префиксних и суфиксних сума (типично линеарно уместо квадратног времена, на пример: суме разних сегмената целобројног низа; број простих бројева у разним целобројним интервалима)
  - техника два показивача (типично линеарно уместо квадратног времена, на пример: број секвенци у серији позитивних бројева које имају дати збир; проналажење парова елемената сортираног низа, таквих да је збир (или разлика) елемената пара једнака датом броју)
  - (издвојено у ново ЕЕА1) ефикасно сортирање низа у односу на подразумевано поређење елемената у циљу добијања ефикаснијег алгорита (типично ен-лог-ен уместо квадратног времена, на пример: пребројавање различитих међу учитаним елементима; одређивање пресека два низа итд.). Напомена: ово сортирање се може извести једноставним позивом библиотеке функције за сортирање.
  - (издвојено у ново ЕЕА1) бинарна претрага сортираног низа у односу на подразумевано поређење елемената (типично лог-ен уместо линеарног, на пример: број елемената сортираног низа већих од дате вредности; број појављивања елемента у сортираном низу)
  - Употреба структура/торки, речника/мапа и скупова ради добијања ефикаснијег алгорита, на пример налажење парова датог збира у низу целих бројева.
  - комбинације претходних техника (на пример: број појављивања елемента у несортираном низу; тројке елемената низа које имају дати збир)



- Репрезентација основних типова
  - Избор одговарајућег типа целобројних података (разумевање проблема прекорачења и познавање опсега и количине заузете меморије за различите целобројне типове, способност да се оцени просторна сложеност алгорита)
  - Конверзија ASCII карактера у број (код) и обрнуто (пребачено у Рад са текстуалним подацима (нискама))
- Рекурзија I
  - Употреба рекурзивних функција без петљи у решавању задатака са малим улазним подацима (због експоненцијалне временске сложености), на пример: формирање бинарних или тернарних n-торки, тј. n-торки од два или три различита елемента, које испуњавају неки услов (као што су све тернарне n-торке у којима нема суседних једнаких елемената); рекурзивни цртежи помоћу ASCII карактера; модификације проблема Ханујских кула
  - додато Алгоритми из области комбинаторике – генерисање пермутација, комбинација и варијација са или без понављања (свих или оних које задовољавају дати услов).

Савет: препоручљиво је вежбање рекурзије и на задацима који се могу једноставно решити и помоћу петље (на пример збир цифара броја, парност броја цифара и сл.), мада такви задаци нису уврштени у тему „Рекурзија I“.

- Рекурзија II
  - (премештено у рек 1) Алгоритми из области комбинаторике – генерисање пермутација, комбинација и варијација са или без понављања (свих или оних које задовољавају дати услов).
  - Техника рекурзивне претраге по дубини и претраге са враћањем (backtracking), на пример раздвајање низа реалних бројева на две групе са што приближнијим збировима.
  - Техника „подели па владај“, на пример, збир k највећих елемената датог низа.
- Елементи теорије бројева II
  - Ератостеново сито
  - Проширени Еуклидов алгоритам за решавање линеарних Диофантових једначина
- Елементи ефикасности алгоритама II
  - Техника динамичког програмирања, на пример у датом низу целих бројева одредити број сегмената са парним збиром.
  - Техника мемоизације рекурзивних функција. Нпр. израчунавање  $f(n)$ , ако је  $f(1)=1$ ,  $f(2k)=f(k)$ ,  $f(2k+1)=f(k+1)+f(k)$

- Бинарна претрага по решењу, на пример одређивање највећег квадрата који се може уписати у низ (слепљених) стубаца ширине 1 и задатих висина.
- Ефикасно сортирање по задатом критеријуму, могуће сложених података, као што су торке, структуре, редови матрице и слични (типично ен-лог-ен уместо квадратног времена, на пример, сортирање тачака равни по растојању од координатног почетка). Напомена: у савременим програмским језицима (C++, Python, C# ...) овакво сортирање се може извести позивом библиотеке функције за сортирање уз употребу додатних параметара.
- Сортирање пребројавањем (counting sort) и разврставањем (radix sort)
- Графови
  - Представљање графа у програму, формирање графа, алгоритми за најкраћа растојања између чворова у графу (Dijkstra, Floyd–Warshall), за најмање разапето дрво (Kruskal, Prim). Не очекују се оптимална решења, довољна су решења квадратне сложености (без употребе union-find алгоритма за Краскалов алгоритам, без редова са приоритетом за Дајкстрин и Примов алгоритам).
  - Претрага у ширину (нпр. најкраћи излаз из лавиринта датог матрицом, бојење просторија у „лабиринту“)
  - Тополошко сортирање графа.

Списак математичких појмова и формула чије познавање се очекује на одговарајућем нивоу такмичења - видети списак тема по нивоима такмичења

- Појам аритметичке и геометријске прогресије, формуле за следећи члан, за n-ти члан
- Формуле за координате тачке, добијене симетријом дате тачке у односу на неку од координатних оса или координатни почетак у равни; за координате средишта дужи; за координате тачке, добијене транслацијом дате тачке за дати вектор у равни
- Питагорина теорема, формула за еуклидско растојање тачака датих координатама