

ОДСЕК ЗА СОФТВЕРСКО ИНЖЕЊЕРСТВО
АЛГОРИТМИ И СТРУКТУРЕ ПОДАТАКА 2
2020-2021
- први домаћи задатак -

Опште напомене:

1. Пре одбране сви студенти раде тест знања који се ради на рачунару коришћењем система Moodle (<http://elearning.rcub.bg.ac.rs/moodle/>). **Сви студенти треба да креирају налог и пријаве се на курс пре почетка лабораторијских вежби.** Пријава на курс ће бити прихваћена и важећа само уколико се студент региструје путем свог налога електронске поште на серверу **mail.student.etf.bg.ac.rs**.
2. Домаћи задатак 1 састоји се од два програмска проблема. Студенти проблеме решавају **самостално**, на програмском језику С или С++.
3. Реализовани програми треба да комуницирају са корисником путем једноставног менија који приказује реализоване операције и омогућава сукцесивну примену операција у произвољном редоследу.
4. Унос података треба омогућити било путем читања са стандардног улаза, било путем читања из датотеке.
5. Решења треба да буду отпорна на грешке и треба да кориснику пружи јасно обавештење у случају детекције грешке.
6. Приликом оцењивања, биће узето у обзир рационално коришћење ресурса. **Примена рекурзије се неће признати као решење проблема које може освојити максималан број поена.**
7. За све недовољно јасне захтеве у задатку, студенти треба да усвоје разумну претпоставку у вези реализације програма. Приликом одбране, демонстраторе треба обавестити која претпоставка је усвојена (или које претпоставке су усвојене) и која су ограничења програма (на пример, максимална димензија низа и слично). Неоправдано увођење ограничавајуће претпоставке повлачи негативне поене.
8. Одбрана првог домаћег задатка ће се обавити према распореду који ће накнадно бити објављен на сајту предмета. Пријава за одбрану омогућена је преко Moodle система. Детаљније информације су објављене на предметном сајту.
9. За решавање задатака који имају више комбинација користити следеће формуле.
(**R** – редни број индекса, **G** – последње две цифре године уписа):
$$i = (R + G) \bmod 2$$
$$j = (R + G) \bmod 3$$
10. Предаја домаћих ће бити омогућена преко Moodle система до **уторка, 27.10.2020. у 23:59**. Детаљније информације су објављене на предметном сајту.
11. Предметни наставници задржавају право да изврше проверу сличности предатих домаћих задатака и коригују освојени број поена након одбране домаћих задатака, као и да пријаве теже случајеве повреде Правилника о дисциплинској одговорности студената Универзитета у Београду Дисциплинској комисији Факултета.

Задатак 1 – претраживање линеарних структура података [50 поена]

Нека је дата матрица димензија $M \times N$ која садржи целобројне кључеве. Кључеви могу да се понављају. У зависности од редног броја проблема који се решава, познато је да су и врсте и колоне матрице уређене на један од следећих начина:

0. неоппадајуће
1. нерастуће

У општем случају линеаризацијом матрице по врстама или колонама се не добија уређени низ. Осмислити и имплементирати функцију за **ефикасну** претрагу овакве структуре података на задати кључ.

Програм треба да омогући следеће функционалности:

1. **[10 поена]** Унос матрице из датотеке или са стандардног улаза
2. **[10 поена]** Генерисање матрице са описаним карактеристикама
3. **[15 поена]** Претрагу матрице на задати кључ
4. **[10 поена]** Евалуацију перформанси претраживања генерисањем секвенце кључева у задатом опсегу и одређивањем просечног броја корака потребног за претраживање.

Пример матрице 5×4 чије су и врсте и колоне уређене неоппадајуће

10	15	20	25
25	42	45	80
36	56	75	90
46	75	95	115
57	89	111	139

[5 поена] Корисник са програмом интерагује путем једноставног менија. Програм треба да испише садржај менија, а затим да чека да корисник изабере (унесе путем тастатуре) редни број неке од понуђених ставки, након чега, пре извршења, од корисника очекује да по потреби унесе додатне параметре. Поступак се понавља све док корисник у менију не изабере опцију за прекид програма.

Додатак (25.10.2020): Треба омогућити и испис матрице.

Задатак 2 – имплементација AVL стабла са понављањем кључева [50 поена]

Допунити претходни програм операцијама које илуструју рад са висински балансираним стаблом бинарног претраживања (AVL стаблом) које дозвољава понављање кључева. У стабло се смештају целобројни кључеви.

У програм треба додати следеће функционалности:

1. [10 поена] Формирање стабла на основу кључева у матрици, линеаризацијом матрице по врстама
2. [5 поена] Претрагу стабла на задату вредност кључа
3. [10 поена] Уметање кључа у стабло
4. [10 поена] Форматиран испис садржаја стабла
5. [5 поена] Брисање стабла.

У зависности од редног броја проблема који се решава, уметање поновљених кључева реализовати на следећи начин:

0. Уметањем поновљеног кључа као **следбеника** истог кључа
1. Уметањем поновљеног кључа као **претходника** истог кључа
2. Коришћењем **бројача** који за сваки чвор памти број понављања одговарајућег кључа

[5 поена] Корисник са програмом интерагује путем једноставног менија. Програм треба да испише садржај менија, а затим да чека да корисник изабере (унесе путем тастатуре) редни број неке од понуђених ставки, након чега, пре извршења, од корисника очекује да по потреби унесе додатне параметре. Поступак се понавља све док корисник у менију не изабере опцију за прекид програма.

[5 поена] Омогућити поређење перформанси приликом претраге матрице и одговарајућег AVL стабла (тек формираног, без накнадног уметања кључева) на исти скуп кључева. Програм треба да омогући унос кључева који се претражују и са стандардног улаза и из датотеке.

Напомена: Поени за интеракцију са корисником се **не добијају** уколико се само исписује мени, без одговарајућих позива функција у позадини. Код исписа стабла, испис неке комбинације обилазака стабла се не прихвата као адекватно решење, већ стабло треба адекватно форматирати.