

## АЛГОРИТМИ И СТРУКТУРЕ ПОДАТАКА 1 2018-2019

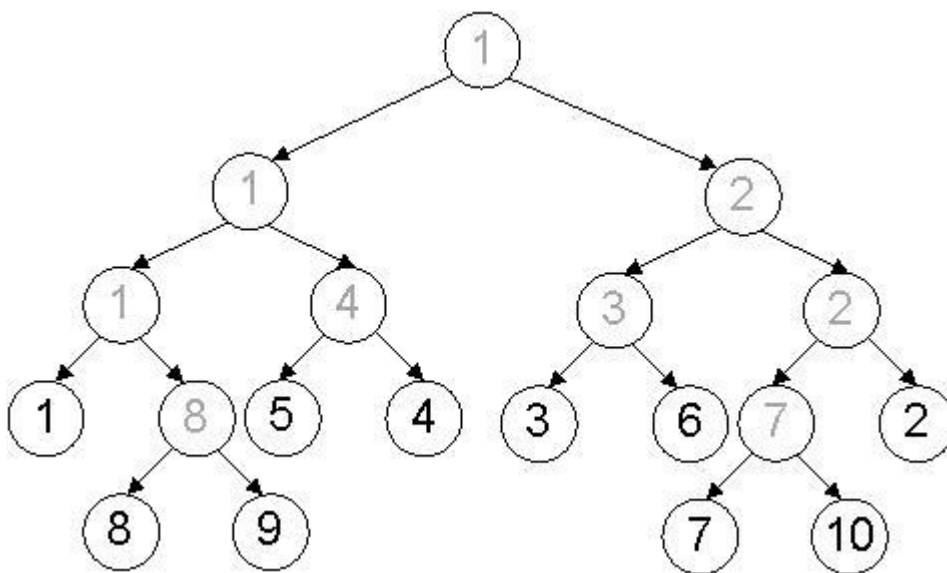
### - домаћи задатак 2 -

#### 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 2 састоји се од једног програмског проблема. Студенти проблем решавају **самостално**, на програмском језику Pascal или C, по избору.
3. Реализовани програм треба да комуницира са корисником путем једноставног менија који приказује реализоване операције и омогућава сукцесивну примену операција у произвољном редоследу.
4. Решење треба да буде отпорно на грешке и треба да кориснику пружи јасно обавештење у случају детекције грешке.
5. Приликом оцењивања, биће узето у обзир рационално коришћење ресурса. **Коришћење рекурзије у решењима повлачи значајно умањење броја поена. Коришћење готових структура података није дозвољено.**
6. 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 матрице и слично). Неоправдано увођење ограничавајуће претпоставке повлачи негативне поене.
7. Одбрана другог домаћег задатка ће се обавити у **понедељак, 15.04.2019. и уторак, 16.04.2019.** према распореду који ће благовремено бити објављен на сајту предмета.
8. Формула за редни број комбинације проблема **i** и **j** који треба решавати је следећа: (R – редни број индекса, G – последње две цифре године уписа):
$$i = (R + G) \bmod 3 + 1$$
$$j = (R + G) \bmod 2 + 1$$
9. Име датотеке која се предаје мора бити **dz2p1.(pas|c)**.
10. Предметни наставници задржавају право да изврше проверу сличности предатих домаћих задатака и коригују освојени број поена након одбране домаћих задатака.

## Задатак – турнирско стабло (100 поена)

Нека се посматра тениски турнир на коме учествује  $N$  играча. Сваки играч је описан својим именом и презименом, као и редним бројем који представља позицију играча на листи носилаца турнира. Турнир се одвија по куп систему у одређеном броју рунди. У оквиру сваке рунде, играч који добије меч пролази у наредну рунду турнира. Да би се формирао костур турнира по коме се врши одигравање мечева, потребно је формирати бинарно стабло одговарајуће дубине. Стабло се састоји од интерних и екстерних чворова. Екстерни чвор стабла представља једног играча, а интерни чвор стабла представља резултат одигравања меча између два играча. За потребе формирања костура турнира, сматра се да у наредну рунду увек пролази играч који се налази на бољој позицији на листи носилаца. Притом, приликом спаривања играча у оквиру одговарајуће рунде турнира, треба водити рачуна да се састану најбоље и најлошије пласирани играч, затим други најбоље и други најлошије пласирани играч и тако редом. Турнирско стабло је увек пуно стабло. Један могући изглед турнирског стабла за 10 играча дат је на слици 1. Сивом бојом су означени предвиђени победници међусобних мечева, док су црном бојом означене иницијалне позиције носилаца турнира.



Слика 1. Турнирско стабло за 10 играча

Након формирања костура турнира по коме ће се одигравати мечеви, у само стабло се уносе стварни резултати мечева на основу којих се на крају одређује победник турнира. Подаци о победнику се увек налазе у корену стабла.

**[75 поена]** Написати програм који илуструје рад са описаним турнирски стаблом. Програм треба да омогући:

1. **[20 поена]** Формирање стабла које представља костур турнира на основу задате листе од  $N$  играча
2. **[15 поена]** Попуњавање стабла резултатима мечева по рундама
3. **[5 поена]** Испис победника турнира
4. **[15 поена]** Исписивање свих могућих парова играча у задатој, незавршеној рунди турнира
5. **[10 поена]** Испис стабла (погодно форматиран, по нивоима)
6. **[10 поена]** Брисање стабла

**[15 поена]** Зависно од дефинисаног редног броја **i** имплементирати и једну од следећих операција:

1. Одређивање броја чворова стабла
2. Одређивање интерне дужине пута у стаблу
3. Одређивање висине стабла

Све наведене операције треба реализовати путем одговарајућих потпрограма чији је један од аргумената стабло над којим се врше операције. Зависно од редног броја **j** потребне обиласке у стаблу имплементирати **искључиво** коришћењем:

1. *preorder* алгоритма
2. *level order* алгоритма

**[10 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.

Више информација о проблему се може видети на следећим страницама:

[https://en.wikipedia.org/wiki/Single-elimination\\_tournament](https://en.wikipedia.org/wiki/Single-elimination_tournament)

[https://www.wimbledon.com/en\\_GB/draws/index.html](https://www.wimbledon.com/en_GB/draws/index.html)