2021 Струкрури данни

Георги Филев F104081

Домашно 3A. Двуклонова Рекурсия

1. Сортиране по Дялове. Запишете рекурсивна дефиниция на нареден масив, както е дадена в лекциите. Един масив е нареден ако се състои от един елемент който е „нареден“, n= 1, И аналогично нареден масив от n елемента е такъв който се състои от 2 залепени „наредени“ масива (ляв и десен) за всеки елемнт на левия масив който е по-малък от десния.
2. Съставете следния седем-местен масиив:



104080043

Проиграйте ***принципната схема*** на алгоритъма Сортиране по Дялове за получения масив, като използвате схема подобна на тази от лекциите. Проиграйте разделянето на масива на „потъване“, като на всяка стъпка от илюстрираното потъване приемате като еталон за сравнение един **произволен** елемент от разделяния масив. Това, че ЕГН-то на мнозина започва с нули не пречи по никакъв начин на работата.



A white paper with writing on it

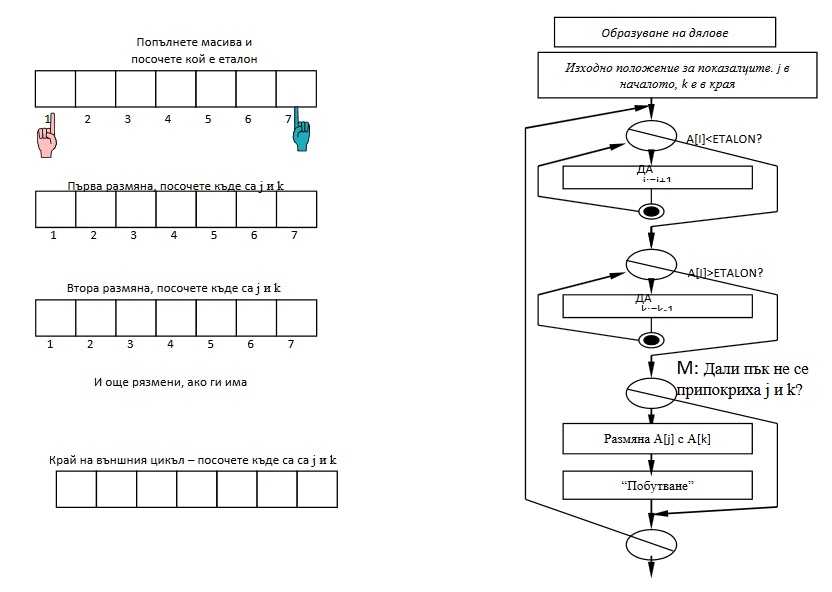
Description automatically generated with low confidence

1. Съставете обща схема на развитието на този рекурсивен процес като програма, т.е. всички повикани копия за вашите данни. Изберете за еталон средата на масива. Едно копие да изглежда така:

Diagram

Description automatically generated

1. Проиграйте на ръка алгоритъма за образуване на дялове за най-началнато, т.е. първото извикване на рекурсивния процес, като ползвате схема, подобна на тази:



Дефинираме коя е лявата и коя е дясната страна.

После първо проверяваме дали лявата страна е по-малка от лявата , след това ако лявата е по-голяма следва че това е Дъно и трябва да обърнем стойностите. Обаче ако не е следва че потъваме и съставяме нашия еталон (примерно някои от средните елементи на масива тоест arr[(десен индекс + ляв индекс) /2]. Така образувахме дялове.

След което докато левия дял и неговия индекс(по точно стойностите на индексите) са по-малки от еталона, алгоритъма минава през тях като ги проверява докато не намери такъв елемент ,че той да е по голям от еталона. След което алгоритъма чака да се случи аналогично същото нещо със десния дял, но той трябва да намери по малка стойност сред индексите спрямо еталона.

След което стойностите на двата отбелязани(„виновни“) индекса се разменят. Съответено алгоритъма продължава докато индекса на левия дял не навлезе в дясната част или обратното.

В последствие се извиква QSort на подмасивите(дяловете) образувани впоследствие на разделянето с цел и те да бъдат подредени.

ДОМАШНОТО СЕ ПРЕДАВА С ТОЧКИТЕ НА ЗАДАНИЕТО