נוסחת נסיגה לפונקציה למציאת הערך הקטן במערך לא ממויין (דוגמה 3 במצגת, שקף 10)

הגדרות:

T(n): מספר הפעולות הנדרש לחיפוש במערך עם n איברים

T(n/2): מספר הפעולות הנדרש לחיפוש במערך עם n/2 איברים

T(1) = 1 T(n/2) = 2 T(n/4) +1 1

T(n) = 2 T(n/2) + 1 2

= 2 [ 2 T(n/4) +1 ] +1 3

= 4 T(n/4) +2+1 T(n/4) = 2 T(n/8) +1 4

= 4 [2 T(n/8) +1] +2 +1 5

= 8 T(n/8) +4+2+1 6

= 2i T(n/2i) +(2i -1) true for all i, specifically for i = lg(n) 7

n/2i = 1 n = 2i **2lg(n) = n** i = lg(n) 8

= 2lg(n) T(n/2lg(n)) +(2lg(n) -1) 9

= n T(1) + n – 1 10

= n\*1+n-1 = 2n – 1 11

--------------------------------------

מוסכמות כתיבת לוגים

בסיס

log(n) 10

ln(n) *e*

lg(n) 2

log3(n) 3

נוסחת נסיגה לפונקציה לחיפוש בינרי במערך ממויין (דוגמה 4 במצגת, שקף 12)

T(1) = 1

T(n) = 1 + T(n/2) T(n/2) = 1 + T(n/4)

= 1 + [1 + T(n/4)]

= 2 + T(n/4) T(n/4) = 1 + T(n/8)

= 2 + [ 1 + T(n/8) ]

= 3 + T(n/8)

= i + T(n/2i) true for every i specifically for i=lg(n)

= lg(n) + T(n/2lg(n))

= lg(n) + T(1)

= lg(n) + 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 2 | 14 | 35 | 42 | 43 | 45 | 47 | 49 | 51 | 56 | 68 | 72 | 73 | 84 | 87 | 92 |

נוסחת נסיגה לפונקציה למציאת מספר במערך לא ממויין (דוגמה 2 במצגת, שקף 8)

T(1) = 1

T(n) = T(n-1) + 1 T(n-1) = T((n-1)-1) + 1

= T(n-2) + 1

= [T(n-2) + 1] + 1

=T(n-2) + 2 T(n-2) = T(n-3) + 1

=[T(n-3) + 1] + 2

=T(n-3) + 3

=T(n-i) + i true for all i, specifically n-1

n-i = 1 -> i=n-1

= T(n-(n-1)) + (n-1)

=T(1) + n-1

= 1 + n – 1

= n

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 2 | 14 | 35 | 92 | 23 | 25 | 47 | 49 | 1 | 56 | 68 | 2 | 73 | 14 | 87 | 12 |

מחלק הגדול ביותר של שני מספרים

Greatest Common Divisor (GCD)

GCD(18, 12) = 6

GCD(24, 12) = 12

GCD(24, 17) = GCD(17, 7) = GCD (7, 3) = GCD (3, 1) = GCD (1, 0) = 1

GCD(40, 25) = 40 % 25 = 15

= GCD(25, 15) 25 % 15 = 10

= GCD(15, 10) 15 % 10 = 5

= GCD(10, 5) 10 % 5 = 0

=GCD(5, 0) = 5 5 % 0 = NaN

משפט: אם מספר מחלק שני מספרים ללא שארית, אז הוא גם מחלק את השארית של החלוקה של הגדול בקטן

int GCD (int big, int small)

{

If (small == 0)

return big;

else

return GCD(small, big % small);

}

7 % 3 = 1

8 % 3 = 2

9 % 3 = 0