

ikebana – descrierea soluției (Autor Prof. Szabo Zoltan – Gr. Sc. „Petru Maior” Reghin)
1. Soluție de 20 de puncte

Se rezolvă cu metoda backtracking, generând fiecare soluție în parte.

2. Soluție de 60 de puncte (O(n))

Pornim de la următoarele formule recursive cu semnificația că

$a[i][j]$ reprezintă numărul de aranjamente florale ce conțin j flori, pe ultima poziție fiind o azalee, iar între ea și cea mai apropiată gerberă din față sa se găsesc $i-1$ flori.

$o[i][j]$ reprezintă numărul de aranjamente florale ce conțin j flori, pe ultima poziție fiind o orhidee, iar între ea și cea mai apropiată gerberă din față sa se găsesc $i-1$ flori.

$ah[i][j]$ reprezintă numărul de aranjamente florale ce conțin j flori, pe ultimele două poziții fiind o azalee și o hortensie, iar între ultima hortensie și cea mai apropiată gerberă din față sa se găsesc $i-1$ flori.

$oh[i][j]$ reprezintă numărul de aranjamente florale ce conțin j flori, pe ultimele două poziții fiind o orhidee și o hortensie, iar între ultima hortensie și cea mai apropiată gerberă din față sa se găsesc $i-1$ flori.

$g[j]$ reprezintă numărul de aranjamente florale ce conțin j flori, pe ultima poziție fiind o gerberă.

$b[i][j]$ reprezintă numărul de aranjamente florale ce conțin j flori, pe ultima poziție fiind o floare diferită de azalee, orhidee, hortensie și gerberă, iar între ea și cea mai apropiată gerberă din față sa se găsesc $i-1$ flori.

Formulele recursive pentru fiecare tip de floare în parte (valorile inițiale le depistați singuri):	Având în vedere că $a[i][j]=o[i][j]$ și $ah[i][j]=oh[i][j]$ pentru orice i și j , numărul de șiruri recursive se reduce astfel:
$b[i][j] = a[i-1][j-1] + o[i-1][j-1] + t * b[i-1][j-1]$ <p style="text-align: right;">pt $j=2 \dots n$ pt $i=2 \dots p-1$</p>	$b[i][j] = 2 * a[i-1][j-1] + t * b[i-1][j-1]$ <p style="text-align: right;">pt $j=2 \dots n$ pt $i=2 \dots p-1$</p>
$a[i][j] = a[i-1][j-1] + o[i-1][j-1] + oh[i-1][j-1] + t * b[i-1][j-1]$ <p style="text-align: right;">pt $j=2 \dots n$ pt $i=2 \dots p-1$</p>	$a[i][j] = 2 * a[i-1][j-1] + ah[i-1][j-1] + t * b[i-1][j-1]$ <p style="text-align: right;">pt $j=2 \dots n$ pt $i=2 \dots p-1$</p>
$o[i][j] = a[i-1][j-1] + o[i-1][j-1] + ah[i-1][j-1] + t * b[i-1][j-1]$ <p style="text-align: right;">pt $j=2 \dots n$ pt $i=2 \dots p-1$</p>	
$ah[i][j] = a[i-1][j-1]$ <p style="text-align: right;">pt $j=2 \dots n$ pt $i=2 \dots p-1$</p>	$ah[i][j] = a[i-1][j-1]$ <p style="text-align: right;">pt $j=2 \dots n$ pt $i=2 \dots p-1$</p>
$oh[i][j] = o[i-1][j-1]$ <p style="text-align: right;">pt $j=2 \dots n$ pt $i=2 \dots p-1$</p>	

$b[p][j] = a[p-1][j-1] + o[p-1][j-1] + t*b[p-1][j-1] + a[p][j-1] + o[p][j-1] + t*b[p][j-1]$ <p style="text-align: right;">pt j=2 .. n</p>	$b[p][j] = 2*a[p-1][j-1] + t*b[p-1][j-1] + 2*a[p][j-1] + t*b[p][j-1]$ <p style="text-align: right;">pt j=2 .. n</p>
$a[p][j] = a[p-1][j-1] + o[p-1][j-1] + oh[p-1][j-1] + t*b[p-1][j-1] + a[p][j-1] + o[p][j-1] + oh[p][j-1] + t*b[p][j-1]$ <p style="text-align: right;">pt j=2 .. n</p>	$a[p][j] = 2*a[p-1][j-1] + ah[p-1][j-1] + t*b[p-1][j-1] + 2*a[p][j-1] + ah[p][j-1] + t*b[p][j-1]$ <p style="text-align: right;">pt j=2 .. n</p>
$o[p][j] = a[p-1][j-1] + o[p-1][j-1] + oh[p-1][j-1] + t*b[p-1][j-1] + a[p][j-1] + o[p][j-1] + oh[p][j-1] + t*b[p][j-1]$ <p style="text-align: right;">pt j=2 .. n</p>	
$ah[p][j] = a[p-1][j-1] + a[p][j-1]$ <p style="text-align: right;">pt j=2 .. n</p>	$ah[p][j] = a[p-1][j-1] + a[p][j-1]$ <p style="text-align: right;">pt j=2 .. n</p>
$oh[p][j] = o[p-1][j-1] + o[p][j-1]$ <p style="text-align: right;">pt j=2 .. n</p>	
$g[j] = a[j-1][j-1] + o[j-1][j-1] + t*b[j-1][j-1]$ <p style="text-align: right;">pt j= 2 .. p</p>	$g[j] = 2*a[j-1][j-1] + t*b[j-1][j-1]$ <p style="text-align: right;">pt j= 2 .. p</p>
$g[j] = a[p][j-1] + o[p][j-1] + t*b[p][j-1]$ <p style="text-align: right;">pt j= p+1 .. n</p>	$g[j] = 2*a[p][j-1] + t*b[p][j-1]$ <p style="text-align: right;">pt j= p+1 .. n</p>

Soluția problemei este $g[n] + 2*a[i][n] + t*b[i][n]$ $i=1..p$

3. Soluție de 100 de puncte ($O(p^3 \log n)$)

Din recurența anterioară construim formula cu înmulțire de matrici de forma $A^{n-p} * B[p] = B[n]$, unde matricea A este de dimensiuni $3p \times 3p$, iar matricile B[p] și B[n] sunt de dimensiune $3p \times 1$. Exponențierea se poate rezolva în $O(\log n)$, deci această metodă are complexitatea $O(p^3 \log n)$.