

Scara – descriere soluție

Metoda de programare utilizată: backtracking.

Reprezentarea informațiilor

1. Vectorul în care generăm soluțiile: sol , cu n componente, cu semnificația $sol[i]$ =înălțimea treptei i
2. Soluția care necesită efort minim o vom reține în vectorul sol_{min}
3. Pentru a verifica dacă toate înălțimile sunt distincte vom construi vectorul caracteristic al înălțimilor: uz cu M componente (nu putem folosi înălțimi mai mari decât M)
 $uz[i]=1$, dacă înălțimea i a fost utilizată și 0 în caz contrar
4. $htot$ =suma înălțimilor treptelor până la un moment dat
5. $efmin$ =efortul consumat pentru sol_{min}
6. Pentru a calcula efortul minim necesar la fiecare pas, vom utiliza un vector ef cu n componente, cu semnificația:
 $ef[i]$ =efortul minim necesar pentru a urca primele i trepte din soluția memorată în vectorul sol

Condiții interne

1. $sol[i]$ din $\{1, 2, \dots, M\}$
2. $sol[i] \neq sol[j]$, pentru orice $i \neq j$
3. $sol[1] + sol[2] + \dots + sol[N] = H$

Efortul minim pentru a urca treptele $1, 2, \dots, i$ se calculează astfel:

$ef[i] = \min\{ef[i-1] + sol[i],$
 $\min\{ef[j] + p + (sol[j+1] + sol[j+2] + \dots + sol[i]) / (i-j), \text{ dacă } (sol[j+1] + sol[j+2] + \dots + sol[i]) \leq M\}$
 $j = i-2, i-3, \dots$