

Descrierea soluției: Problema_2 Felinare

Autor: prof. Marinel Șerban

Problema se rezolva simuland operatiile respective pe un vector.

```
For i := 0 To n-1 Do F[i] := TRUE;    {toate felinarele sunt aprinse}
Toate := FALSE;                      {marchez asta}
While NOT Toate Do                   {cat timp mai sunt felinare stinse}
  For i := 0 To n-1 Do               {treci pe la toate felinarele}
    Begin
      Inc(Operatii);                  {la fiecare felinar ai o operatie}
      If i=0 Then                     {daca esti la primul}
        If F[n-1] Then                {si ultimul este aprins}
          F[i] := NOT F[i]            {schimba starea primului}
        Else                           {altfel nu schimba nimic}
      Else                             {altfel esti undeva in cerc}
        If F[i-1] Then                {daca felinarul anterior este
aprins}
          F[i] := NOT F[i];            {schimba starea felinarului
curent}
      Toate := TRUE;                   {presupun ca toate sunt aprinse}
      j := 0;                          {plec de la primul}
      While (j<=n-1) AND Toate Do     {si caut unul stins}
        Begin
          If NOT F[j] Then             {daca l-am gasit}
            Toate := FALSE;            {marchez}
            Inc(j)                     {trec la urmatorul}
          End;
          If Toate Then Break           {daca toate sunt aprinse termin}
        End;
      WriteLn(Fout, Operatii);         {afisez numarul de operatii}
```

Pentru cazurile specificate in enunt exista si formule care permit determinarea imediata a numarului de operatii:

- pentru n de forma 2^k numarul de operatii este $n*n-1$
- pentru n de forma 2^{k+1} numarul de operatii este $n*n-n+1$