

Sudoku på en TI-83

Her er så et lille program der kan løse en sudoku på en TI-83. Det bruger sygt meget hukommelse, så hvis du har fyldt din lommeregner godt op med alt muligt, brokker den sig nok. Kommentarer sendes til jonas@imf.au.dk.

Her er lige et par informationer om, hvorledes teksten skal forstås. -> betyder STORE. [A] betyder matricen A. L1 betyder listen L_1 . Når jeg engang får fat i et link-kabel, skal jeg nok ligge programmet ud i et bedre format.

Sudoku-løseren består af tre programmer, der skal gemmes med de navne som overskriften antyder. Man skal give den en 9×9 -matrix med de indgange man kender. Dem man ikke kender skal være nul.

1 SUDOKU

```
Input "SUDOKU-MATRIX=", [A]
```

```
{9,9}->dim([B])
```

```
{9,1}->dim([C])
```

```
{2,3,5,7,11,13,17,19,23}->L1
```

```
{1,1,1,4,4,4,7,7,7}->L2
```

```
{1}->L3
```

```
1->L
```

```
1->M
```

```
Lbl 1
```

```
Disp "NY SUDOKU"
```

```
0->P
```

```
1->R
```

```
Fill(1, [B])
```

```
For(I,1,9)
```

```
For(J,1,9)
```

```

[A] (I, J)->X
If not(X=0)
Then
prgmSUDOMASK
P+1->P
End
End
End
Disp "MASKE OPDATERET"

While R>0
0->R:9->D

Disp "TAL PLACERET:",P

For(I,1,9)
For(J,1,9)
0->C

If [A] (I, J)=0
Then

For(X,1,9)
If not([B] (I, J)/L1(X)=int([B] (I, J)/L1(x)))
Then
C+1->C
X->Q
End
End

If C=1
Then
Q->[A] (I, J)
1+P->P
1+R->R
Q->X
prgmSUDOMASK
End

If C<D
Then
C->D:I->F:J->G
End

```

```

End

If P=81
Then
Disp "DONE!","RES GEMT I [A]"
Stop
End

End
End

End

For(X,1,9)
F->I:G->J
If not([B](I,J)/L1(X)=int([B](I,J)/L1(X)))
Then
X->[A](I,J)
augment([C],[A])->[C]
augment(L3,{L+1})->L3
O->[A](I,J)
End
End

prgmSUDOTREE
Goto 1

```

2 SUDOMASK

```

For(A,1,9)
If not([B](I,A)/L1(X)=[B](I,A)/L1(X))
[B](I,A)*L1(X)->[B](I,A)

If not([B](A,J)/L1(X)=[B](A,J)/L1(X))
[B](A,J)*L1(X)->[B](A,J)
End

For(A,L2(I),L2(I)+2)
For(B,L2(J),L2(J)+2)
If not([B](A,B)/L1(X)=[B](A,B)/L1(X))
[B](A,B)*L1(X)->[B](A,B)
End

```

End

Return

3 SUDOTREE

For(N,M+1,dim(L3))

If L3(N)=L

Then

N->M

Goto 1

End

End

For (N,1,dim(L3))

If L3(N)=L+1

Then

L+1->L

N->M

Goto 1

End

End

Lbl 1

For(A,1,9)

For(B,1,9)

[C] (A,9(N-1)+1+B)->[A] (A,B)

End

End

Return