

Mamouni My Ismail
Professeur agrégé de mathématiques
Enseignant en classes de MP
CPGE My Youssef
Rabat, Maroc
mamouni.myismail@gmail.com
myprepa.ifrance.com

Corrigé abrégé Epreuve Informatique Concours Centrale Sup Elec, 2004

STUDENT > **restart:**

On commence par déclarer les écoles, le nombre de postes ouverts pour chaque école

STUDENT > **Ecoles:=[E1,E2,E3];N:=[1,2,1];candidats:=[C1,C2,C3,C4];**

Ecoles := [E1, E2, E3]

N := [1, 2, 1]

candidats := [C1, C2, C3, C4]

Puis le classement de chaque école pour les candidats

STUDENT > **E1:=[C2,C4];E2:=[C4,C3,C1,C2];E3:=[C1,C2,C3];**

E1 := [C2, C4]

E2 := [C4, C3, C1, C2]

E3 := [C1, C2, C3]

Les vœux de chaque candidat classés dans l'ordre de préférence, la 1ère case représente un numéro pour identifier le candidat

STUDENT > **C1:=[1,3,2];C2:=[2,3,2,1];C3:=[3,2,3];C4:=[4,1,2];**

C1 := [1, 3, 2]

C2 := [2, 3, 2, 1]

C3 := [3, 2, 3]

C4 := [4, 1, 2]

Pour chaque école, on va chercher les candidats à éliminer, en commençant bien sûr par le dernier classée pour cette même école. Pour cela on dénombre les candidats les mieux classés pour lesquels cette école représente le 1er choix, si ce nombre dépasse le nombre de poste ouverts par la dite école, alors notre candidat est éliminé.

```
STUDENT > choix_ecole:=proc(i) local  
           Ecole,j,Nbr,k,candidat,a,b,NewEcole;  
STUDENT > Ecole:=op(i,Ecoles):NewEcole:=NULL:  
STUDENT > for j from nops(Ecole) to 1 by -1 do  
STUDENT > Nbr:=0;  
STUDENT > for k from j-1 to 1 by -1 do  
STUDENT > candidat:=op(k,Ecole):  
STUDENT > if op(2,candidat)=i then Nbr:=Nbr+1:  
STUDENT > else
```

```

STUDENT > fi:
STUDENT > od:
STUDENT > if Nbr>=op(i,N) then print(`Le candidat éliminé est
numéro`=op(1,op(j,Ecole)));
STUDENT > else print(`Le candidat non éliminé
`=op(1,op(j,Ecole));NewEcole:=NewEcole,C[op(1,op(j,Ecol
e))]:
STUDENT > fi:
STUDENT > od:
STUDENT > print(`Le choix de l'école est
`= [NewEcole]);RETURN([NewEcole]);
STUDENT > end:

```

On applique notre petit programme pour le 1ère école

```

STUDENT > E[1]:=choix_ecole(1);

```

*Le candidat non éliminé = 4
Le candidat non éliminé = 2
Le choix de l'école est = [C₄, C₂]
E₁ := [C₄, C₂]*

On applique notre petit programme pour le 2ème école

```

STUDENT > E[2]:=choix_ecole(2);

```

*Le candidat non éliminé = 2
Le candidat non éliminé = 1
Le candidat non éliminé = 3
Le candidat non éliminé = 4
Le choix de l'école est = [C₂, C₁, C₃, C₄]
E₂ := [C₂, C₁, C₃, C₄]*

On applique notre petit programme pour le 3ème école

```

STUDENT > E[3]:=choix_ecole(3);

```

*Le candidat éliminé est numéro = 3
Le candidat éliminé est numéro = 2
Le candidat non éliminé = 1
Le choix de l'école est = [C₁]
E₃ := [C₁]*

Maintenant chaque candidat va éliminer toute école qu'elle juge inutile, c'est à dire pour la quelle il est sûr d'avoir mieux. Comment? il commence par la dernière et il vérifie si une école mieux classée va le prendre dans sa liste principale.

```

STUDENT > for i from 1 to nops(Ecoles) do
STUDENT > liste_principale[i]:= [seq(op(j,E[i]),j=1..N[i])];
STUDENT > od;

```

liste_principale₁ := [C₄]

$liste_principale_2 := [C_2, C_1]$

$liste_principale_3 := [C_1]$

[STUDENT >

[*On s'inspirant de cet exemple on inverse les role écoles-candidats pour trouver les écoles à éliminer pour chaque candidat....A vous de le faire*