

---

# TP 17 - Exercices

---

## Exercice 1

1. Taper le programme suivant et l'exécuter plusieurs fois.
2. Quelle expérience le programme permet-il de simuler ?

```
rand('seed',getdate('s'))
n=25; r=3; b=3; v=3;
for k=1:n do
    d=r+b+v
    tirage=rand()
    if tirage<r/d then disp('rouge')
        else if tirage<(r+b)/d then
            disp('blanche')
            b=b-1
        else
            disp('verte')
            v=v-1
        end
    end
end
```

## Exercice 2

Compléter les scripts pour simuler la variable aléatoire  $X$ .

1. Une urne contient 3 boules blanches et 2 boules noires. On effectue  $n = 100$  tirages avec remise et  $X$  compte le nombre de boules blanches obtenues à l'issue des tirages.

```
n=100
X=_____
for k=1:n do
    if _____ then X=_____
    end
end
disp(X)
```

2. Une urne contient 8 boules blanches et 4 boules noires. On effectue  $n = 6$  tirages sans remise et  $X$  compte le nombre de boules blanches obtenues à l'issue des tirages.

```
n=6 ; bb=8 ; bn=4
X=_____
for k=1:n do
    if _____ then X=_____ ; bb=_____
        else bn=_____
    end
end
```

```
end
disp(X)
```

3. Une urne contient initialement une boule blanche et une boule noire. On effectue  $n = 10$  tirages avec remise, sachant qu'en plus de la boule que l'on remet dans l'urne, on ajoute  $a = 2$  boules de la même couleur dans l'urne.  $X$  compte de nombre de boules blanches obtenues à l'issue des tirages
4. Une urne contient 3 boules blanches et 2 boules noires. On effectue des tirages avec remise et  $X$  compte le nombre de tirages nécessaires pour obtenir la première boule blanche.

```

X=_____
testarret=0
while _____ do
    X=_____
    if _____ then testarret=_____
    end
end
disp(X)

```

5. Une urne contient 1 boule blanche et 4 boules noires. On effectue  $n = 5$  tirages avec remise et  $X$  compte le nombre de tirages nécessaires pour obtenir la première boule blanche, et vaut 0 si on n'obtient pas de boule blanche à l'issue des 5 tirages
6. Une urne contient 3 boules blanches et 4 boules noires. On effectue des tirages sans remise et  $X$  compte le nombre de tirages nécessaires pour ne plus avoir de boules blanches dans l'urne.

```

X=0
bb=3 ; bn=4
while _____ do
    X=_____
    if _____ then _____
    else _____
    end
end
disp(X)

```

7. Une urne contient 3 boules blanches et 4 boules noires. On effectue des tirages sans remise et  $X$  compte le nombre de tirages nécessaires pour ne plus avoir de boules blanches ou de boules noires dans l'urne.

### Exercice 3

Une urne contient initialement 15 boules numérotées de 1 à 15. On effectue 10 tirages sans remise dans cette urne et on s'intéresse à la suite des numéros obtenus à l'issue de ces 10 tirages.

On décrit informatiquement le contenu initial de l'urne à l'aide d'une matrice ligne à 15 colonnes, notée  $A$ , dont les coefficients sont les numéros des boules. Après chaque tirage, on modifie la matrice  $A$  de façon à ce qu'elle contienne les numéros des boules restantes (elle perd donc une colonne à chaque étape).

1. Quelle instruction Scilab permet d'obtenir la matrice  $A$  sans taper la liste des entiers de 1 à 15 ?
2. Que fait la suite d'instructions suivantes en supposant que  $n$  est le nombre de colonnes de la matrice  $A$ .

```
tirage=grand(1,1,'uin',1,n)
disp(A(tirage))
A=[ A(:,1:tirage-1) , A(:,tirage+1:n) ]
```

3. En déduire une façon de compléter le programme suivant afin qu'il permette de simuler les 10 tirages et affiche, dans l'ordre, la liste des numéros obtenus.

```
A=... // question 1
res=[]
for k=1:10 do
    n=...
    tirage=...
    res=[res,...]
    A=[ ... , ... ]
end
disp(res)
```

4. Comment change la première ligne du programme afin qu'il simule 10 tirages sans remise dans une urne contenant initialement trois boules de chaque numéro compris entre 1 et 5 ?