# Ouvrage en forme de trapèze rectangle ou isocèle: répartition de *a* augmentations sur *n* rangs

Dans ce document, nous répartissons les augmentations sur les rangs impairs.
Nous supposons que le nombre total de rangs est pair. Pour désigner un *double rang*, utilisons la notation suivante: **1 dr = 2 rg**.

Selon le contexte, "un rang avec *a* augm." signifie:
 * un rang avec *a* augm. sur un côté (ouvrage en forme de trapèze rectangle), ou
 * un rang avec *a* augm. de chaque côté (ouvrage en forme de trapèze isocèle).

## 1. Cas $1 \leq a$ et $2a \leq n$:

#### 1.1.1 Exemple numérique: répartir 13 augmentations sur 100 rangs

Avec les 50 dr, former 13 groupes de dr.
50 divisé par 13 donne 3, reste 11. On peut donc former 11 groupes de 3+1=4 dr et 13-11=2 groupes de 3 dr.
Dans chacun des 13 groupes, un augmentation est à faire.
Pour faire le lien avec l'algorithme ci-dessous,
    n1=50=nombre de dr à répartir en groupes;
    k1=13=nombre total de groupes à former;
    r1=11=(nombre de groupes de p1=4 dr);
    s1=2=(nombre de groupes de q1=3 dr).

#### 1.1.2 Algorithme pour répartir *a* augm. sur les rangs impairs

Soit *a* augmentations à répartir sur $\frac{n}{2}$ doubles rangs. Avec ces $\frac{n}{2}$ doubles rangs, on va former *a* groupes de doubles rangs. Chaque groupe de doubles rangs contient une augmentation. Afin que chaque groupe contienne au moins un double rang, nous exigeons que $\frac{n}{2} \geq a \geq 1$, c'est-à-dire $1 \leq a$ et $2a \leq n$.

---

```
n1 = Quotient[n, 2];
k1 = a;
q1 = Quotient[n1, k1];
r1 = Mod[n1, k1];
p1 = q1 + 1;
s1 = k1 - r1;
```

---

Relations : $1 \leq k1 \leq n1$, $q1 \geq 1$, $p1 \geq 2$, $r1 + s1 = k1$, $r1 * p1 + s1 * q1 = n1$

#### 1.1.3 Procédures d'affichage

assemble(nfois, nrangs, naugm) écrit le texte "tous les nrangs rg: nfois x naugm augm.";
groupe(nrangs, naugm)  écrit le texte "nrangs rg plus haut: naugm augm.";
alterne(nfois, t1, t2) écrit le texte "répéter nfois fois [ t1, t2 ]".

---

```
assemble[nfois_, nrangs_, naugm_] :=
  Which[nfois >= 2,
   Switch[naugm,
    1, "tous les " <> ToString[nrangs] <> " rg: " <> ToString[nfois] <> " x 1 
augm.",
    2, "tous les " <> ToString[nrangs] <> " rg: " <> ToString[nfois] <> " x 1 augm. 
double",
    3, "tous les " <> ToString[nrangs] <> " rg: " <> ToString[nfois] <> " x 1 augm.
```

```
triple",
    _, "tous les " <> ToString[nrangs] <> " rg: " <> ToString[nfois] <> " x " <>
ToString[naugm] <> " augm."],
   nfois == 1,
   Switch[naugm,
    1, ToString[nrangs] <> " rg plus haut: 1 augm.",
    2, ToString[nrangs] <> " rg plus haut: 1 augm. double",
    3, ToString[nrangs] <> " rg plus haut: 1 augm. triple",
    _, ToString[nrangs] <> " rg plus haut: " <> ToString[naugm] <> " augm."]];
groupe[nrangs_, naugm_] := assemble[1, nrangs, naugm];
alterne[nfois_, t1_, t2_] := Which[
   nfois == 1, Print[t1, "; ", t2],
   nfois >= 2, Print["répéter ", ToString[nfois], " fois [ ", t1, "; ", t2, " ]"]];
```

### 1.1.4 Traitement et affichage des cas particuliers

Si r1=0, alors n1=s1*q1;
si s1=0, alors n1=r1*p1;
si r1=s1, alors n1=r1*(p1+q1);

```
Which[r1 == 0, Print[assemble[s1, 2 q1, 1]]; Exit[],
    s1 == 0, assemble[r1, 2 p1, 1]; Exit[],
    r1 == s1, alterne[r1, groupe[2 p1, 1], groupe[2 q1, 1]]; Exit[]]
```

### 1.2.1 Exemple numérique: intercaler 2 (groupes de 3 rg) parmi 11 (groupes de 4 rg)

Avec les 11 (groupes de 4 rg), former 2+1=3 assemblages de (groupes de 3 rg).
11 divisé par 3 donne 3, reste 2. On peut donc former 2 assemblages de 3+1=4 (groupes de 4 rg) et 3-2=1 assemblage de 3 (groupes de 4 rg).
Les 2 groupes de 3 rg sont à intercaler entre les 3 assemblages.
Pour faire le lien avec l'algorithme ci-dessous,

n2=11=nombre de (groupes de 4 dr);
k2=3=nombre total d'assemblages à former;
r2=2=(nombre d'assemblages de p2=4 groupes);
s2=1=(nombre d'assemblages de q2=3 groupes).

### 1.2.2 Algorithme pour intercaler *r1* groupes parmi *s1* groupes

Hypothèse : $1 \le r1 < s1$, sinon permuter $(r1, p1)$ et $(s1, q1)$ :

```
If[r1 > s1, e = r1; r1 = s1; s1 = e; e = p1; p1 = q1; q1 = e];
n2 = s1;
k2 = r1 + 1;
q2 = Quotient[n2, k2];
r2 = Mod[n2, k2];
p2 = q2 + 1;
s2 = k2 - r2;
```

Relations : $2 \le k2 \le n2$, $q2 \ge 1$, $p2 \ge 2$, $r2 + s2 = k2$, $r2 * p2 + s2 * q2 = n2$

### 1.3.1 Exemple numérique: interpréter et afficher

D'après 1.2.1, on a
    répéter 2 fois [4 groupes de 4 dr]
    répéter 1 fois [3 groupes de 4 dr]
Entre chaque assemblage, intercalons un groupe de 3 dr:
    répéter 2 fois [4 groupes de 4 dr, 1 groupe de 3 dr]
    3 groupes de 4 dr
Un dr = 2 rg
    répéter 2 fois [4 groupes de 8 rg, 1 groupe de 6 rg]
    3 groupes de 8 rg
Dans chaque groupe, un augmentation est à faire
    répéter 2 fois [4 x (8 rg avec 1 augm.), 1 x (6 rg avec 1 augm.)]
    3 x (8 rg avec 1 augm.)

### 1.3.2 Algorithme de l'affichage

```
If[s2 >= 1,
 alterne[r2, assemble[p2, 2 q1, 1], groupe[2 p1, 1]];
 alterne[s2 - 1, assemble[q2, 2 q1, 1], groupe[2 p1, 1]];
 Print[assemble[q2, 2 q1, 1]]
 ,
 alterne[r2 - 1, assemble[p2, 2 q1, 1], groupe[2 p1, 1]];
 Print[assemble[q2, 2 q1, 1]]]
```

## 2. Cas $2 \leq n \leq 2a$

### 2.1.1 Exemple numérique: répartir 12 augmentations sur 18 rangs

On a donc 9 dr. Avec les 12 augm., former 9 groupes d'augmentations.
12 divisé par 9 donne 1, reste 3. On peut donc former 3 groupes de 1+1=2 augm. et 9-3=6 groupes de 1 augm.
Chaque groupe correspond à un double rang.
Pour faire le lien avec les formules qui suivent:
    n1=12=nombre d'augmentations à répartir en groupes;
    k1=9=nombre total de groupes à former;
    r1=3=(nombre de groupes de p1=2 augm.);
    s1=6=(nombre de groupes de q1=1 augm.).

### 2.1.2 Algorithme pour répartir *a* augmentations sur un rang de *n* mailles

Soit $a$ augmentations à répartir parmi $\frac{n}{2}$ doubles rangs.Avec ces $a$ augmentations, on va former $\frac{n}{2}$ groupes d'augmentations. Chaque groupe correspond à un double rang. Afin que chaque groupe contienne au moins une augmentation, il faut que $a \geq \frac{n}{2} \geq 1$, c'est-à-dire $2 \leq n \leq 2\,a$.

```
n1 = a;
k1 = Quotient[n, 2];
q1 = Quotient[n1, k1];
r1 = Mod[n1, k1];
p1 = q1 + 1;
s1 = k1 - r1;
```

Relations : $1 \leq k1 \leq n1$ , $r1 + s1 = k1$, $r1 * p1 + s1 * q1 = n1$

### 2.1.3 Traitement et affichage des cas particuliers

Si r1=0, alors n1=s1*q1;
si s1=0, alors n1=r1*p1;
si r1=s1, alors n1=r1*(p1+q1);

```
Which[r1 == 0, Print[assemble[s1, 2, q1]]; Exit[],
    s1 == 0, Print[assemble[r1, 2, p1]]; Exit[],
    r1 == s1, alterne[r1, groupe[2, p1], groupe[2, q1]]; Exit[]]
```

- ### 2.2.1 Exemple numérique: intercaler 3 (groupes de 2 augm.) parmi 6 (groupes de 1 augm.)

Avec les 6 (groupes de 1 augm.), former 3+1=4 assemblages de (groupes de 1 augm.).
6 divisé par 4 donne 1, reste 2. On peut donc former 2 assemblages de 1+1=2 (groupes de 1 augm.) et 4-2=2 assemblages de 1 (groupe de 1 augm.).
Les 3 (groupes de 2 augm.) seront intercalés entre les 4 assemblages.
Pour faire le lien avec l'algorithme ci-dessous,
n2=6=nombre de (groupes de 1 augm.);
k2=4=nombre total d'assemblages à former;
r2=2=(nombre d'assemblages de p2=2 groupes);
s2=2=(nombre d'assemblages de q2=1 groupe).

- ### 2.2.2 Algorithme pour intercaler *r1* groupes parmi *s1* groupes

Hypothèse : $1 \leq r1 < s1$, sinon permuter (r1, p1) et (s1, q1) :

```
If[r1 > s1, e = r1; r1 = s1; s1 = e; e = p1; p1 = q1; q1 = e];
n2 = s1;
k2 = r1 + 1;
q2 = Quotient[n2, k2];
r2 = Mod[n2, k2];
p2 = q2 + 1;
s2 = k2 - r2;
```

Relations : $2 \leq k2 \leq n2$, $q2 \geq 1$, $p2 \geq 2$, $r2 + s2 = k2$, $r2 * p2 + s2 * q2 = n2$

- ### 2.3.1 Exemple numérique: interpréter et afficher

D'après 2.2.1, on a
répéter 2 fois [2 groupes de 1 augm.]
répéter 2 fois [1 groupe de 1 augm.]
Entre chaque assemblage, intercalons un groupe de 2 augm.:
répéter 2 fois [2 groupes de 1 augm., 1 groupe de 2 augm.]
répéter 1 fois [1 groupe de 1 augm., 1 groupe de 2 augm.]
1 groupe de 1 augm.
Chaque groupe est un double rang qui contient le nombre d'augmentations indiqué:
répéter 2 fois [2 x (2 rg avec 1 augm.), 1 x (2 rg avec 2 augm.)]
1 x (2 rg avec 1 augm.), 1 x (2 rg avec 2 augm.)
1 x (2 rg avec 1 augm.)

#### 2.3.2 Algorithme de l'affichage

```
If[s2 >= 1,
 alterne[r2, assemble[p2, 2, q1], groupe[2, p1]];
 alterne[s2 - 1, assemble[q2, 2, q1], groupe[2, p1]];
 Print[assemble[q2, 2, q1]]
 ,
 alterne[r2 - 1, assemble[p2, 2, q1], groupe[2, p1]];
 Print[assemble[q2, 2, q1]]]
```

## 3. Liens hypertextes

#### 3.1 Calculateur en ligne

http://www.deleze.name/antoinette/TravauxManuels/Tricot/index.html

#### 3.2 Mathématiques pour le tricot

http://www.deleze.name/marcel/culture/tricot/index.html

# Ouvrage en forme de trapèze rectangle ou isocèle: répartition de *d* diminutions sur *n* rangs

**Analogue: il suffit de remplacer "augmentation" par "diminution".**