

**Définition 6.1** - *algorithme probabiliste*

Un algorithme est dit *probabiliste* s'il effectue au moins un *choix aléatoire entraînant une variation comportementale*.

**Définition 6.2** - *algorithme déterministe*

Un algorithme est dit *déterministe* s'il est non probabiliste : son comportement est toujours le même pour une même entrée.

**Définition 6.3** - *algorithme de Las Vegas*

Un algorithme probabiliste est dit *de Las Vegas* si :

- il renvoie toujours une solution correcte ;
- son temps d'exécution est régi par une variable aléatoire.

**Définition 6.4** - *loi géométrique*

Une *loi géométrique de paramètre  $p$*  est une loi modélisant le *nombre d'essais nécessaires jusqu'au premier succès* dans une suite d'expériences indépendantes identiques. Si  $X$  suit une loi géométrique, alors :

$$\forall k \in \mathbb{N}^*, \mathbb{P}(X = k) = (1 - p)^{k-1}p$$

On a alors  $\mathbb{E}(X) = \frac{1}{p}$  et  $\mathbb{V}(X) = \frac{1-p}{p^2}$ .

**Implémentation** - *exemple d'algorithme de Las Vegas*

```
1 | int las_vegas(int* t, int taille){
2 |     while(true) {
3 |         int k = rand() % taille;
4 |         if (t[k] == 1){
5 |             return k;
6 |         }
7 |     }
8 | }
```

**Définition 6.5** - *algorithme de Monte-Carlo*

Un algorithme probabiliste est dit *de Monte-Carlo* si :

- il renvoie sous une certaine probabilité une solution correcte ;
- son temps d'exécution est constant, indépendant des choix aléatoires.