

# Contents

1	Introduction	2
---	--------------	---

**Programme 1:**

```
import numpy as np

# le nombre de page
n = 12

# Poids ou importance des pages
m = np.array([2, 1, 1, 1, 3, 1, 2, 1, 2, 1, 1, 1])

# Le nombre de liens sortants de chaque page
L = np.array([1, 1, 3, 3, 4, 10, 3, 2, 3, 1, 2, 1])

# matrice de transition
P = np.zeros((n, n))

for i in range(n):
    for j in range(n):
        if L[j] != 0:
            P[i, j] = m[j] / L[j]

# la loi initiale
initial_distribution = np.ones(n) / n

# Algorithme de puissances matricielles pour résoudre l'équation  $m = m * P$ 
iterations = 30
current_distribution = initial_distribution.copy()
for _ in range(iterations):
    current_distribution = np.dot(current_distribution, P)

# Afficher la distribution stable (distribution finale)
print("la loi stationaire:")
print(current_distribution)
```

## Chapter 1

# Introduction