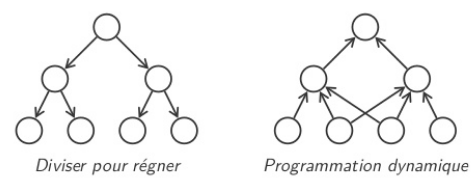
Université IBN KHALDOUN –TIARET- Faculté de Mathématiques et Informatique

Département d'informatique, année universitaire 2019/2020.

**Cours**: Structures de données avancées et algorithme

**Travaux pratiques: complexité algorithmique**

* La programmation dynamique s'appuie sur le principe d'optimalité de Bellman : une solution optimale d'un problème s'obtient en combinant des solutions optimales à des sous-problèmes.
* La méthode de programmation dynamique, comme la méthode diviser pour régner, résout des problèmes en combinant des solutions de sous-problèmes. Les algorithmes diviser-pour-régner partitionnent le problème en sous-problèmes indépendants qu’ils résolvent récursivement, puis combinent leurs solutions pour résoudre le problème initial. La méthode diviser-pour-régner est inefficace si on doit résoudre plusieurs fois le même sous-problème. Par exemple, l'algorithme suivant est inefficace :

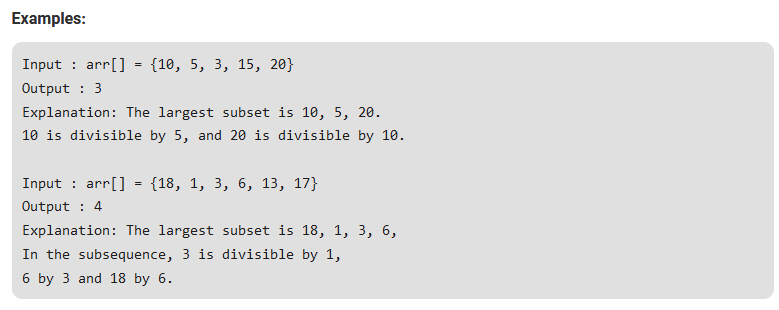


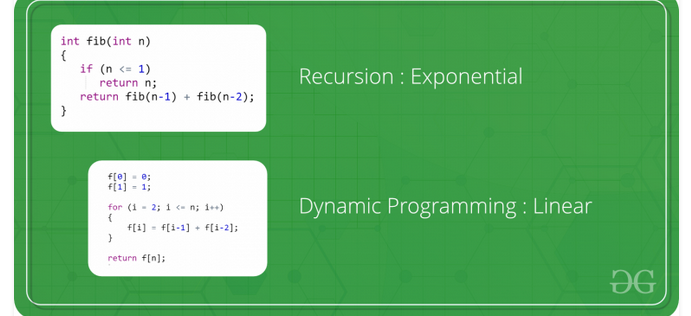
**Concevoir une procédure de programmation dynamique en quatre étapes :**

1. Définir les sous-problèmes.
2. Identifier une relation de récurrence entre les solutions des sous-problèmes.
3. En déduire un algorithme récursif avec mémorisation ou une approche du bas vers le haut
4. Résoudre le problème original à partir des solutions des sous-problèmes

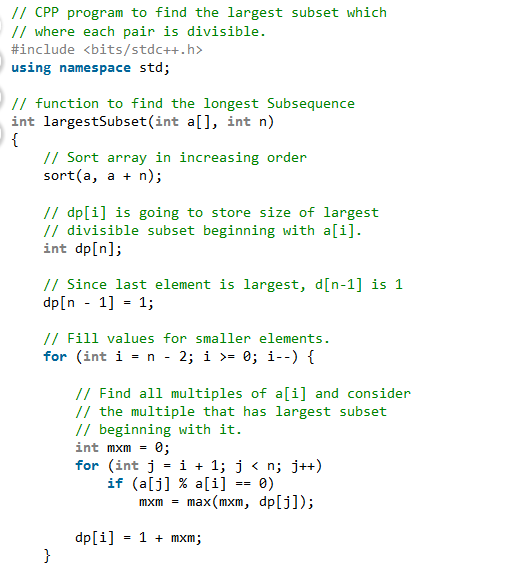
**Problème 1 : Plus grand sous-ensemble de paires divisibles ou Largest divisible pairs subset**

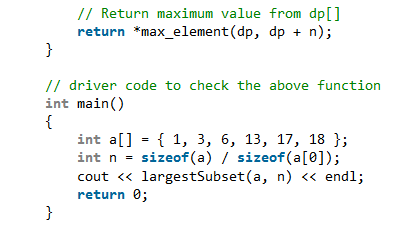
Étant donné un tableau de n éléments distincts, trouvez la longueur du plus grand sous-ensemble tel que chaque paire du sous-ensemble soit telle que l'élément le plus grand de la paire soit divisible par un élément plus petit.





Exemple de F





**Suite de Fibonacci -> programmation dymanique**

