

Exercice 1 heure

Un brin d'ADN est une séquence formée de 4 bases azotées : L'adénine (A), la cytosine (C), la guanine (G) et la thymine (T).

Exemple : **GCTATTCGAG**

Un double-brin d'ADN est constitué de 2 brins d'ADN de même longueur, parallèles et liés. L'adénine se lie toujours avec la thymine et la cytosine avec la guanine.

Exemple :



Ecrire une classe BrinADN comportant :

- Un attribut appelé `brinAdn` de type chaîne de caractères.
- Un constructeur ayant comme paramètre une chaîne de caractères et qui construit un objet de type BrinADN si cette chaîne est une séquence ADN valide.
- Une méthode `public boolean sequenceValide(string S)` qui vérifie si une séquence donnée est valide ou pas. Dans le cas où la chaîne passée au constructeur n'est pas conforme (contient un autre caractère que A, C, G et T), le constructeur lance une exception de type `SequenceException` qui doit être définie.
- Une méthode `public void sousSequence (String s)` qui vérifie si une séquence `s` donnée est une sous-séquence de la séquence `brinAdn` et affiche le résultat. Pour résoudre cette question utiliser la méthode `String substring (int beginIndex, int endIndex)` qui renvoie la sous-chaîne de caractères comprise entre `beginIndex` et `endIndex`.
- Une méthode `public void doubleBrinAdn (String s)` qui vérifie si une séquence donnée peut former un double brin d'ADN avec la séquence `brinAdn` et affiche le résultat.
- Une méthode `public double pourcentageBase (char car)` qui reçoit en paramètre un caractère représentant une des bases azotées (A, C, G ou T) et retourne le pourcentage de présence de la base dans la séquence `brinADN`. Cette méthode lance une exception de type `BaseException` dans le cas où le caractère passé en paramètre n'est pas conforme. L'exception `BaseException` doit être définie.

Exemple : dans la séquence **GCTATTCGAG** il y a 20% de cytosine

Pour tester la classe BrinADN, la classe main doit recevoir en arguments de la ligne de commande deux chaînes de caractère s_1 et s_2 et un caractère `car` et effectuer les opérations suivantes :

1. Construire un objet `brin` de type BrinADN avec la séquence s_1 .
2. Vérifier si la séquence s_2 est une sous-séquence de l'attribut `brinAdn` de l'objet `brin` construit précédemment.
3. Vérifier si la séquence s_2 peut former un double brin avec l'attribut `brinAdn` de l'objet `brin`.
4. Calculer le pourcentage de présence de la base azotée correspondant au caractère `car` donné en argument.