

Rapport SAE 2.02 d'évaluation

Adrian GUERRERO



BLAGNAC

Présentation de l'outil d'évaluation :	2
Simplicité :	3
Sobriété :	8
Efficacité :	12
Classement par catégorie :	17
Simplicité :	17
Sobriété Numérique :	17
Efficacité :	17

Présentation de l'outil d'évaluation :

Codacy :

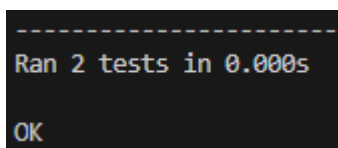
Codacy est une plateforme qui aide les développeurs à écrire du code propre, sécurisé et efficace. Elle offre une boîte à outils complète pour améliorer la qualité du code, détecter les vulnérabilités de sécurité et optimiser les performances des équipes d'ingénierie. Avec Codacy, les développeurs peuvent monitorer et appliquer des normes de codage, détecter et corriger les erreurs de sécurité, et améliorer la couverture des tests pour éviter les changements cassants.

Simplicité :

01simplicite.py : 11 (10 barème + 1 outil d'évaluation) (5)

Commentaires :

Le code fourni passe les tests initial avec les fonctions fournies, cependant il manque 2 fonctions, RLE (itération) et UnRLE (itération), ce qui pénalise le développeur parce qu'il ne passe pas tous les tests initiaux.



```
-----  
Ran 2 tests in 0.000s  
  
OK
```

Résultats des outils d'évaluations :

Codacy:

Issues 0

Your file does not have any issues.

Les résultats renvoyés par Codacy montrent que le code est propre et ne renvoie aucun Issues, ce qui rajoute un point au développeur.

28simplicite.java : 12 (10 barème + 1 analyse d'évaluation + 1 test Sup) (4)

Commentaires :

Le code fourni passe les tests initial avec les fonctions fournies, cependant il manque 2 fonctions, UnRLE et UnRLE (itération), ce qui pénalise le développeur parce qu'il ne passe pas tous les tests initiaux, mais les tests supplémentaires sont bien réussis dans un délai très bien.

Résultats Tests Supplémentaires :

```
-----  
111a111b111c  
Temps d'exécution : 2 millisecondes
```

Résultats des outils d'évaluations :

Codacy:

Issues 0 Duplication 3

Your file does not have any issues.

Les résultats renvoyés par Codacy montrent que le code est propre et ne renvoie aucun Issues, ce qui rajoute un point au développeur.

61simplicite.c : 19 (-1 erreur critique renvoyé par outils dévaluation) (3)

Commentaires :

Le code fourni passe les tests initial et les tests supplémentaires.

Résultats des outils d'évaluations :

Codacy:

Issues 9		
Filter	All categories	All levels
MINOR	Security	Does not handle strings that are not \0-terminated; if given one it may perform an over-read (it could cause a crash if unprotected) (CWE-126).
8 char* chaineFinale = (char*)malloc(strlen(text) * 2 + 1);		
MINOR	Security	The `strlen` family of functions does not handle strings that are not null terminated.
8 char* chaineFinale = (char*)malloc(strlen(text) * 2 + 1);		
MINOR	Security	Does not handle strings that are not \0-terminated; if given one it may perform an over-read (it could cause a crash if unprotected) (CWE-126).
13 for (int i = 0; i < strlen(text); i++) {		
MINOR	Security	The `strlen` family of functions does not handle strings that are not null terminated.
13 for (int i = 0; i < strlen(text); i++) {		
CRITICAL	Security	Use sprintf_s, snprintf, or vsnprintf instead.
19 sprintf(chaineFinale + indexChaineFinale, "%d%c", compteurLettre, lettre);		
MINOR	Security	Does not handle strings that are not \0-terminated; if given one it may perform an over-read (it could cause a crash if unprotected) (CWE-126).
32 if (i < strlen(text) - 1) {		
MINOR	Security	The `strlen` family of functions does not handle strings that are not null terminated.
32 if (i < strlen(text) - 1) {		
MINOR	Security	The `strlen` family of functions does not handle strings that are not null terminated.
64 int length = strlen(chaine);		
MINOR	Security	Does not handle strings that are not \0-terminated; if given one it may perform an over-read (it could cause a crash if unprotected) (CWE-126).
64 int length = strlen(chaine);		

Les résultats renvoyés par Codacy montrent que le code a 9 Issues qui sont légères la plupart, sauf 1 qui est une remarque critique à cause de l'utilisation d'une variable locale.

66simplicite.java : 20 (2)

Commentaires :

Le code fourni passe les tests initial et les tests supplémentaires sont bien réussis dans un délai raisonnable.

Résultats Tests Supplémentaires :

```
-----  
111a111b111c  
Temps d'exécution : 19 millisecondes
```

Résultats des outils d'évaluations :

Codacy:

Issues 2 Duplication 1

Filter  All categories  All levels  All patterns 

 MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
5	public static String RLE(String input) {	

 MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
26	public static String RLE(String input, int iteration) throws AlgoException{	

Les résultats renvoyés par Codacy montrent que le code est propre et ne renvoie aucun Issues grave, seulement 2 retours qui ne pénalisent pas le développeur, de convention pour Codacy.

68simplicite.java : 20 (1)

Commentaires :

Le code fourni passe les tests initial et les tests supplémentaires sont bien réussis dans un délai raisonnable, surtout le code est très bien documenté et commentés, chaque ligne explique son fonctionnement.

Résultats Tests Supplémentaires :

```
-----  
111a111b111c  
Temps d'exécution : 17 millisecondes  
-----  
abc  
Temps d'exécution : 3 millisecondes
```

Résultats des outils d'évaluations :

Codacy:

Issues 2

Filter All categories All levels All patterns

MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
22	public static String RLE(String in) {	

MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
63	public static String RLE(String in, int iteration) {	

Size

Lines of code:	146
Source lines of code:	66
Commented lines of code:	63

Les résultats renvoyés par Codacy montrent que le code est très propre, bien commentés et ne renvoie aucun Issues grave, seulement 2 retours de convention qui ne pénalisent pas le développeur.

Sobriété :

03sobriete.java : 12 (4)

Commentaires :

Le code fourni passe $\frac{3}{4}$ des tests initial, ce qui pénalise le développeur parce qu'il ne passe pas tous les tests initiaux.

Résultats des outils d'évaluations :

Codacy:

Issues 4

Filter All categories All levels All patterns

MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
13	<code>public static String RLE(String in) {</code>	
MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
53	<code>public static String RLE(String in, int iteration) throws AlgoException {</code>	
MEDIUM	Code Style	Avoid reassigning parameters such as 'in'
58	<code>in = RLE(in);</code>	
MEDIUM	Code Style	Avoid reassigning parameters such as 'in'
112	<code>in = unRLE(in);</code>	

Les résultats renvoyés par Codacy montrent que le code peut être amélioré, le code a 2 Issue moyenne à propos de l'utilisation d'une variable.

50sobriete.java : 20 (2)

Commentaires :

Le code fourni passe tous les tests initiaux, les tests supplémentaires sont bien réussis dans un délai raisonnable et le code a une structure assez intéressante.


Résultats Tests Supplémentaires :

```
-----  
111a111b111c  
Temps d'exécution : 19 millisecondes  
-----  
abc  
Temps d'exécution : 3 millisecondes
```

Résultats des outils d'évaluations :

Codacy:

Issues 4

Filter  All categories  All levels  All patterns 

 MINOR	Code Style	All classes, interfaces, enums and annotations must belong to a named package
3	public class sobriete{	
 MINOR	Code Style	The utility class name 'sobriete' doesn't match '[A-Z][a-zA-Z0-9]*'
3	public class sobriete{	
 MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
4	public static String RLE(String in){int s=in.length();String r="";for(int i=0;i<s;i++){char l=in.ch	
 MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
6	public static String RLE(String in,int iteration)throws AlgoException{if(iteration==1){return RLE(i	

Les résultats renvoyés par Codacy montrent que le code peut être amélioré, le code a 2 Issue moyenne à propos de l'utilisation d'une variable.

60sobriete.java : 20 (3)

Commentaires :

Le code fourni passe tous les tests initiaux, les tests supplémentaires sont bien réussis dans un très bon délai.

Résultats Tests Supplémentaires :

```
-----  
111a111b111c  
Temps d'exécution : 1 millisecondes  
-----  
abc  
Temps d'exécution : 0 millisecondes
```

Résultats des outils d'évaluations :

Codacy:

Issues 3

Filter All categories All levels All patterns

MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
5	public static String RLE(String in){	
MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
32	public static String RLE(String in, int iteration) throws AlgoException{	
MEDIUM	Code Style	Avoid reassigning parameters such as 'iteration'
42	iteration--;	

Les résultats renvoyés par Codacy montrent que le code est propre, le code a 1 Issue moyenne à propos de l'utilisation d'une variable.

61sobriete.py : 20 (1)

Commentaires :

Le code fourni passe les tests initial et les tests supplémentaires sont bien réussis dans un très bon délai.

```
-----  
Ran 4 tests in 0.002s  
  
OK
```

Résultats des outils d'évaluations :

Codacy:



Les résultats renvoyés par Codacy montrent que le code est propre, mais le code a une Issue moyenne à propos de l'utilisation d'une variable.

Efficacité :

11efficacite.java : 20 (2)

Commentaires :

Le code fourni passe les tests initial et les tests supplémentaires sont bien réussis dans un très bon délai.

Résultats Tests Supplémentaires :

```
-----  
111a111b111c  
Temps d'exécution : 1 millisecondes  
-----  
abc  
Temps d'exécution : 0 millisecondes
```

Résultats des outils d'évaluations :

Codacy:

Issues 2

Filter  All categories  All levels  All patterns 

	MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
4	public static String RLE(String in) {		

	MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
32	public static String RLE(String in, int iteration) throws AlgoException{		

Les résultats renvoyés par Codacy montrent que le code est propre et ne renvoie aucun Issues grave, seulement 2 retours qui ne pénalisent pas le développeur, de convention pour Codacy.

12efficacite.c : 5 (5)





Commentaires :





Le code fourni ne passe pas les tests initiaux ni arrive à retourner un objet, il y a un problème de pointeur, ce qui pénalise le développeur.

Résultats des outils d'évaluations :

Codacy:

Issues 6

Filter  All categories  All levels  All patterns 

 MINOR	Security	Does not handle strings that are not \0-terminated; if given one it may perform an over-read (it could cause a crash if unprotected) (CWE-126).	<pre>7 if (chaine == NULL strlen(chaine) == 0) {</pre>
 MINOR	Security	The `strlen` family of functions does not handle strings that are not null terminated.	<pre>7 if (chaine == NULL strlen(chaine) == 0) {</pre>
 MINOR	Security	The `strlen` family of functions does not handle strings that are not null terminated.	<pre>11 int len = strlen(chaine);</pre>
 MINOR	Security	Does not handle strings that are not \0-terminated; if given one it may perform an over-read (it could cause a crash if unprotected) (CWE-126).	<pre>11 int len = strlen(chaine);</pre>
 CRITICAL	Security	Use <code>sprintf_s</code> , <code>snprintf</code> , or <code>vsnprintf</code> instead.	<pre>21 index += sprintf(&resultat[index], "%d%c", cpt, premierCaractere);</pre>
 CRITICAL	Security	Use <code>sprintf_s</code> , <code>snprintf</code> , or <code>vsnprintf</code> instead.	<pre>34 sprintf(&resultat[index], "%d%c", cpt, premierCaractere);</pre>

Les résultats renvoyés par Codacy montrent que le code contient deux remarques critiques au niveau de la sécurité du code à cause l'utilité des variables locales, ceux qui peuvent être grave.

14efficacite.java : 20 (1)

Commentaires :

Le code fourni passe les tests initial et les tests supplémentaires sont bien réussis dans un très bon délai.





Résultats Tests Supplémentaires :




```
-----  
111a111b111c  
Temps d'exécution : 1 millisecondes  
-----  
abc  
Temps d'exécution : 1 millisecondes
```

Résultats des outils d'évaluations :

Codacy:

Issues 3

Filter  All categories  All levels  All patterns 

 MINOR	Code Style	All classes, interfaces, enums and annotations must belong to a named package
5	public class Efficacite {	
 MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
13	public static String RLE(String in) {	
 MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
34	public static String RLE(String in, int iteration) throws AlgoException {	

Size

Lines of code:	73
Source lines of code:	36
Commented lines of code:	33

Les résultats renvoyés par Codacy montrent que le code est très propre, bien commentés et ne renvoie aucun Issues grave, seulement 2 retours qui ne pénalisent pas le développeur, de convention pour Codacy.

27efficacite.java : 20 (4)

Commentaires :

Le code fourni passe les tests initial et les tests supplémentaires sont bien réussis dans un délai satisfaisant.





Résultats Tests Supplémentaires :

```
-----  
111a111b111c  
Temps d'exécution : 17 millisecondes  
-----  
abc  
Temps d'exécution : 4 millisecondes
```

Résultats des outils d'évaluations :

Codacy:

Issues 4

Filter  All categories  All levels  All patterns 

MINOR	Code Style	All classes, interfaces, enums and annotations must belong to a named package
4	public class efficacite{	
MINOR	Code Style	The utility class name 'efficacite' doesn't match '[A-Z][a-zA-Z0-9]*'
4	public class efficacite{	
MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
5	public static String RLE(String in){	
MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
21	public static String RLE(String in, int iteration) {	

Les résultats renvoyés par Codacy montrent que le code est très propre, bien commentés et ne renvoie aucun Issues grave, seulement 2 retours qui ne pénalisent pas le développeur, de convention pour Codacy.

44efficacite.java : 20 (3)

Commentaires :

Le code fourni passe les tests initial et les tests supplémentaires sont bien réussis dans un très bon délai.





Résultats Tests Supplémentaires :





```
-----  
111a111b111c  
Temps d'exécution : 1 millisecondes  
-----  
abc  
Temps d'exécution : 1 millisecondes
```

Résultats des outils d'évaluations :

Codacy:

Issues 4

Filter  All categories  All levels  All patterns 

 MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
10	<code>public static String RLE(String input) {</code>	
 MINOR	Code Style	The static method name 'RLE' doesn't match '[a-z][a-zA-Z0-9]*'
39	<code>public static String RLE(String in, int iteration) throws AlgoException {</code>	
 MEDIUM	Code Style	Avoid reassigning parameters such as 'in'
52	<code>in = RLE(in);</code>	
 MEDIUM	Code Style	Avoid reassigning parameters such as 'in'
86	<code>in = unRLE(in);</code>	

Les résultats renvoyés par Codacy montrent que le code est propre, le code a 2 Issue moyenne à propos de l'utilisation d'une variable.

Classement par catégorie :

Simplicité :

- 68simplicite.java (1)
- 66simplicite.java (2)
- 61simplicite.c (3)
- 28simplicite.java (4)
- 01simplicite.py (5)

Sobriété Numérique :

- 61sobriete.py (1)
- 50sobriete.java (2)
- 60sobriete.java (3)
- 03sobriete.java (4)

Efficacité :

- 14efficacite.java (1)
- 11efficacite.java (2)
- 44effiacite.java (3)
- 27efficacite.java (4)
- 12efficacite.c (5)