

# INFO0946 : Introduction à la Programmation

## Challenge 4 (Modules & Invariants de Boucle)

Benoit Donnet, Simon Liénardy

### 1 Énoncé du Problème

Simon souhaite disposer d'une énumération des couples de nombres premiers dits *jumeaux*. Deux nombres premiers<sup>1</sup>, avec  $n > p$ , sont dits jumeaux si  $n - p = 2$ .

Par exemple

```
3 5
5 7
11 13
17 19
29 31
41 43
59 61
71 73
101 103
107 109
```

sont des couples d'entiers premiers jumeaux. Le premier élément de chaque couple donne le nombre  $p$ , le deuxième le nombre  $n$ . Cet exemple représente aussi l'output de votre programme sur la sortie standard.

#### 1.1 Interfaces des Modules

Dans ce Challenge, nous vous demandons d'implémenter les **trois** modules suivants :

1. La fonction `est_premier(unsigned int x)` permet de déterminer si le nombre naturel  $x$  est premier ou pas. Son interface est la suivante :

```
1  /*
2  * PRÉCONDITION : /
3  * POSTCONDITION : est_premier vaut 1 si x est un nombre premier. 0 sinon.
4  */
5  int est_premier(unsigned int x);
```

2. La fonction `sont_jumeaux(unsigned int p, unsigned int n)` permet de déterminer si le couple  $(p, n)$  correspond à des nombres premiers jumeaux. Son interface est la suivante :

```
1  /*
2  * PRÉCONDITION : n > p
3  * POSTCONDITION : sont_jumeaux vaut 1 si p et n sont premiers jumeaux. 0 sinon.
4  */
5  int sont_jumeaux(unsigned int p, unsigned int n);
```

3. La procédure `affiche_jumeaux(unsigned int x)` affiche la liste des nombres premiers jumeaux entre 1 et  $x$  non compris. Son interface est la suivante :

```
1  /*
2  * PRÉCONDITION : /
3  * POSTCONDITION : Tous les nombres premiers jumeaux entre 1 et x (exclus) sont
4  *                  affichés à l'écran.
5  */
6  void affiche_jumeaux(unsigned int x);
```

---

1. Pour rappel, l'entier 1 n'est pas un nombre premier.

## 1.2 Squelette du Code

L'exemple d'output donné plus haut a été généré avec le code suivant :

```
1 #include <stdio.h>
2
3 int main() {
4     affiche_jumeaux(110);
5 } //fin programme
```

Dans ce Challenge, nous vous demandons d'implémenter les modules déclarés plus haut. Vous devez fournir uniquement le corps des modules, puisque leurs signatures sont déjà écrites pour vous dans le squelette.

Dans le corps de chacun des modules, vous pouvez définir toutes les variables que vous jugez nécessaires. Soyez néanmoins précis sur les types de ces variables et n'utilisez pas le caractère underscore (« \_ ») dans vos noms de variable.

Lors de votre soumission, vous devrez fournir le corps de chaque module ainsi que les Invariants Graphiques et les Fonctions de Terminaison nécessaires. La façon de formuler les Invariants Graphiques est indiquée dans la Sec. 3 de cet énoncé tandis que la façon de formuler une Fonction de Terminaison est décrite dans la Sec. 4. Le fichier servant de canevas à votre soumission est disponible sur [eCampus](#) (Sec. Challenges).

## 1.3 Contraintes à Respecter

Attention, pour résoudre ce problème, vous devez respecter ces contraintes. Ne pas le faire peut entrainer jusqu'à une note nulle pour ce Challenge (CAFÉ vérifiera en ce sens votre code) :

- l'instruction `return` ne peut être que la dernière instruction de vos modules. Elle ne peut pas apparaître dans le Corps de la Boucle;
- vous n'avez pas le droit d'utiliser des instructions exotiques (e.g., `break`, `continue`) vous permettant de court-circuiter une boucle;
- vous ne pouvez pas utiliser un tableau ou une structure;
- complétez correctement le squelette en ne modifiant pas les signatures qui s'y trouvent déjà;
- **le code que vous soumettez ne peut pas faire appel à une autre librairie que `stdio.h`.**

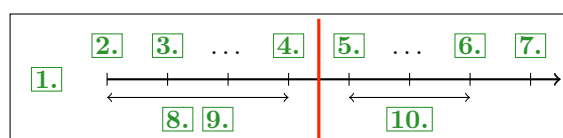
## 2 Agenda

Votre challenge doit être soumis pour le **vendredi 04/12, 20h00**, au plus tard. Pour rappel, vous disposez de maximum trois essais.

## 3 Spécifier un Invariant Graphique

### 3.1 Module `est_premier()`

Voici un Invariant Graphique muet<sup>2</sup> :



Les boîtes 2. à 7. et 9. doivent être remplacées par des constantes ou des noms de variables.

Toutes les boîtes ne doivent pas forcément être précisées : si vous indiquez que la valeur de la boîte 2.. est «  $k$  », il est implicite que la valeur de la boîte 3.. est «  $k + 1$  » : inutile alors de le préciser.

Pour les autres boîtes (1., 8. et 10.), les choix disponibles sont résumés dans le tableau 1.

---

2. Tout ce qui est dans le cadre fait partie de l'Invariant Graphique

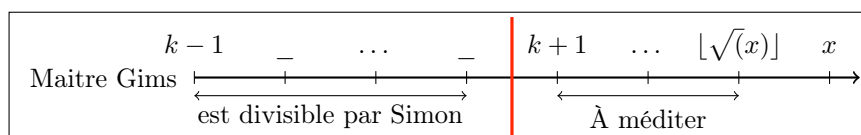
Boîte 1.	Boîtes 6. et 7.	Boîte 8.	Boîte 10.
1. $\mathbb{R}$	1. $x - 1$	1. divise	1. À afficher
2. Maitre Gims	2. $x$	2. ne divise pas	2. À multiplier
3. $\mathbb{Z}$	3. $x + 1$	3. multiplie	3. À additionner
4. ligne	4. $\log(x) - 1$	4. ne multiplie pas	4. À diviser
5. nombres	5. $\log(x)$	5. additionne	5. À parcourir
6. $\mathbb{C}$	6. $\log(x) + 1$	6. n'additionne pas	6. À méditer
7. $x$	7. $\lfloor \sqrt{(x)} \rfloor - 1$	7. est divisible par	7. À compter
8. Diviseurs	8. $\lfloor \sqrt{(x)} \rfloor$	8. n'est pas divisible par	8. À tester
9. $\mathbb{N}$	9. $\lfloor \sqrt{(x)} \rfloor + 1$	9. est facteur de	9. À énumérer

TABLE 1 – Invariant de Boucle : Possibilités de choix pour les boîtes 1., 6. à 8. et 10..

Veuillez indiquer, dans le fichier de soumission, à la réponse sur l'Invariant Graphique, le numéro de la boîte, suivi d'un point, suivi de la valeur numérique ou du nom de variable ou constante de votre choix (pour les boîtes de 2. à 5.) et 9. ou d'un nombre entre 1 et 9 correspondant à votre choix (pour les boîtes 1., 6. à 8. et 10.).

### 3.1.1 Encodage dans le Fichier de Soumission

Si vous pensez qu'un bon Invariant Graphique serait :



Alors<sup>3</sup>, complétez le fichier de réponse comme suit :

1. 2
2. k-1
3. \_
4. \_
5. k+1
6. 8
7. 2
8. 7
9. Simon
10. 6

Il est toujours possible d'affecter une variable ou une constante d'un modificateur +1 ou -1. Il suffit dans ce cas d'écrire ce +1 ou -1 comme montré ci-dessus. N'introduisez pas de parenthèses.

**Dans tous les cas,** si vous pensez que la bonne réponse consiste à ne rien écrire à l'emplacement d'une boîte, n'inscrivez rien comme réponse ou un « \_ ».

**Pour votre facilité,** (et pour les distrait-e-s qui en oublieraient un), les numéros des 10 boîtes sont déjà inscrits dans le fichier de réponse.

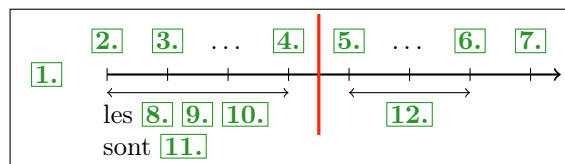
3. Vous êtes conscient que ce n'est pas la bonne réponse ?

## 3.2 Module `sont_jumeaux()`

Une brève analyse en sous-problèmes et une bonne compréhension de l'énoncé devraient vous convaincre que l'implémentation de ce module ne nécessite pas de boucle (et donc d'Invariant de Boucle). Si vous n'êtes pas convaincu, relisez attentivement cet énoncé.

## 3.3 Module `affiche_jumeaux()`

Voici un Invariant Graphique muet <sup>4</sup> :



Les boîtes **2.** à **7.** doivent être remplacées par des constantes ou des noms de variables.

Toutes les boîtes ne doivent pas forcément être précisées : si vous indiquez que la valeur de la boîte **2.** est «  $k$  », il est implicite que la valeur de la boîte **3.** est «  $k + 1$  » : inutile alors de le préciser.

Pour les autres boîtes (**1.** et **8.** à **12.**), les choix disponibles sont résumés dans le tableau 2.

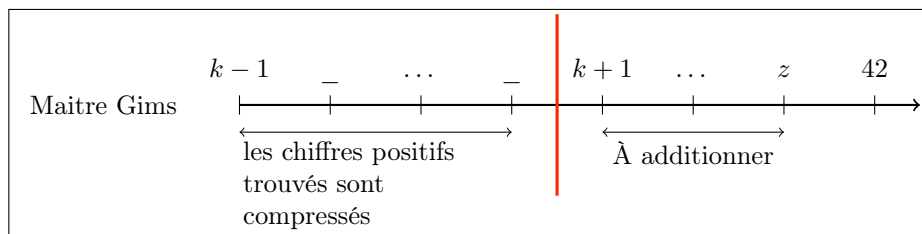
Boîte 1.	Boîte 8. et 9.	Boîte 10.	Boîte 11.	Boîte 12.
1. $\mathbb{R}$	1. mots	1. triplés	1. calculés	1. À afficher
2. Maître Gims	2. composés	2. derniers	2. affichés	2. À multiplier
3. $\mathbb{Z}$	3. nombres	3. premiers	3. compressés	3. À additionner
4. ligne	4. chiffres	4. jumeaux	4. décidés	4. À diviser
5. $\mathbb{N}$	5. positifs	5. impairs	5. effectués	5. À parcourir
6. $\mathbb{C}$	6. phrases	6. additionnés	6. parcourus	6. À méditer
7. $x$	7. premiers	7. parcourus	7. testés	7. À compter
8. Diviseurs	8. impairs	8. calculés	8. divisés	8. À tester
9. $\mathbb{N}$	9. négatifs	9. trouvés	9. énumérés	9. À énumérer

TABLE 2 – Invariant de Boucle : Possibilités de choix pour les boîtes **1.**, **8.** à **12.**.

Veuillez indiquer, dans le fichier de soumission, à la réponse sur l'Invariant Graphique, le numéro de la boîte, suivi d'un point, suivi de la valeur numérique ou du nom de variable ou constante de votre choix (pour les boîtes de **2.** à **7.**) ou d'un nombre entre 1 et 9 correspondant à votre choix (pour les boîtes **1.** et **8.** à **12.**).

### 3.3.1 Encodage dans le Fichier de Soumission

Si vous pensez qu'un bon Invariant Graphique serait :



Alors <sup>5</sup>, complétez le fichier de réponse comme suit :

4. Tout ce qui est dans le cadre fait partie de l'Invariant Graphique
5. Vous êtes conscient que ce n'est toujours pas la bonne réponse ?

```

1. 2
2. k-1
3. _
4. _
5. k+1
6. z
7. 42
8. 4
9. 5
10. 9
11. 3
12. 3

```

Il est toujours possible d'affecter une variable ou une constante d'un modificateur +1 ou -1. Il suffit dans ce cas d'écrire ce +1 ou -1 comme montré ci-dessus. N'introduisez pas de parenthèses.

**Dans tous les cas**, si vous pensez que la bonne réponse consiste à ne rien écrire à l'emplacement d'une boîte, n'inscrivez rien comme réponse ou un « \_ ».

**Pour votre facilité**, (et pour les distrait-e-s qui en oublieraient un), les numéros des 12 boîtes sont déjà inscrits dans le fichier de réponse.

## 4 Fonction de Terminaison

La Fonction de Terminaison<sup>6</sup> permet de fournir la preuve que la boucle se termine. Dans ce Challenge, nous vous demandons de fournir la Fonction de Terminaison de chacune de vos boucles (puisque 1 Invariant de Boucle == 1 boucle).

Dans le fichier de réponses, aux questions sur la Fonction de Terminaison, nous vous demandons de nous la fournir sous la forme d'une **expression C** valide.

La correction s'assurera entre autres :

- Que cette expression utilise des variables de votre code (la variable d'itération sera déduite de l'invariant correspondant !);
- Que son domaine est bien l'ensemble des Entiers (positifs si le gardien est vrai);
- Que sa valeur décroît strictement entre deux itérations.

### 4.1 Exemple

Si vous pensez que la Fonction de Terminaison de votre code, qui manipule la variable `toto` et la constante `G` devrait être :

$$t = G + 17 - \text{toto}$$

Encodez :

```
G + 17 - toto
```

N'indiquez pas « `F =` » ou « `t =` » mais seulement l'expression qui sert à calculer la valeur de votre Fonction de Terminaison. De plus, n'ajouter pas « `> 0` » à votre soumission. en effet,  $G + 17 - \text{toto} > 0$  n'est pas une fonction à valeur entière mais Booléenne<sup>7</sup>.

6. À ne confondre ni avec le Critère d'Arrêt, ni avec le Gardien de Boucle

7. Et donc c'est incorrect...

## 5 Soumettre une Archive .zip

**Remarque** À partir de maintenant, les fichiers squelettes ont pour extension « .c » de sorte que votre éditeur préféré active la coloration syntaxique correcte par défaut. Gardez en tête que ce squelette n'est pas un fichier C valide. Ne tentez donc pas de le compiler.

Pour tous les challenges, un fichier servant de canevas pour la soumission du challenge est disponible sur la page web du cours<sup>8</sup>. Le nom du fichier est `challengeX.c` où X est remplacé par le numéro du challenge. Le squelette pour ce challenge 4 est donc contenu dans le fichier `challenge4.c`. Par la suite, libre à vous de modifier le nom du fichier que vous soumettez, cela n'a pas d'importance.

Tous les challenges doivent être compressés en une archive « .zip ». Voici comment procéder sur les systèmes d'exploitation les plus courants.

**Sous Windows** Il suffit de cliquer sur le fichier à l'aide du bouton droit de la souris, sélectionner « Envoyer vers... » et sélectionner ensuite « Dossier compressé ».

**Sous Linux (Ubuntu, Fedora, Linux Mint, ...)** Il suffit de cliquer sur le fichier à l'aide du bouton droit de la souris, sélectionner « Compresser... ». Veillez bien à sélectionner « .zip » dans la liste des extensions possibles pour le fichier.

**Sous OS X** Cliquez sur le fichier en maintenant la touche Contrôle enfoncée (ou cliquez avec 2 doigts), sélectionnez « Compresser ».

**Dans tous les cas** Ne soumettez pas de fichier `.tar.gz`, `.7z`, `.rar` ou autre! C'est bien un fichier `.zip` qui est attendu. Le nom de l'archive importe peu, tant que c'est une archive zip valide, dont le nom se termine bien par « .zip » **et ne comporte pas de caractères spéciaux comme des espaces, des parenthèses, etc.**

**Si vous commettez un erreur dans la soumission,** comme par exemple :

- Donner un mauvais nom à l'archive ou utiliser des caractères inhabituels (e.g., des parenthèses);
- Soumettre deux fois d'affilée en cliquant trop rapidement;
- Mal placer les réponses dans le fichier `challenge4.txt`
- ...

C'est dommage pour vous<sup>9</sup>. Redoublez d'attention la prochaine fois! Pour autant, la plateforme de soumission et le soucis d'équité entre tous les étudiants ne permettent pas de vous octroyer une nouvelle soumission.

---

8. eCampus, Sec. Challenges.

9. Nous partageons votre peine.