

INFO0946 : Introduction à la Programmation

Challenge 1 (Instructions Itératives)

Benoit Donnet, Simon Liénardy

1 Énoncé du Problème

La population des Sims Alpha est de X d'habitants. Il s'avère qu'elle augmente de 500.000 habitants par an. À contrario, celle des Sims Bêta n'est que de Y habitants ($Y > 33$). Par contre, elle augmente de 3% par an.

Ce Challenge a pour but de déterminer le nombre d'années nécessaires pour que la population de Sims Bêta dépasse celle des Sims Alpha.

À titre d'informations, le tableau ci-dessous vous donne quelques résultats attendus :

Alpha	Bêta	Années
0	1.000.000	0
1.000.000	0	IMPOSSIBLE
1.000.000	33	IMPOSSIBLE
1.000.000	34	558
1.000.000	500.000	176

TABLE 1 – Exemples de résultats. Pensez à tester votre code dans plusieurs cas de figure.

Le squelette de votre code est le suivant :

```
1 #include <stdio.h>
2
3 int main() {
4     unsigned int population_alpha = ...;
5     unsigned int population_beta = ...;
6     unsigned int annees;
7
8     //votre code viendra ici (variables + instructions)
9
10    printf("Nombre_d'annees:_%u\n", annees);
11 } //fin programme
```

Les variables `population_alpha` et `population_beta` correspondent aux tailles initiales des populations pour les Sims Alpha (X dans le 1^{er} paragraphe) et des Sims Beta (Y dans le 1^{er} paragraphe). Vous devez considérer, lors de la soumission, que chacune de ces deux variables est déjà initialisée à une certaine valeur positive¹. La variable `annees` (non initialisée – c'est à vous de le faire dans votre solution) sera celle qui contiendra le résultat de votre calcul, suivant les règles décrites ci-dessus. Dans tous les cas, vous **ne** devez **pas** redéclarer ces variables, sous peine de faire échouer la correction automatique (ce qui mènera, forcément, à une cote nulle).

Lors de votre soumission, vous devrez fournir trois versions de votre code. La première version utilise une boucle de type `while`, la seconde une boucle de type `for`. Enfin, la dernière version se base sur une boucle de type `do ...while`. Chacune des version ne doit contenir qu'une et une seule boucle.

Pour chaque version, vous ne devez fournir que l'**extrait** de code qui remplace la ligne 8 du squelette ci-dessus : « //votre code viendra ici ». En particulier **ne** recopiez **pas** la commande `printf` de la ligne 10 du squelette.

1. `population_beta` est garantie strictement supérieure à 33.

Contraintes supplémentaires Vous ne pouvez pas utiliser de variables dont le nom débute par un underscore (i.e. « _ »). Vous ne pouvez pas utiliser les mots-clés suivants : `goto`, `break`, `continue`, `return`.

Conseil Faites bien attention aux types des variables que vous utilisez et à ce qu'elles représentent.

2 Agenda

Votre challenge doit être soumis pour le **vendredi 04/10, 18h00**, au plus tard. Pour rappel, vous disposez de maximum trois essais.

3 Soumettre une Archive .zip

Pour tous les challenges, un fichier servant de canevas pour la soumission du challenge est disponible sur la page web du cours². Le nom du fichier est `challengeX.txt` où X est remplacé par le numéro du challenge. Le squelette pour ce challenge 1 est donc contenu dans le fichier `challenge1.txt`. 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 ;
- Soumettre deux fois d'affilée en cliquant trop rapidement ;
- Mal placer les réponses dans le fichier `challenge1.txt`
- ...

C'est dommage pour vous³. 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.

2. <http://www.ecampus.ulg.ac.be>, Sec. Challenges.

3. Nous partageons votre peine.