

Devoir #1

I. Néologismes

Problème inspiré de <https://prologin.org/train/2014/semifinal/neologisme> mais modifié de la manière suivante :

- En entrée, on n'a pas besoin des tailles des mots¹.
- En sortie, afficher OUI si le mot de la première ligne est un néologisme, NON sinon.
- Exemples d'entrées/sorties :

```
$ g++ neologisme.cpp
$ ./a.out < neologisme.1in
OUI
$ ./a.out < neologisme.2in
NON
```

II. Cybersécurité

On fait appel à vos cyber-talents pour vérifier si des mots de passe sont sécuritaires. Un mot de passe est considéré sécuritaire s'il satisfait *toutes* les conditions suivantes :

- il a une longueur d'au moins 8 caractères,
- il contient au moins une lettre minuscule (`[a-z]`),
- il contient au moins une lettre majuscule (`[A-Z]`),
- il contient au moins un chiffre (`[0-9]`),
- il contient au moins un caractère spécial parmi :

```
!#$%&*+~/. : , ; = ? @ [ ] ^ _ { | } ~
```

En entrée, vous trouverez sur la première ligne un entier n indiquant le nombre de mots de passe à vérifier. Puis sur chacune des n lignes suivantes, un mot de passe à vérifier.

En sortie, votre programme doit afficher un seul entier correspondant au nombre de mots de passe sécuritaires.

1. Dans des langages bas niveau tels que le C, il est nécessaire de connaître à l'avance la taille [maximale] d'une chaîne de caractères de type `char []`.

Exemple d'entrée :

8
12345678
Mot_de_passe
Mot=passe123
1c\$9qC%#
VLnU-7=
UdeM_2022
Universite_de_Moncton
U-de-M@2022

Sortie :

4

En effet, seuls les quatre mots de passe

Mot=passe123
1c\$9qC%#
UdeM_2022
U-de-M@2022

satisfont chacune des conditions décrites plus haut.