

7 recherche dans une liste d'étudiant

On suppose dans cet exercice que les dates sont représentées par des objets de la classe `Date` et que cette classe définit, comme dans le tp, les méthodes permettant les comparaisons de deux dates. Le constructeur de cette date prend en paramètre trois entiers (`jour: int`, `mois: int`, `annee: int`) et initialise les attributs de même nom.

On dispose également une classe `Etudiant` dont le constructeur initialise les attributs `nip: int`, `nom: str`, `prenom: str`, `naissance: Date` et une variable `etudiants` contenant une liste d'étudiants :

```
etudiants = [Etudiant(99998132, "Calbuth", "Raymond", Date(12, 12, 1987)),
              Etudiant(99994451, "Talon", "Achille", Date(7, 11, 1963)),
              Etudiant(99996348, "Calbuth", "Monique", Date(29, 7, 1987)),
              Etudiant(99995433, "Blanc-Sec", "Adèle", Date(17, 4, 1976)),
              Etudiant(99997674, "Brisefer", "Benoit", Date(15, 12, 1960)),
              Etudiant(99998324, "Lagaffe", "Gaston", Date(28, 2, 1957))]
```

1. en utilisant la fonction `recherche_seq` vue en cours et sans la modifier, expliquer comment savoir si, oui ou non, il existe :

- un étudiant né le 29 janvier 1965 ;
- un étudiant né en 1960 ;
- un étudiant dont l'anniversaire est aujourd'hui ;
- un étudiant dont le prénom est Timoléon ;
- un étudiant dont le nom commence par Cal ;
- un étudiant dont le prénom contient exactement un caractère a.

Pour chaque cas, vous donnerez la fonction de comparaison.

2. Pour chacun des points précédents, dire si la recherche est fructueuse ou non sur l'exemple et combien d'appel à la fonction de comparaison ont eu lieu.

1. `recherche_seq(Etudiant(0, "", "", Date(29, 01, 1965)), etudiants,`
`lambda x, y: x.naissance == y.naissance)` → Pas fructueux :c

2. `recherche_seq(Etudiant(0, "", "", Date(01, 01, 1960)), etudiants,`
`lambda x, y: x.naissance.annee == y.naissance.annee)` → fructueux.

3. `recherche_seq(Etudiant(0, "", "", Date(05, 03, 1900)), etudiants,`
`lambda x, y: x.naissance.jour == y.naissance.jour and`
`x.naissance.mois == y.naissance.mois)` → :c

4.

