

--	--

# MITST04 : Évaluation finale – Partie 2

samedi 14 mai 2022, durée 2 heures

## Important

- Le code source sera déposé dans Moodle à la fin de l'évaluation.
- Aucun document n'est autorisé, sauf l'emploi des pages de manuel (man).
- L'énoncé est à respecter le plus scrupuleusement possible.
- **ATTENTION** : Les jeux d'essais qui seront utilisés par l'enseignant pour noter vos programmes seront différents de l'exemple fourni.

## Cahier des charges

Écrire un programme C nommé `chercheNumbers` dans lequel

1. Le processus principal (le père), créera un processus fils.
2. Le processus père ouvrira un fichier texte en lecture. On supposera que le fichier contient des nombres entiers (type `int`), à raison d'un nombre par ligne. Ce fichier jouera le rôle de dictionnaire.
3. Le processus fils enverra des nombres entiers (`int`) au processus père. Les nombres seront lus sur l'entrée standard.
4. Le processus père recevra les nombres et en recherchera l'occurrence dans son fichier dictionnaire.
5. Si le nombre est trouvé dans le fichier, le père affichera « OK », « NOK » sinon.
6. Modifier le programme pour que, au lieu d'afficher lui même le résultat de la recherche, le père envoie « OK » ou « NOK » à son fils.
7. Lorsque EOF sera reçu, le fils affichera les statistiques : quantité de nombres trouvés dans le dictionnaire sur nombre de recherches effectuées.

## Indications :

- Les questions 4 et suivantes peuvent être vues comme des raffinements. Selon les possibilités de chacun, il est déjà possible d'écrire un programme qui réponde aux questions 1, puis 2, puis 3, ... et plus si possible.
- La recherche d'un nombre dans un fichier pourra être avantageusement confiée à une fonction dédiée que l'on écrira.
- Initialement, le fichier ouvert par le père pourra être défini en dur par une chaîne de caractères constante, puis si le temps le permet, on pourra en faire un argument de la ligne de commande.
- Idem pour les nombres soumis au père par le fils, on pourra initialement les lire au clavier puis ensuite procéder à une redirection de l'entrée standard pour les lire dans un fichier de test, comportant un nombre par ligne.
- Un jeu d'essai comprenant deux fichiers est fourni dans Moodle (juste en-dessous de l'énoncé : `dico.txt`, `cherche.txt`). Le résultat de l'affichage du programme lorsqu'il sera exécuté avec ce jeu d'essai devra être le plus proche possible de la sortie reproduite ci-dessous. Au final, les sorties d'affichage devraient être produites par le fils, sur la sortie d'erreur (`stderr`), bien qu'au début on puisse se contenter de faire afficher le résultat par le père.

➤ ./cherche

Fils envoie : 90

Père lit : 90

Père répond : NOK

Fils envoie : 98

Père lit : 98

Père répond : NOK

Fils envoie : 50

Père lit : 50

Père répond : OK

Fils envoie : 132

Père lit : 132

Père répond : NOK

Fils envoie : 890

Père lit : 890

Père répond : NOK

Fils envoie : 20

Père lit : 20

Père répond : OK

2 nombres trouvés sur 6 soumis