

Contrôle de Info C

Durée : 1h30

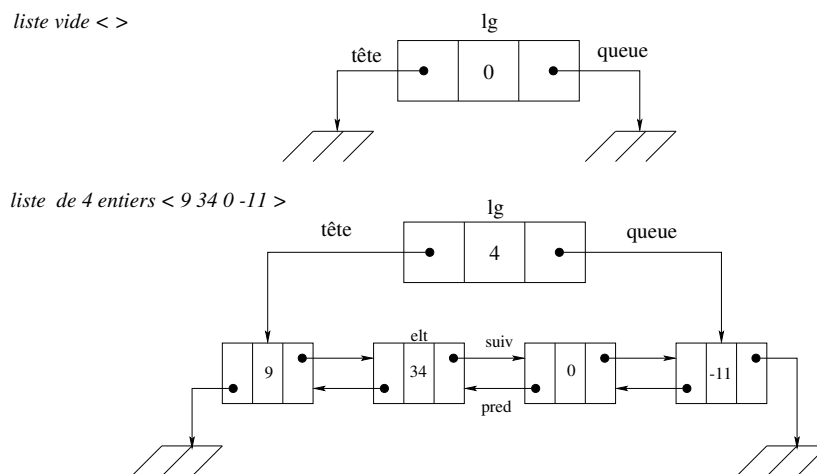
Aucun document autorisé

Nom :

Prénom :

On souhaite mettre en œuvre la notion de *liste linéaire* (telle que vue en TD) à l'aide d'une **structure dynamique doublement chaînée**. Les éléments de la liste sont placés dans nœuds reliés par des pointeurs.

Le schéma donné ci-dessous, montre, par exemple, l'organisation d'une liste vide $< >$, et de la liste d'entiers $< 9, 34, 0, 11 >$.



- 1. À partir du schéma précédent, déclarez les types `liste`, `noeud`, et `pnoeud` (pointeur sur `noeud`) du fichier `liste.h`. Dans la suite de ce DS, vous utiliserez ces déclarations de type.

```
#pragma once;
typedef int T;
typedef ... pnoeud;
typedef ... liste;
```

- 2. Écrivez les fonctions `listeVide` et `longueur` qui renvoient, respectivement, une liste vide et la longueur d'une liste. Ces fonctions ont les en-têtes suivants :

liste listeVide(void) et int longueur(const liste l).

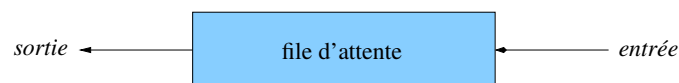
This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

- 3. Écrivez la fonction `ième` qui renvoie l'élément de rang `r` d'une liste `l`. Le rang est compris entre 1 et `longueur(l)`.

[illegible]

- [illegible]

- [illegible]



Sur une file de type `file`, on définit les fonctions et procédures suivantes :

1. `file fileVide(void)` //renvoie une file vide
2. `bool estVide(const file f)` //teste si la file vide est vide ou non
3. `T premier(const file f)` //renvoie l'élément en tête de file
4. `void enfiler(file *f, const T x)` //ajoute un élément en queue de file
5. `void défiler(file *f)` //supprime l'élément en tête file

-
- This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

8. Dans une fonction `main` :

- déclarez et initialisez une file générique `f` ;
- enfiler successivement la chaîne de caractères `"bonjour"` et les réels `double` 12.99 et `-3.14` ;
- affichez le premier élément de la file `f` et défilez `f` ;
- affichez le premier élément de la file `f` et défilez `f` ;

Vous pouvez écrire des fonctions auxiliaires.

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Un fichier (**PAS UN FICHIER DE TEXTE**) contient une suite de réels **double**. La suite est de longueur quelconque, éventuellement vide.

- 9. Écrivez la fonction `fileDeDoubles` qui lit un fichier de réels **double** et renvoie une file d'attente contenant les réels **double** contenus dans fichier. Les éléments de la file d'attente restent déclarés de type **void ***. L'en-tête de la fonction est le suivant :

```
file fileDeDoubles(const char *nomf)
```

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

-
- This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.