

Introduction à l'informatique II (CSI 1501)

Professeur: Marcel Turcotte

Date: 3 mars 2001, heure: 10h00 à 12h00, salle: ART 026

Identification

Nom: _____ prénom: _____

Numéro d'étudiant: _____

Consignes

1. Livres fermés,
2. Sans calculatrice ou toute autre forme d'aide,
3. Répondez à toutes les questions,
4. Répondez sur ce questionnaire, utilisez le verso des pages si nécessaire, mais vous ne pouvez remettre aucune page additionnelle.
5. Ne retirez pas l'agrafe.

Barème

Question 1 (30 points)

(a) Écrivez une interface, `PairInterface`, pour une classe pouvant contenir une paire de chaînes de caractères (type *String*). L'interface doit contenir les méthodes suivantes:

- `equals (PairInterface p)`
- `getFirst ()`
- `getSecond ()`
- `swap ()`

(b) Voici une implémentation partielle de la class *Pair*. Il s'agit d'une implémentation chaînée; avec une classe intérieure, *Node*, comme toutes nos autre implémentations chaînée. Cette classe représente une paire à l'aide d'une liste chaînée de longueur 2. Une variable d'instance, *first*, pointe sur le noeud contenant le premier élément de la paire, et ce noeud point vers un autre qui contient la seconde valeur de la paire.

```
class Pair implements PairInterface {
    private class Node {
        String value;
        Node next;
    }
    private Node first;
}
```

Complétez cette classe de sorte qu'elle implémente l'interface *PairInterface*. Vous ne devez ajouter aucune variable d'instance.

En plus, vous devez écrire un constructeur pour la classe possédant 2 paramètres, tous deux de type *String*, qui sont utilisés afin d'initialiser la première et la seconde valeur de la paire.

Notez que la méthode *swap()* doit échanger les références et non les valeurs.

Page blanche.

Question 2 (20 points)

Étant donné la classe *LinkedStack*, écrivez une méthode d'instance, *length*, telle que *s.length()* retourne le nombre d'éléments de la pile *s*.

Question 3 (20 points)

Écrivez une méthode de classe (*static*) ayant un seul paramètre, de type *QueueInterface* (une l'interface pour une file). La méthode renverse l'ordre des éléments de la file. Les éléments de la file sont de type *String*. Vous pouvez assumer l'existence d'une classe *Stack*, qui implémente *StackInterface*, et une classe *Queue*, qui implémente *QueueInterface*.

Question 4 (28 points)

(a) Expliquez la différence entre,

```
String s = null;
```

et,

```
String s = "";
```

(b) Étant donné la méthode suivante,

```
public static void meth (String s) {  
    s = "NEW";  
}
```

quelle est la sortie produite par les énoncés suivants:

```
String x = "OLD";  
meth (x);  
System.out.println (x);
```

(c) Évaluez l'expression postfixe (RPN — *reverse polish notation*) suivante ...

(d) Traduisez l'expression infixe suivante ... en notation postfixe (RPN).