Examen: Structures de données en C

Durée: 1h30

Consignes.

- Aucune calculatrice nécessaire.
- Réponses courtes et lisibles.
- Le barème est indiqué par question (total: 20 points).

Exercice 1 — QCM et vrai/faux (6 points)

1.1 QCM (3 points) Cochez la bonne réponse.

1. L'opération `pop` sur une pile retire:
   a) l'element le plus ancien
   b) l'element le plus récent
   c) un element aléatoire
2. Une file circulaire avec un tableau de taille N utilise le modulo pour:
   a) trier les elements
   b) recycler les indices
   c) tester l'egalite
3. Le temps moyen de `lookup` dans une map implementee par liste est:
   a) O(1)
   b) O(log n)
   c) O(n)

1.2 Vrai/Faux (3 points) Indiquez V ou F.

1. Dans un BST, tous les éléments du sous-arbre gauche sont strictement inférieurs à la racine. _____
2. Une liste chaînée permet un accès direct en O(1) à l'élément d'indice k. _____
3. Une pile est suffisante pour évaluer une expression postfixe. _____

Exercice 2 — Traces et simulation (4 points)

2.1 Pile (2 points) Pile vide. On effectue: `push(4)`, `push(7)`, `pop()`, `push(2)`, `top()`.

- Donnez la valeur renvoyee par `pop()`.
- Donnez la valeur renvoyee par `top()`.

2.2 File (2 points) File vide. On effectue: `enqueue(1)`, `enqueue(2)`, `dequeue()`, `enqueue(3)`, `dequeue()`.

- Donnez la 1re valeur retirée.
- Donnez la 2e valeur retirée.

Exercice 3 — Compléter du code (6 points)

3.1 Pile (2 points) Compléter `stack_top`.

```
int stack_top(const Stack *s, int *out) {
  if (/* ??? */) return 0;
  /* ??? */
  return 1;
}
```

3.2 File (2 points) Compléter `queue_dequeue`.

```
int queue_dequeue(Queue *q, int *out) {
  if (/* ??? */) return 0;
  /* ??? */
  /* ??? */
  return 1;
}
```

3.3 Liste (2 points) Compléter l'insertion en tete (version pointeurs).

```
Node *list_push_front(Node *head, int value) {
  Node *n = malloc(sizeof(Node));
  if (!n) return head;
  /* ??? */
  /* ??? */
  return n;
}
```

Exercice 4 — Analyse de complexité (2 points)

Donnez la complexité temporelle (en O) :

1. `push` sur une pile par tableau. _____
2. `lookup` dans une map par liste de paires. _____

Exercice 5 — Conception rapide (2 points)

On veut traiter des impressions selon leur ordre d'arrivée, avec un temps d'attente acceptable.

- Quelle structure choisir ? _____
- Justifiez en une phrase. _____

Exercice 6 — Arbre binaire de recherche (2 points)

Insérer les valeurs suivantes dans un BST vide: `8, 3, 10, 1, 6, 14`.

- Dessinez l'arbre final (schéma simple).
- Donnez le parcours infixe.

Barème total: 20 points. Bon courage!