

# Corrigé: Structures de données en C

Examen 1h30

## Exercice 1 — QCM et vrai/faux

### 1.1 QCM

1. b) l'élément le plus récent.
2. b) recycler les indices.
3. c)  $O(n)$ .

### 1.2 Vrai/Faux

1. V
2. F
3. V

## Exercice 2 — Traces et simulation

### 2.1 Pile

- `pop()` renvoie 7.
- `top()` renvoie 2.

### 2.2 File

- 1re valeur retirée: 1.
- 2e valeur retirée: 2.

## Exercice 3 — Compléter du code

### 3.1 Pile

```
int stack_top(const Stack *s, int *out) {
    if (s->top < 0) return 0;
    *out = s->data[s->top];
    return 1;
}
```

### 3.2 File

```
int queue_dequeue(Queue *q, int *out) {
    if (q->size == 0) return 0;
    *out = q->data[q->head];
    q->head = (q->head + 1) % QUEUE_MAX;
    q->size--;
    return 1;
}
```

### 3.3 Liste

```

Node *list_push_front(Node *head, int value) {
    Node *n = malloc(sizeof(Node));
    if (!n) return head;
    n->value = value;
    n->next = head;
    return n;
}

```

## Exercice 4 — Analyse de complexité

1. push sur pile tableau:  $O(1)$ .
2. lookup map par liste:  $O(n)$ .

## Exercice 5 — Conception rapide

- Structure: file (FIFO).
- Justification: ordre d'arrivée = ordre de traitement.

## Exercice 6 — Arbre binaire de recherche

Arbre final:

```

      8
     / \
    3  10
   / \  \
  1  6  14

```

Parcours infixe: 1, 3, 6, 8, 10, 14.