

Manipulation de bits en Java

La manipulation de bits utilise des opérateurs binaires pour résoudre des problèmes de manière efficace. Explorons la recherche d'un seul nombre et la vérification si un nombre est une puissance de 2.

1. Un seul nombre : la magie du XOR

Trouver le seul nombre non répété dans un tableau où tous les autres apparaissent deux fois en utilisant XOR.

Implémentation en Java

```
public class SingleNumber {  
    public static int singleNumber(int[] nums) {  
        int result = 0;  
        for (int num : nums) result ^= num;  
        return result;  
    }  
  
    public static void main(String[] args) {  
        int[] nums = {2, 2, 1, 1, 4};  
        System.out.println("Single number: " + singleNumber(nums));  
    }  
}
```

Sortie : Single number: 4

2. Puissance de 2 : Vérification des bits

Un nombre est une puissance de 2 s'il a exactement un bit 1 dans sa représentation binaire.

Implémentation en Java

```
public class PowerOfTwo {  
    public static boolean isPowerOfTwo(int n) {  
        return n > 0 && (n & (n - 1)) == 0;  
    }  
  
    public static void main(String[] args) {  
        int n = 16;
```

```

        System.out.println(n + " is power of 2: " + isPowerOfTwo(n));
        n = 18;
        System.out.println(n + " is power of 2: " + isPowerOfTwo(n));
    }
}

```

Sortie :

16 is power of 2: true
18 is power of 2: false

Conclusion à travers les blogs

- **Triage** : Quicksort, merge sort et bubble sort couvrent la vitesse, la stabilité et la simplicité.
- **Chaînes de caractères** : KMP, tries et Rabin-Karp gèrent la correspondance de motifs et les recherches de préfixes.
- **Recherche** : Les recherches binaires et linéaires offrent des solutions logarithmiques et linéaires.
- **Graphes** : DFS, BFS, Dijkstra et Kruskal traitent les traversées, les chemins et les MST.
- **Manipulation de bits** : XOR et vérifications de bits résolvent des problèmes complexes avec un minimum d'opérations.

Chaque catégorie a ses forces —choisissez le bon outil pour votre problème et expérimitez avec ces implémentations ! Faites-moi savoir si vous souhaitez plus de profondeur ou des algorithmes supplémentaires dans l'une des catégories.