

Pares e Ímpares

Nomes: Verina Hani Mekhail Wadie RA: 23.01266-8

João Pedro de Souza Cruz RA: 23.00057-0

The screenshot shows the Beecrowd website interface. At the top, there's a navigation bar with links: HOME, PERFIL, NEWS, OPORTUNIDADES, ACADEMIC, CONTESTS, PROBLEMAS, SUBMISSÕES, RANKS, and SAIR. The user profile is 'Hi, verinawadie1' with email 'verinawadie1@gmail.com'. The main content area is divided into a sidebar and a main panel. The sidebar contains links: AO VIVO (O que os outros estão resolvendo), LISTAR (Liste todas as suas submissões), TENTADO (Problemas ainda não resolvidos), FAQs (Precisa de ajuda?), and RESPOSTAS (O que isso significa?). The main panel shows the 'CÓDIGO FONTE' section for submission #39738783. It details the problem (1259 - Pares e Ímpares), the response (Accepted), the language (Java 19), the time (2.761s), the size (2.59 KB), and the submission date (16/05/2024 09:45:44). Below this, the source code is displayed in a monospace font.

```
1 import java.util.Scanner;
2
3 public class Main{
4
5     public int partition_par(int p, int r, int v[]){
6         int x = v[r];
7         int i = p-1;
8         for(int j = p; j < r; j++){
9             if(v[j] <= x){
10                 i = i + 1;
11                 int aux = v[i];
12                 v[i] = v[j];
13                 v[j] = aux;
14             }
15         }
16         i = i + 1;
17         v[r] = v[i];
18         v[i] = x;
19         return i;
20     }
21 }
```

```
import java.util.Scanner;

public class Bee{

    public int partition_par(int p, int r, int v[]){
        int x = v[r];
        int i = p-1;
        for(int j = p; j < r; j++){
            if(v[j] <= x){
                i = i + 1;
                int aux = v[i];
                v[i] = v[j];
                v[j] = aux;
            }
        }
        i = i + 1;
        v[r] = v[i];
        v[i] = x;
        return i;
    }
}
```

```

public void quicksort_par(int p, int r, int v[]){
    if(p < r){
        int q = partition_par(p, r, v);
        quicksort_par(p, q - 1, v);
        quicksort_par(q + 1, r, v);
    }
}

public int partition_impar(int p, int r, int v[]){
    int x = v[r];
    int i = p-1;
    for(int j = p; j < r; j++){
        if(v[j] >= x){
            i = i + 1;
            int aux = v[i];
            v[i] = v[j];
            v[j] = aux;
        }
    }
    i = i + 1;
    v[r] = v[i];
    v[i] = x;
    return i;
}

public void quicksort_impar(int p, int r, int v[]){
    if(p < r){
        int q = partition_impar(p, r, v);
        quicksort_impar(p, q - 1, v);
        quicksort_impar(q + 1, r, v);
    }
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    int N = sc.nextInt();
    int[] vetor = new int[N];
    int cont_par = 0;
    int cont_impar = 0;

    for(int i = 0; i < N; i++){
        vetor[i] = sc.nextInt();
    }
}

```

```

        if(vetor[i] % 2 == 0){
            cont_par++;
        }else{
            cont_impar++;
        }
    }

    int[] vetor_par = new int[cont_par];
    int[] vetor_impar = new int[cont_impar];

    int index_par = 0;
    int index_impar = 0;

    for(int i = 0; i < N; i++){
        if (vetor[i] % 2 == 0 ){
            vetor_par[index_par++] = vetor[i];
        }
    }

    Bee bee = new Bee();
    bee.quicksort_par(0, index_par - 1, vetor_par);

    for(int i = 0; i < N; i++){
        if (vetor[i] % 2 == 1 ){
            vetor_impar[index_impar++] = vetor[i];
        }
    }

    bee.quicksort_impar(0, index_impar - 1, vetor_impar);

    for(int i = 0; i < cont_par; i++){

        System.out.println(vetor_par[i]);
    }

    for(int i = 0; i < cont_impar; i++){
        System.out.println(vetor_impar[i]);
    }
}
}

```

