

## 篮球比赛

### 题目描述：

篮球（5V5）比赛中，每个球员拥有一个战斗力，每个队伍的所有球员战斗力之和为该队伍的总体战斗力。现有 10 个球员准备分为两队进行训练赛，教练希望 2 个队伍的战斗力差值能够尽可能的小，以达到最佳训练效果。给出 10 个球员的战斗力，如果你是教练，你该如何分队，才能达到最佳训练效果？请输出该分队方案下的最小战斗力差值。

### 输入描述：

10 个篮球队员的战斗力（整数，范围[1,10000]），战斗力之间用空格分隔，如：10 9 8 7 6 5 4 3 2 1

不需要考虑异常输入的场景。

### 输出描述：

最小的战斗力差值，如：1

### 示例 1

#### 输入：

10 9 8 7 6 5 4 3 2 1

#### 输出：

1

#### 说明：

1 2 5 9 10 分为一队，3 4 6 7 8 分为一队，两队战斗力之差最小，输出差值 1。备注：球员分队方案不唯一，但最小战斗力差值固定是 1

```
import java.util.HashSet;
```

```
import java.util.Scanner;
```

```
import java.util.Set;
```

```
// 注意类名必须为 Main, 不要有任何 package xxx 信息
```

```

public class Main {

    static Set<Integer> set = new HashSet<>();

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int[] arr = new int[10];

        int i = 0;

        int total = 0;

        while(sc.hasNextInt()){

            arr[i] = sc.nextInt();

            total += arr[i];

            i++;

        }

        select5(arr,0,0);

        int min = Integer.MAX_VALUE;

        for(int num : set){

            min = Math.min(min, Math.abs(total-2*num));

        }

        System.out.println(min);

    }
}

```

```
public static void select5(int[] arr, int sum, int cnt){  
  
    if(cnt == 5){  
  
        set.add(sum);  
  
        return;  
  
    }  
  
    for(int i = 0; i < arr.length; i++){  
  
        int num = arr[i];  
  
        if(num != 0){  
  
            arr[i] = 0;  
  
            select5(arr, sum+num, cnt+1);  
  
            arr[i] = num;  
  
        }  
  
    }  
  
}  
  
}
```