Problem 1200: Minimum Absolute Difference

Problem Information

Difficulty: Easy
Acceptance Rate: 71.00%
Paid Only: No
Tags: Array, Sorting

Problem Description

Given an array of **distinct** integers `arr`, find all pairs of elements with the minimum absolute difference of any two elements.

Return a list of pairs in ascending order(with respect to pairs), each pair `[a, b]` follows

* `a, b` are from `arr` * `a < b` * `b - a` equals to the minimum absolute difference of any two elements in `arr`

Example 1:

Input: arr = [4,2,1,3] **Output:** [[1,2],[2,3],[3,4]] **Explanation:** The minimum absolute difference is 1. List all pairs with difference equal to 1 in ascending order.

Example 2:

Input: arr = [1,3,6,10,15] **Output:** [[1,3]]

Example 3:

Input: arr = [3,8,-10,23,19,-4,-14,27] **Output:** [[-14,-10],[19,23],[23,27]]

Constraints:

* `2 <= arr.length <= 105` * `-106 <= arr[i] <= 106`

Code Snippets

C++:

```
class Solution {
public:
vector<vector<int>> minimumAbsDifference(vector<int>& arr) {

}
};
```

Java:

```
class Solution {
public List<List<Integer>> minimumAbsDifference(int[] arr) {

}
}
```

Python3:

```
class Solution:
def minimumAbsDifference(self, arr: List[int]) -> List[List[int]]:
```