class SearchSuggestionSystem {

constructor(products) {

this.products = products.sort();

}

getSuggestions(searchWord) {

const suggestions = [];

let prefix = "";

let start = 0;

let products = this.products;

for (let char of searchWord) {

prefix += char;

// Using binary search to find first match for prefix

start = this.\_lowerBound(products, prefix, start);

const list = [];

for (let i = start; i < Math.min(start + 3, products.length); i++) {

if (products[i].startsWith(prefix)) {

list.push(products[i]);

} else {

break;

}

}

suggestions.push(list);

}

return suggestions;

}

\_lowerBound(arr, prefix, start) {

let left = start, right = arr.length;

while (left < right) {

let mid = Math.floor((left + right) / 2);

if (arr[mid] < prefix) {

left = mid + 1;

} else {

right = mid;

}

}

return left;

}

}