Maximum sum of increasing order elements from n arrays

```
// CPP program to find maximum sum
// by selecting a element from n arrays
#include <bits/stdc++.h>
#define M 4
using namespace std;

// To calculate maximum sum by
// selecting element from each array
int maximumSum(int a[][M], int n) {

// Sort each array
for (int i = 0; i < n; i++)
  sort(a[i], a[i] + M);

// Store maximum element
// of last array
int sum = a[n - 1][M - 1];
int prev = a[n - 1][M - 1];
int i, j;

// Selecting maximum element from
// previoulsy selected element
for (i = n - 2; i >= 0; i--) {
  for (j = M - 1; j >= 0; j--) {
  if (a[i][j] < prev) {
    prev = a[i][j];
    sum += prev;
    break;
  }
  }

  // j = -1 means no element is
  // found in a[i] so return 0
  if (j == -1)
  return 0;
}

return sum;
}

// Driver program to test maximumSum
int main() {
int arr[][M] = {{1, 7, 3, 4},
        {4, 2, 5, 1},
        {9, 5, 1, 8}};
int n = sizeof(arr) / sizeof(arr[0]);
cout << maximumSum(arr, n);
```

```
return 0;
}
```