sum of two elements closest to zero

```
#include <stdio.h>
#include <limits.h>
int swap(int *a, int *b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
}

int partition(int * arr, int l, int r)
{
    int pivot = arr[r];
    int i = l-1;
    for(int j=l; j<r; j++)
    {
        if(arr[j]<pivot)
        {
            i++;
            swap(&arr[i],&arr[j]);
        }
    }
    swap(&arr[i+1],&arr[r]);

    return (i+1);
}

void qsort(int *arr, int l, int r)
{
    if(l<r)
    {
        int p = partition(arr, l, r);
        qsort(arr, l , p-1);
        qsort(arr, p+1, r);
    }
}


void sumClosetToZeroPair(int *arr, int size)
{
```

```
    int curr_sum, closest_sum = INT_MAX, l_index=0, r_index=size-1,
min_l_index=0, min_r_index=size-1;
    while(l_index < r_index)
    {
        curr_sum = arr[l_index] + arr[r_index];
        if(abs(curr_sum)<abs(closest_sum))
        {
            closest_sum = curr_sum;
            min_l_index = l_index;
            min_r_index = r_index;
        }
        if(curr_sum < 0)
          l_index ++;
        else
          r_index --;
    }
    printf("The pair whose sum is close to zero are %d, %d",
arr[min_l_index], arr[min_r_index]);
}



int main()
{
    int arr[] = {0,59,-9,69,-79,84};
    qsort(arr, 0, 5);
    sumClosetToZeroPair(arr,6);
    return 0;
}
```