#include <stdio.h>

void swap(int \**a*, int \**b*)

{

        int temp = \**a*;

        \**a* = \**b*;

        \**b* = temp;

}

void selectionSort(int \**arr*, int *size*)

{

        for(int i = 0; i < *size* - 1; i++)

        {

                int Min = i;

                for(int j = i + 1; j < *size*; j++)

                {

                        if(*arr*[j] < *arr*[Min])

                        {

                                Min = j;

                        }

                }

                swap(&*arr*[i], &*arr*[Min]);

        }

}

void insertionSort(int *arr*[], int *size*)

{

   int i, j, key;

   for (i = 1; i < *size*; i++)

   {

       key = *arr*[i];

       j = i-1;

       while (j >= 0 && *arr*[j] > key)

       {

*arr*[j+1] = *arr*[j];

           j = j-1;

       }

*arr*[j+1] = key;

   }

}

void shakerSort(int \**arr*, int *size*)

{

        int left = 0;

        int right = *size* - 1;

        int k = 0;

        while (left < right)

        {

                for (int i = left; i < right; i++)

                {

                        if (*arr*[i] > *arr*[i + 1])

                        {

                                swap(*arr*[i], *arr*[i + 1]);

                                k = i;

                        }

                }

                right = k;

                for (int i = right; i > left; i--)

                {

                        if (*arr*[i] < *arr*[i - 1])

                        {

                                swap(*arr*[i], *arr*[i - 1]);

                                k = i;

                        }

                }

                left = k;

        }

}

void bubbleSort(int \**arr*, int *size*)

{

        for (int i = 0; i < *size* - 1; i++)

                for (int j = *size* - 1; j > i; j--)

                        if (*arr*[j] < *arr*[j - 1])

                                swap(*arr*[j], *arr*[j - 1]);

}

int partition(int \**arr*, int *low*, int *high*)

{

        int pivot = *arr*[*high*];

        int left = *low*;

        int right = *high* - 1;

        while(1)

        {

                while(left <= right && *arr*[left] < pivot)

                        left++;

                while(right >= left && *arr*[right] > pivot)

                        right--;

                if(left >= right) break;

                swap(*arr*[left], *arr*[right]);

                left++;

                right--;

        }

        swap(*arr*[left], *arr*[*high*]); *//swap pivot and left*

        return left; *//new pivot*

}

void quickSort(int \**arr*, int *low*, int *high*)

{

        if(*low* < *high*)

        {

                int temp = partition(*arr*, *low*, *high*);

                quickSort(*arr*, *low*, temp - 1);

                quickSort(*arr*, temp + 1, *high*);

        }

}

void Output(int \**arr*, int *size*)

{

        for(int i = 0; i < *size*; i++)

                printf("%d ", *arr*[i]);

}

int main()

{

        int Min = 0;

        int arr[5] = {5,4,3,2,1};

        int size = sizeof(arr) / sizeof(int);

*// quickSort(&arr, 0, size - 1);*

*// selectionSort(&arr, size);*

*// insertionSort(&arr, size);*

*// bubbleSort(&arr, size);*

*// shakerSort(&rr, size);*

        Output(arr, size);

}