Bài 4:

Program: int sort

Input: int A[], int n

Output: sorted array

{

    // count number of 0's

    zeros = 0;

    for i = 0 to (n-1) do

    {

        if A[i] = 0 then zeros++;

    }

    // put 0's at the beginning

    k = 0;

    while (zeros--) do A[k++] = 0;

    // fill all remaining elements by 1

    while (k < n) do A[k++] = 1;

}

Bài 10:

void printArray(int arr[], int n)

{

    for i = 0 to (n-1) cout << arr[i] << " ";

}

Program: void merge

Input: int X[], int Y[], int m, int n

Output: Merged sorted array

{

    For i = 0 to (m-1) do

    {

        // compare the current element of `X[]` with the first element of `Y[]`

        if (X[i] > Y[0]) then

        {

            swap(X[i], Y[0]);

            first = Y[0];

            // move `Y[0]` to its correct position to maintain the sorted

            // order of `Y[]`. Note: `Y[1…n-1]` is already sorted

            for k = 1 to (n-1) and Y[k] < first do {Y[k - 1] = Y[k];}

            Y[k - 1] = first;

        }

    }

}