```c
// C program to convert min Heap to max Heap

#include <stdio.h>

void swap(int* a, int* b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
}

// to heapify a subtree with root at given index
void MaxHeapify(int arr[], int i, int N)
{
    int l = 2 * i + 1;
    int r = 2 * i + 2;
    int largest = i;

    if (l < N && arr[l] > arr[i])
        largest = l;
    if (r < N && arr[r] > arr[largest])
        largest = r;
    if (largest != i) {
        swap(&arr[i], &arr[largest]);
        MaxHeapify(arr, largest, N);
    }
}

// This function basically builds max heap
void convertMaxHeap(int arr[], int N)
{
    // Start from bottommost and rightmost
    // internal mode and heapify all internal
    // modes in bottom up way
    for (int i = (N - 2) / 2; i >= 0; --i)
        MaxHeapify(arr, i, N);
}

// A utility function to print a given array
// of given size
void printArray(int* arr, int size)
{
    for (int i = 0; i < size; ++i)
        printf("%d ", arr[i]);
}

// Driver's code
int main()
{
    // array representing Min Heap
    int arr[] = { 3, 5, 9, 6, 8, 20, 10, 12, 18, 9 };
    int N = sizeof(arr) / sizeof(arr[0]);

    printf("Min Heap array : ");
    printArray(arr, N);

    // Function call
    convertMaxHeap(arr, N);

    printf("\nMax Heap array : ");
    printArray(arr, N);

    return 0;
}

```