```c
//50(a) Copying an array to another array using pointer
/**
 * C program to copy an array to another array using pointers
 */

#include <stdio.h>

#define MAX_SIZE 100 // Maximum array size


/* Function declaration to print array */
void printArray(int arr[], int size);


int main()
{
    int source_arr[MAX_SIZE], dest_arr[MAX_SIZE];
    int size, i;

    int *source_ptr = source_arr;   // Pointer to source_arr
    int *dest_ptr   = dest_arr;     // Pointer to dest_arr

    int *end_ptr;


    /*
     * Input size and elements in source array
     */
    printf("Enter size of array: ");
    scanf("%d", &size);
    printf("Enter elements in array: ");
    for (i = 0; i < size; i++)
    {
        scanf("%d", (source_ptr + i));
    }


    // Pointer to last element of source_arr
    end_ptr = &source_arr[size - 1];


    /* Print source and destination array before copying */
    printf("\nSource array before copying: ");
    printArray(source_arr, size);

    printf("\nDestination array before copying: ");
    printArray(dest_arr, size);



    /*
     * Run loop till source_ptr exists in source_arr
     * memory range.
     */
```

```
    while(source_ptr <= end_ptr)
    {
        *dest_ptr = *source_ptr;

        // Increment source_ptr and dest_ptr
        source_ptr++;
        dest_ptr++;
    }


    /* Print source and destination array after copying */
    printf("\n\nSource array after copying: ");
    printArray(source_arr, size);

    printf("\nDestination array after copying: ");
    printArray(dest_arr, size);


    return 0;
}


/**
 * Function to print array elements.
 *
 * @arr     Integer array to print.
 * @size    Size of array.
 */
void printArray(int *arr, int size)
{
    int i;

    for (i = 0; i < size; i++)
    {
        printf("%d, ", *(arr + i));
    }
}


while(source_ptr <= end_ptr)
    *(dest_ptr++) = *(source_ptr++);
Output
Enter size of array: 10
Enter elements in array: 10 -1 100 90 87 0 15 10 20 30

Source array before copying: 10, -1, 100, 90, 87, 0, 15, 10, 20, 30,
Destination array before copying: 0, 0, 127, 127, 0, 1, 0, 16777472, 0,
0,

Source array after copying: 10, -1, 100, 90, 87, 0, 15, 10, 20, 30,
Destination array after copying: 10, -1, 100, 90, 87, 0, 15, 10, 20, 0,
```

```c
//50(b) Reverse an array using pointer
/**
 * C program to reverse an array using pointers
 */

#include <stdio.h>

#define MAX_SIZE 100

/* Function declaration */
void printArr(int *arr, int size);


int main()
{
    int arr[MAX_SIZE];
    int size;
    int *left = arr;  // Pointer to arr[0]
    int *right;


    // Input size of array
    printf("Enter size of array: ");
    scanf("%d", &size);

    right = &arr[size - 1];  // Pointer to arr[size - 1]

    /*
     * Input elements in array
     */
    printf("Enter elements in array: ");
    while(left <= right)
    {
        scanf("%d", left++);
    }


    printf("\nArray before reverse: ");
    printArr(arr, size);


    // Make sure that left points to arr[0]
    left = arr;


    // Loop to reverse array
    while(left < right)
    {
        /*
         * Swap element from left of array to right of array.
         */
        *left    ^= *right;
```

```
        *right   ^= *left;
        *left    ^= *right;

        // Increment left array pointer and decrement right array pointer
        left++;
        right--;
    }


    printf("\nArray after reverse: ");
    printArr(arr, size);


    return 0;
}



/**
 * Function to print array using pointer.
 *
 * @arr     Pointer to array.
 * @size    Size of the array.
 */
void printArr(int * arr, int size)
{
    // Pointer to arr[size - 1]
    int * arrEnd = (arr + size - 1);

    /* Loop till last array element */
    while(arr <= arrEnd)
    {
        printf("%d, ", *arr);

        // Move pointer to next array element.
        arr++;
    }
}
Output
Enter size of array: 10
Enter elements in array: 10 20 30 40 50 60 70 80 90 100

Array before reverse: 10, 20, 30, 40, 50, 60, 70, 80, 90, 100,
Array after reverse: 100, 90, 80, 70, 60, 50, 40, 30, 20, 10
```