Arrays (6-8-2025)

1. Write a program to read and print elements of an array.

Input: take n as the input n is used to print n element

Process : create a for loop to scan the array and print the array

Output: print the array

Program:

#include <stdio.h>

void main()

{

int i,a[5];

for(i=0;i<5;i++)

{

scanf("%d",&a[i]);

printf("%d\n",a[i]);

}

}



1. Write a program to find the sum of elements of an array.

Input: take n as the input n is used to print n element

Process : create a for loop to scan the array do sum of the array

Output: print sum the array

Program:

#include <stdio.h>

void main()

{

int i,a[5],sum=0;

for(i=0;i<5;i++)

{

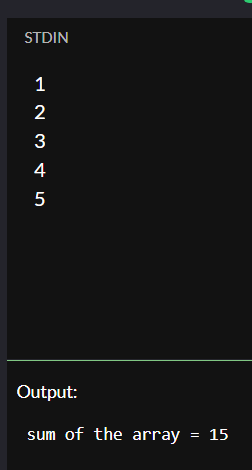
scanf("%d",&a[i]);

sum=sum+a[i];

}

printf(" sum of the array = %d",sum);

}



1. Write a program to find the maximum and minimum element in an array

Input: take input max=0,I;

Process: check the maximum int the array

Output : print the maximum no in the array

Program:.

#include <stdio.h>

void main()

{

int i,max=0,a[5];

for(i=0;i<5;i++)

{

scanf("%d",&a[i]);

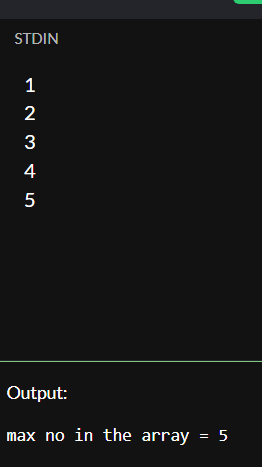
if(max<a[i])

max=a[i];

}

printf("max no in the array = %d",max);

}



1. Write a program to reverse an array.

**Input**: Size of the array n Array elements

**Process**: Swap the first element with the last, second with second-last, etc.

**Output** :The array printed in reverse order

Program:

#include <stdio.h>

int main() {

int n, i, a[100];

printf("Enter the number of elements: ");

scanf("%d", &n);

printf("Enter %d elements:\n", n);

for(i = 0; i < n; i++) {

scanf("%d", &a[i]);

}

printf("Array in reverse order:\n");

for(i = n - 1; i >= 0; i--) {

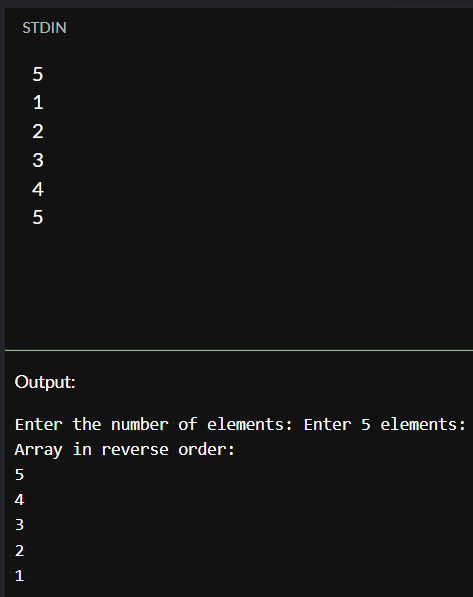
printf("%d \n", a[i]);

}

printf("\n");

return 0;

}



1. Write a program to search for an element in an array (linear search).

**Input:**

* Number of elements n
* Array of n integers
* Key (element to search)

**Process:**

* Loop through the array from index 0 to n-1
* Compare each element with the key
* If found, print the index (or position)
* If not found after loop, print "Not found"

**Output:**

* Message indicating whether the element was found, and its position (if found)

Program:

#include <stdio.h>

int main() {

int a[100], n, i, key, found = 0;

printf("Enter the number of elements: ");

scanf("%d", &n);

printf("Enter %d elements:\n", n);

for(i = 0; i < n; i++) {

scanf("%d", &a[i]);

}

printf("Enter the element to search: ");

scanf("%d", &key);

for(i = 0; i < n; i++) {

if(a[i] == key) {

printf("Element %d found at position %d\n", key, i + 1);

found = 1;

break;

}

}

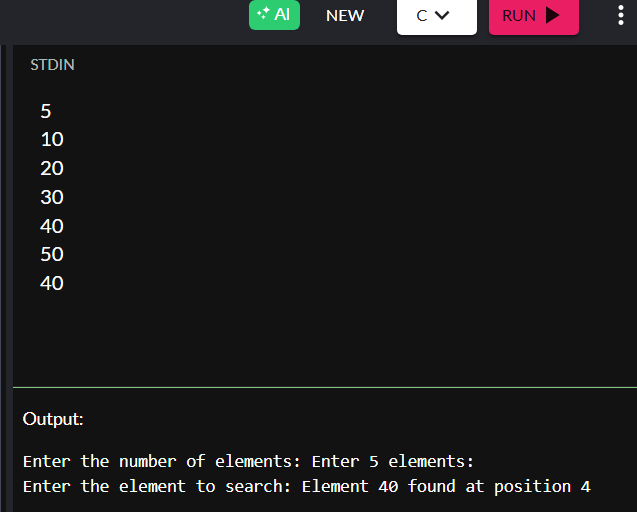
if(!found) {

printf("Element %d not found in the array\n", key);

}

return 0;

}



1. Write a program to sort an array in ascending order.

**Input:**

* Number of elements n
* Array of n integers

**Process:**

* Use **nested loops** to compare and sort elements (using **Bubble Sort** or **Selection Sort**)
* Swap elements to arrange them in ascending order

**Output:**

* The sorted array in ascending order

Program:

#include <stdio.h>

int main() {

int a[100], n, i, j, temp;

printf("Enter the number of elements: ");

scanf("%d", &n);

printf("Enter %d elements:\n", n);

for(i = 0; i < n; i++) {

scanf("%d", &a[i]);

}

// Bubble Sort to sort array in ascending order

for(i = 0; i < n - 1; i++) {

for(j = 0; j < n - i - 1; j++) {

if(a[j] > a[j + 1]) {

// Swap

temp = a[j];

a[j] = a[j + 1];

a[j + 1] = temp;

}

}

}

printf("Array in ascending order:\n");

for(i = 0; i < n; i++) {

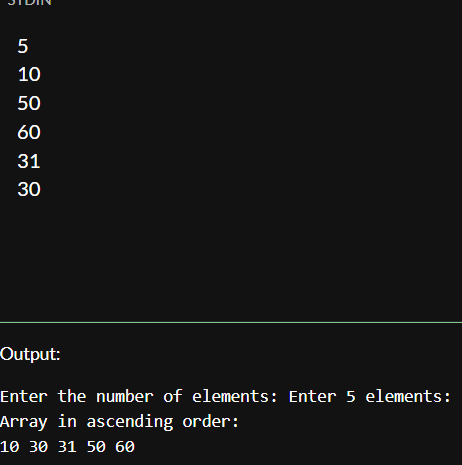
printf("%d ", a[i]);

}

printf("\n");

return 0;

}



1. Write a program to insert an element in an array.

Input:

* Number of elements n
* Array elements
* Position pos to insert
* Element val to insert

**Process:**

* Shift all elements from position pos to the right
* Insert the new element at position pos

**Output:**

* The array after insertion, with the new element added

Program:

#include <stdio.h>

int main() {

int a[100], n, i, pos, val;

printf("Enter the number of elements: ");

scanf("%d", &n);

printf("Enter %d elements:\n", n);

for(i = 0; i < n; i++) {

scanf("%d", &a[i]);

}

printf("Enter the position to insert (1 to %d): ", n + 1);

scanf("%d", &pos);

if(pos < 1 || pos > n + 1) {

printf("Invalid position!\n");

return 1;

}

printf("Enter the value to insert: ");

scanf("%d", &val);

// Shift elements to the right

for(i = n; i >= pos; i--) {

a[i] = a[i - 1];

}

a[pos - 1] = val; // Insert element

n++; // Increase size

printf("Array after insertion:\n");

for(i = 0; i < n; i++) {

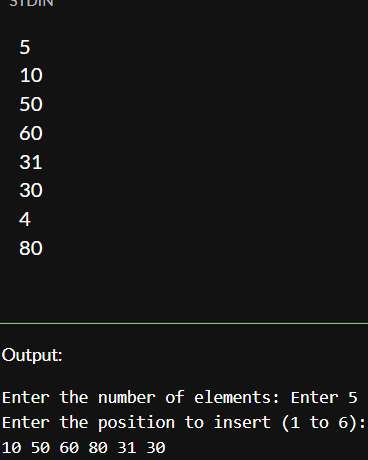
printf("%d ", a[i]);

}

printf("\n");

return 0;

}



1. Write a program to delete an element from an array.

**Input:**

* Number of elements n
* Array elements
* Position pos of the element to delete

**Process:**

* Shift all elements after pos one step to the left to overwrite the deleted element

**Output:**

* The array after deletion

Program:

#include <stdio.h>

int main() {

int a[100], n, i, pos;

printf("Enter the number of elements: ");

scanf("%d", &n);

printf("Enter %d elements:\n", n);

for(i = 0; i < n; i++) {

scanf("%d", &a[i]);

}

printf("Enter the position to delete (1 to %d): ", n);

scanf("%d", &pos);

if(pos < 1 || pos > n) {

printf("Invalid position!\n");

return 1;

}

// Shift elements to the left from the position

for(i = pos - 1; i < n - 1; i++) {

a[i] = a[i + 1];

}

n--; // Reduce array size

printf("Array after deletion:\n");

for(i = 0; i < n; i++) {

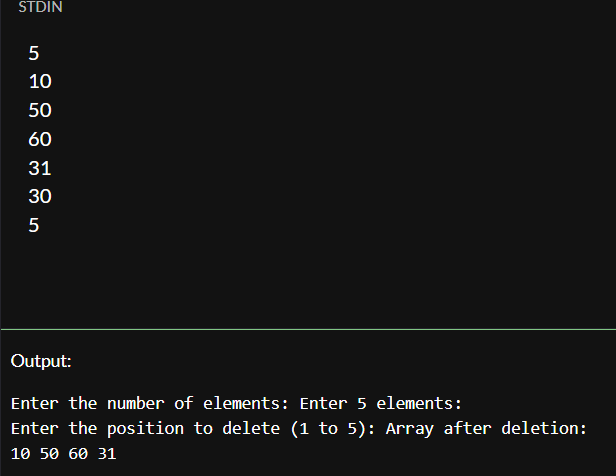
printf("%d ", a[i]);

}

printf("\n");

return 0;

}



1. Write a program to find the frequency of elements in an array.

**Input:**

* Number of elements n
* Array a[n] of integers

**Process:**

* Loop through the array
* For each element, count how many times it appears in the array
* Ensure duplicates are not counted multiple times (track visited elements)

**Output:**

* Each unique element along with its frequency (how many times it appears)

Program:

#include <stdio.h>

int main() {

int a[5], b[5], i, j, k = 0, flag;

// Input array

printf("Enter 5 elements:\n");

for(i = 0; i < 5; i++) {

scanf("%d", &a[i]);

}

// Remove duplicates

for(i = 0; i < 5; i++) {

flag = 0;

for(j = 0; j < k; j++) {

if(a[i] == b[j]) {

flag = 1;

break;

}

}

if(flag == 0) {

b[k] = a[i];

k++;

}

}

// Print array after removing duplicates

printf("Array after removing duplicates:\n");

int count=0;

for(i = 0; i < k; i++)

{

count=0;

for(j=0;j<5;j++)

{

if(b[i]==a[j])

count++;

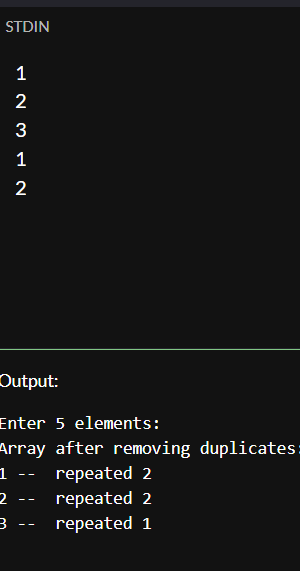
}

printf("%d -- repeated %d\n", b[i],count);

}

printf("\n");

}



1. Write a program to merge two arrays.

**Input:**

* Two arrays a[] and b[]
* Sizes n1 and n2

**Process:**

* Copy all elements from a[] and b[] into a third array c[]

**Output:**

* The merged array c[] containing all elements of a[] followed by b[]

Program:

#include <stdio.h>

int main() {

int a[3] = {1, 2, 3};

int b[2] = {4, 5};

int c[5];

int i, k = 0;

// Merging first array

for(i = 0; i < 3; i++) {

c[k++] = a[i];

}

// Merging second array

for(i = 0; i < 2; i++) {

c[k++] = b[i];

}

// Output

printf("Merged array:\n");

for(i = 0; i < k; i++) {

printf("%d ", c[i]);

}

return 0;

}

