NAME = UPASNA

ROLL NO = 24csu225

ASSIGNMENT:-2

Ques1:- WAP to increace every student marks by 5$ then print updated array?

Sol:- #include <stdio.h>

int main() {

int n;

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

scanf("%d", &n);

int marks[n];

printf("Enter the marks of each student:\n");

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

printf("Student %d: ", i + 1);

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

}

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

marks[i] += 5;

}

printf("\nUpdated marks after adding 5:\n");

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

printf("Student %d: %d\n", i + 1, marks[i]);

}

    return 0;

}

Ques2:- WAP to print grade of students as per their marks given in any array.(>=75-A grade,74 to 60-B grade, 59 to 40 C grade below 40-D grade)?

Sol:- #include <stdio.h>

int main() {

int n;

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

scanf("%d", &n);

int marks[n];

printf("Enter the marks of each student:\n");

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

printf("Student %d: ", i + 1);

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

}

printf("\nGrades of students:\n");

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

printf("Student %d: ", i + 1);

if (marks[i] >= 75) {

printf("A grade\n");

} else if (marks[i] >= 60) {

printf("B grade\n");

} else if (marks[i] >= 40) {

printf("C grade\n");

} else {

printf("D grade\n");

}

}

    return 0;

}

Ques3:- WAP to find who scored first "99" in an array marks?

Sol:- #include <stdio.h>

int main() {

int n, found = 0;

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

scanf("%d", &n);

int marks[n];

printf("Enter the marks of each student:\n");

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

printf("Student %d: ", i + 1);

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

}

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

if (marks[i] == 99) {

printf("\nThe first student who scored 99 is Student %d\n", i + 1);

found = 1;

break;

}

}

if (!found) {

printf("\nNo student scored 99.\n");

}

    return 0;

}

Ques4:- WAP to find who $ how many students have scored 99 in an array marks?

Sol:- #include <stdio.h>

int main() {

int n, count = 0;

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

scanf("%d", &n);

int marks[n];

printf("Enter the marks of each student:\n");

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

printf("Student %d: ", i + 1);

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

}

printf("\nStudents who scored 99:\n");

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

if (marks[i] == 99) {

printf("Student %d\n", i + 1);

count++;

}

}

if (count == 0) {

printf("No student scored 99.\n");

} else {

printf("\nTotal number of students who scored 99: %d\n", count);

}

    return 0;

}

Ques5:- WAP to find sum of all scores in marks array?

Sol:- #include <stdio.h>

int main() {

int n, sum = 0;

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

scanf("%d", &n);

int marks[n];

printf("Enter the marks of each student:\n");

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

printf("Student %d: ", i + 1);

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

sum += marks[i];

}

printf("\nThe sum of all scores is: %d\n", sum) ;

return 0;

}

Ques6:- WAP to find average score of the marks array?

Sol:- include <stdio.h>

int main() {

int n;

float sum = 0.0, average;

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

scanf("%d", &n);

int marks[n];

printf("Enter the marks of each student:\n");

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

printf("Student %d: ", i + 1);

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

sum += marks[i];

}

average = sum / n;

printf("\nThe average score is: %.2f\n", average);

    return 0;

}

Ques7:-Wap to check whether score is even or odd in an array?

Sol:-# include <stdio.h>

int main() {

int n;

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

scanf("%d", &n);

int marks[n];

printf("Enter the marks of each student:\n");

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

printf("Student %d: ", i + 1);

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

}

printf("\nScores and their even/odd status:\n");

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

if (marks[i] % 2 == 0) {

printf("Student %d: %d (Even)\n", i + 1, marks[i]);

} else {

printf("Student %d: %d (Odd)\n", i + 1, marks[i]);

}

   }

return 0;

}

Ques8 WAP to find minimum&maximum score in the marks array?

Sol:- #include <stdio.h>

int main() {

int n;

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

scanf("%d", &n);

int marks[n];

printf("Enter the marks of each student:\n");

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

printf("Student %d: ", i + 1);

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

}

int min = marks[0];

int max = marks[0];

for (int i = 1; i < n; i++) {

if (marks[i] < min) {

min = marks[i];

}

if (marks[i] > max) {

max = marks[i];

}

}

printf("\nMinimum score: %d\n", min);

printf("Maximum score: %d\n", max);

    return 0;

}

Ques9 WAP to find a peak element which is not smaller then its neighbour?

Sol:-# include <stdio.h>

int main() {

int n;

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

scanf("%d", &n);

if (n <= 0) {

printf("Array size must be positive.\n");

return 1;

}

int arr[n];

printf("Enter the elements of the array:\n");

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

printf("Element %d: ", i + 1);

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

}

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

if ((i == 0 || arr[i] >= arr[i - 1]) && (i == n - 1 || arr[i] >= arr[i + 1])) {

printf("Peak element found: %d at index %d\n", arr[i], i);

return 0;

}

}

printf("No peak element found.\n");

    return 0;

}

Ques10 WAP to count prime numbers in an array?

Sol:- #include <stdio.h>

#include <stdbool.h>

bool isPrime(int num) {

if (num <= 1) return false;

for (int i = 2; i \* i <= num; i++) {

if (num % i == 0) {

return false;

}

}

return true;

}

int countPrimes(int arr[], int n) {

int count = 0;

printf("Prime numbers found in the array:\n");

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

if (isPrime(arr[i])) {

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

count++;

}

}

printf("\n");

return count;

}

int main() {

int n;

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

scanf("%d", &n);

if (n <= 0) {

printf("Array size must be positive.\n");

return 1; //

}

int arr[n];

printf("Enter the elements of the array:\n");

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

printf("Element %d: ", i + 1);

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

}

int primeCount = countPrimes(arr, n);

printf("\nTotal number of prime numbers in the array: %d\n", primeCount);

    return 0;

}

Ques11 WAP to implement insert – front,any position in between & end in an array.print the array before insert & after insert?

Sol:- #include <stdio.h>

#define MAX\_SIZE 100

void printArray(int arr[], int size) {

printf("Array: ");

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

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

}

printf("\n");

}

void insertFront(int arr[], int \*size, int value) {

if (\*size >= MAX\_SIZE) return;

printArray(arr, \*size);

for (int i = \*size; i > 0; i--) arr[i] = arr[i - 1];

arr[0] = value;

(\*size)++;

printArray(arr, \*size);

}

void insertAtPosition(int arr[], int \*size, int value, int position) {

if (position < 0 || position > \*size || \*size >= MAX\_SIZE) return;

printArray(arr, \*size);

for (int i = \*size; i > position; i--) arr[i] = arr[i - 1];

arr[position] = value;

(\*size)++;

printArray(arr, \*size);

}

void insertEnd(int arr[], int \*size, int value) {

if (\*size >= MAX\_SIZE) return;

printArray(arr, \*size);

arr[(\*size)++] = value;

printArray(arr, \*size);

}

int main() {

int arr[MAX\_SIZE];

int size = 0;

insertFront(arr, &size, 10);

insertEnd(arr, &size, 20);

insertAtPosition(arr, &size, 15, 1);

insertAtPosition(arr, &size, 25, 5); // Invalid position example

    return 0;

}

Ques 12:-WAP to implement delete-front,any position in between & end in an array.print the array before delete &after delete?

Sol:- #include <stdio.h>

#define MAX\_SIZE 100

void printArray(int arr[], int size) {

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

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

}

printf("\n");

}

void rotateClockwise(int arr[], int size) {

if (size == 0) return;

int last = arr[size - 1];

for (int i = size - 1; i > 0; i--) {

arr[i] = arr[i - 1];

}

arr[0] = last;

}

int main() {

int arr[] = {1, 2, 3, 4, 5};

int size = sizeof(arr) / sizeof(arr[0]);

printf("Original array: ");

printArray(arr, size);

rotateClockwise(arr, size);

printf("Rotated array: ");

printArray(arr, size);

    return 0;

}

Ques13 Given an array ,the task is to cyclically rotate the array the array clockwise by one time.

Examples:

Input : arr[] = (1,2,3,4,5)

Output : arr[] = (5,1,2,3,4)

Input: arr = (2,3,5,5,1)

Output : arr = (1,2,3,4,5)

Sol:-

#include <stdio.h>

#define MAX\_SIZE 100

void printArray(int arr[], int size) {

printf("Array: ");

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

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

}

printf("\n");

}

void insertFront(int arr[], int \*size, int value) {

if (\*size >= MAX\_SIZE) return;

printArray(arr, \*size);

for (int i = \*size; i > 0; i--) arr[i] = arr[i - 1];

arr[0] = value;

(\*size)++;

printArray(arr, \*size);

}

void insertAtPosition(int arr[], int \*size, int value, int position) {

if (position < 0 || position > \*size || \*size >= MAX\_SIZE) return;

printArray(arr, \*size);

for (int i = \*size; i > position; i--) arr[i] = arr[i - 1];

arr[position] = value;

(\*size)++;

printArray(arr, \*size);

}

void insertEnd(int arr[], int \*size, int value) {

if (\*size >= MAX\_SIZE) return;

printArray(arr, \*size);

arr[(\*size)++] = value;

printArray(arr, \*size);

}

int main() {

int arr[MAX\_SIZE];

int size = 0;

insertFront(arr, &size, 10);

insertEnd(arr, &size, 20);

insertAtPosition(arr, &size, 15, 1);

insertAtPosition(arr, &size, 25, 5); // Invalid position example

    return 0;

}

Ques:-14Given an array of n integers. The task is to print the duplicates in the given array. If there are no. duplicates then print-1.

Examples:

Input(2,10,10,100,2,10,11,2,11,2)

Output:2 to 11

Inpu t :(5,40,1,40,100000,1,5,1)

Output:5 40 1?

Sol:- #include <stdio.h>

#define MAX\_SIZE 100

void printDuplicates(int arr[], int size) {

int count[MAX\_SIZE] = {0};

int foundDuplicate = 0;

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

count[arr[i]]++;

}

for (int i = 0; i < MAX\_SIZE; i++) {

if (count[i] > 1) {

printf("%d ", i);

foundDuplicate = 1;

}

}

if (!foundDuplicate) {

printf("-1");

}

printf("\n");

}

int main() {

int arr1[] = {2, 10, 10, 100, 2, 10, 11, 2, 11, 2};

int size1 = sizeof(arr1) / sizeof(arr1[0]);

int arr2[] = {5, 40, 1, 40, 100000, 1, 5, 1};

int size2 = sizeof(arr2) / sizeof(arr2[0]);

printf("Input: ");

printDuplicates(arr1, size1);

printf("Input: ");

printDuplicates(arr2, size2);

    return 0;

}