

Name: Muhammad Mubbashir Khan

Cms:465077

Sec: b

Task:1

#include <iostream>

using namespace std;

int main()

{

// Declare an array of size 10

int arr[10];

// Loop through the array and take input from user

for (int i = 0; i < 10; i++)

{

cout << "Enter element " << i + 1 << ": ";

cin >> arr[i];

}

cout << "The elements of the array are: ";

for (int i = 0; i < 10; i++)

{

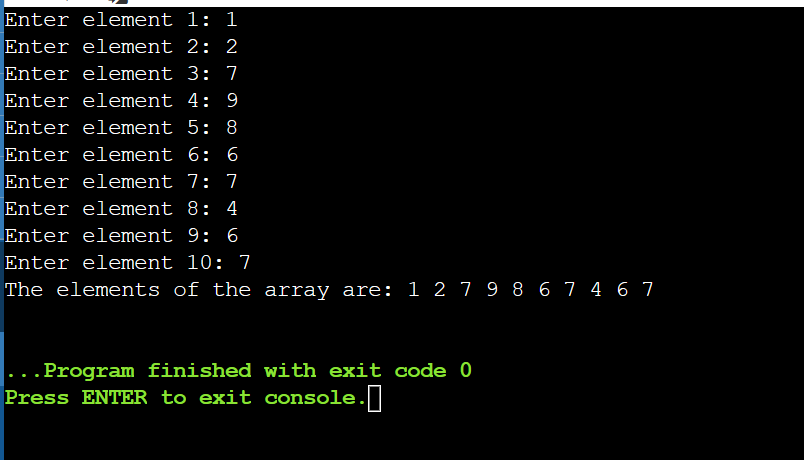
cout << arr[i] << " ";

}

cout << endl;

return 0;

}



Task no:2

#include <iostream>

using namespace std;

int main()

{

// Declare an array of 5 integers

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

// Initialize sum and product variables to 0 and 1 respectively

int sum = 0;

int product = 1;

// Loop through the array elements

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

{

// Add the current element to the sum

sum += arr[i];

// Multiply the current element to the product

product \*= arr[i];

}

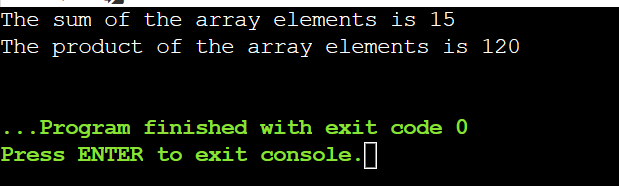
// Print the sum and product

cout << "The sum of the array elements is " << sum << endl;

cout << "The product of the array elements is " << product << endl;

return 0;

}



Task no:3#include <iostream>

using namespace std;

// A function to print a diamond pattern using a single array

void printDiamond(int n) {

// n must be an odd number

if (n % 2 == 0) {

cout << "Invalid input. n must be an odd number." << endl;

return;

}

// Create an array of size n

char arr[n];

// Initialize the array with spaces

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

arr[i] = ' ';

}

// The middle index of the array

int mid = n / 2;

// The number of stars to print in each row

int stars = 1;

// Loop through the rows

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

// The starting index of the stars in the array

int start = mid - (stars - 1) / 2;

// The ending index of the stars in the array

int end = mid + (stars - 1) / 2;

// Fill the array with stars from start to end

for (int j = start; j <= end; j++) {

arr[j] = '\*';

}

// Print the array

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

cout << arr[j];

}

cout << endl;

// Reset the array with spaces

for (int j = start; j <= end; j++) {

arr[j] = ' ';

}

// Update the number of stars for the next row

// If it is the middle row or above, increase by 2

// If it is below the middle row, decrease by 2

if (i < mid) {

stars += 2;

} else {

stars -= 2;

}

}

}

// A main function to test the code

int main() {

// A sample input

int n = 7;

// Call the function

printDiamond(n);

// The output should be:

// \*

// \*\*\*

// \*\*\*\*\*

// \*\*\*\*\*\*\*

// \*\*\*\*\*

// \*\*\*

// \*

return 0;

}

