Traverse an array

locked

Problem | Submissions | Leaderboard | Discussions

Given code snippet (in code stub below) gives you idea regarding three different ways to traverse an array.

Please read comments and output statements carefully!

Try to change different parts of code then rerun your code and see how it affects the output.

Input Format

Run this code with custom input and give following input

5 1 2 3 4 5

Constraints

No constraints

Output Format

Observe the output

Submissions: 54
Max Score: 10
Difficulty: Easy

Rate This Challenge:

Download problem statement

Download all test cases

Suggest Edits

Collapse

Admin Options

Edit Challenge

View Submissions

C

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
```

```c

int main()
{
    int arr[100], number_of_elements, *pntr, i;

    printf("Enter number of elements to be entered (max 100): \n");
    scanf("%d", &number_of_elements);

    if(number_of_elements > 100)
    {
        printf("Elements are more than 100.\n");
        return(-1);
    }
    else
    {
        pntr = arr;
        for(i = 0; i < number_of_elements; i++)
        {
            // please note that & is not needed here before pntr
            // Do you know why? If not, ask your friend or professor
            scanf("%d", pntr);
            pntr++;
            printf("current value of pntr is %p\n", (void *)pntr);
        }

        // reseting pntr to point to first element
        pntr = arr;

        printf("\nPrinting array elements with array index notation using arr\n");
        printf("arr in the beggining = %p\n", (void *)arr);
        for(i = 0; i < number_of_elements; i++)
        {
            printf("%d ", arr[i]);
        }
        printf("\narr at end = %p", (void *)arr);

        printf("\n\nPrinting array elements with array index notation using pntr\n");
        printf("pntr in the beggining = %p\n", (void *)pntr);
        for(i = 0; i < number_of_elements; i++)
        {
            printf("%d ", pntr[i]);
        }
        printf("\npntr at end = %p", (void *)pntr);

        printf("\n\nPrinting array elements with pointer and offset using arr\n");
        printf("arr in the beggining = %p\n", (void *)arr);
        for(i = 0; i < number_of_elements; i++)
        {
            printf("%d ", *(arr + i));
        }
        printf("\narr at end = %p", (void *)arr);

        printf("\n\nPrinting array elements with pointer and offset using pntr\n");
        printf("pntr in the beggining = %p\n", (void *)pntr);
        for(i = 0; i < number_of_elements; i++)
        {
            printf("%d ", *(pntr + i));
        }
        printf("\npntr at end = %p", (void *)pntr);

        // try uncommenting following six lines and see what is the output
        //printf("\nPrinting array elements by traversing an array with arr\n");
        //for(i = 0; i < number_of_elements; i++)
        //{
        //  printf("%d ", *arr);
        //  arr++;
        //}
```

```
72
73          printf("\n\nPrinting array elements by traversing an array with pntr\n");
74          printf("pntr in the beggining = %p\n", (void *)pntr);
75          for(i = 0; i < number_of_elements; i++)
76          {
77              printf("%d ", *pntr);
78              pntr++;
79          }
80          printf("\npntr at end of traversing = %p\n", (void *)pntr);
81          printf("Please note that pntr has changed in this last technique\n");
82      }
83  }
```

Line: 83 Col: 2

Upload Code as File Test against custom input Run Code Submit Code