

Pointers -2

Assignment Solutions



Q1 - Write a program to print the kth element of an array using pointers.

The first line of the input contains the size of the array and the value of k.

The second line of input contains the elements of the array. You can assume that $0 \leq k < \text{size of the array}$.

To access the k-th element of the array using a pointer, we can make the pointer point to the $(a + k - 1)$ i.e. the address of the k-th element. Here 'a' is the name of the array.

Code link - <https://pastebin.com/XrzM7naT>

```
a.cpp > ...
1  #include<bits/stdc++.h>
2  using namespace std;
3
4  int main()
5  {
6      int n, k;
7      cin >> n >> k;
8      int a[n];
9      for(int i = 0; i < n; ++i) {
10         cin >> a[i];
11     }
12     int *ptr = (a + k - 1);
13     cout << *ptr;
14
15     return 0;
16 }
17
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** JUPYTER

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

```
PS C:\Users\akshi\OneDrive\Documents\coding> cd "c:\Users\akshi\OneDrive\Documents\coding\" ; if ($?) { g++ a.cpp -o a } ; if ($?) { .\a }
4 3
10 3 20 52
20
PS C:\Users\akshi\OneDrive\Documents\coding>
```

Q2 - Write a program to find the sum of all the elements of an array. Use pointers to traverse the array.

The first line of the input contains the size of the array.

The second line of input contains the elements of the array.

The process for finding the sum of the array remains the same. The only difference is the way we access the elements. To access the elements of the array we start by making our pointer point to the first element of the array. We can use the dereference operator to get the value of the current element, and increment the pointer to move on to the next element.

Code link - <https://pastebin.com/bsbvTjvG>

```

a.cpp > main()
1  #include<bits/stdc++.h>
2  using namespace std;
3
4  int main()
5  {
6      int n;
7      cin >> n;
8      int a[n];
9      int *ptr = a;
10     for(int i = 0; i < n; ++i) {
11         cin >> *ptr;
12         ptr++;
13     }
14     ptr = a;
15     int sum = 0;
16     for(int i = 0; i < n; i++) {
17         sum += *ptr;
18         ptr++;
19     }
20     cout << sum;
21
22     return 0;
23 }
24

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

PS C:\Users\akshi\OneDrive\Documents\coding> cd "c:\Users\akshi\OneDrive\Documents\coding\" ; if (\$?) { g++ a.cpp -o a } ; if (\$?) { .\a }
4 1 4 3 2
10
PS C:\Users\akshi\OneDrive\Documents\coding>

Q3 - Write a program to traverse the array in the reverse order using pointers.

The first line of the input contains the size of the array.

The second line of input contains the elements of the array.

This question is exactly the same as the previous one. The difference here is that we will initialise the pointer by the address of the last element of the array and to move in the reverse direction, we'll decrement the pointer.

Code link - <https://pastebin.com/ia6Xr7Xg>

```

main.cpp > main()
1  #include <bits/stdc++.h>
2  using namespace std;
3
4  int main()
5  {
6      int n;
7      cin >> n;
8      int a[n];
9      for (int i = 0; i < n; i++)
10     {
11         cin >> a[i];
12     }
13     int *ptr = &a[n - 1];
14     while (1)
15     {
16         cout << *ptr << ' ';
17         if (ptr == a)
18             break;
19         ptr--;
20     }
21     return 0;
22 }
    
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\Coding> cd "d:\Coding\" ; if ($?) { g++ main.cpp -o main } ; if ($?) { .\main }
5 1 2 3 4 5
5 4 3 2 1
PS D:\Coding>
    
```

Q4 - Using pointers, find the sum of elements present on the primary diagonal of the given n x n 2-D matrix.

The first line of input contains the n, the size of the matrix.

The next n lines contain the number of elements present in each row of the matrix.

We already know how to access the diagonal elements of a square matrix, $a[i][i]$, where 'a' is the name of the matrix.

In this question, we need to access the same element but with the help of pointers. To do that we can simply write $*(a + i + i)$.

Code link - <https://pastebin.com/EyCQXWnY>

```
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 int main()
5 {
6     int n;
7     cin >> n;
8     int a[n][n];
9     for(int i = 0; i < n; ++i)
10         for(int j = 0; j < n; j++)
11             cin >> a[i][j];
12     int sum = 0;
13     for(int i = 0; i < n; i++)
14         sum += (*(a + i) + i);
15     cout << sum;
16
17     return 0;
18 }
19
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

```
PS C:\Users\akshi\OneDrive\Documents\coding> cd "c:\Users\akshi\OneDrive\Documents\coding\" ; if ($?) { g++ a.cpp -o a } ; if ($?) { .\a }
```

```
3
```

```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

```
15
```

```
PS C:\Users\akshi\OneDrive\Documents\coding> 
```