

<b>Đã bắt đầu vào lúc</b>	Thứ ba, 5 Tháng mười hai 2023, 7:24 PM
<b>Tình trạng</b>	Đã hoàn thành
<b>Hoàn thành vào lúc</b>	Thứ ba, 5 Tháng mười hai 2023, 9:22 PM
<b>Thời gian thực hiện</b>	1 giờ 57 phút
<b>Điểm</b>	2,00/2,00
<b>Điểm</b>	<b>10,00</b> của 10,00 ( <b>100%</b> )

## Câu hỏi 1

Chính xác

Điểm 1,00 của  
1,00

Hiện thực hàm sau:

**int calcSum(int \*ptr, int n);**

Tính và trả về tổng của các phần tử trong mảng 1 chiều được cho bởi con trỏ.

Trong đó:

**ptr** là con trỏ tới phần tử đầu tiên trong mảng.

**n** là kích thước của mảng.

**Lưu ý:** Bạn cần phải dùng dereference operator (\*) để lấy giá trị của các phần tử trong mảng. Không được dùng subscript operator ([]).

Implement the following function:

**int calcSum(int \*ptr, int n);**

Calculate and return the sum of elements of a 1-dimension array given by a pointer.

Where:

**ptr** is a pointer to the first element in the array.

**n** is the size of the array.

**Note:** You need to use the dereference operator (\*) to get the values of the elements in the array. The subscript operator ([]) cannot be used.

**For example:**

Test	Result
<pre>int arr[] = {1, 2, 3, 4, 5}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	15
<pre>int arr[] = {0, -1, 5, 6, -5, 1, -9, -10, -6, 3}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	-16
<pre>int arr[] = {8, 10, -3, -6, -10, -7, 4, 3, 5, 7}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	11
<pre>int arr[] = {-2, -8, 2, -12, 11, 14, -5, 20, 10, 9, -11, -9, 15, 17, 8}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	59
<pre>int arr[] = {18, 3, -13, -5, -19, 6, 1, 16, -10, -12, 15, 10, 13, -2, -4}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	17

Test	Result
<pre>int arr[] = {-5, 7, 18, -15, 9, -3, -9, 8, -12, 0, 12, -14, -2, 4, -11}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	-13
<pre>int arr[] = {2, 14, 10, 5, 8, 12, 7, 18, 16, 19, 9, 3, 0, 4, 15, 13, 6, 11, 20, 17}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	209
<pre>int arr[] = {16, 18, 0, 13, 7, 15, 1, 9, 3, 20, 19, 4, 11, 6, 2, 8, 5, 17, 10, 14}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	198
<pre>int arr[] = {12, 17, 20, 19, 18, 13, 10, 15, 11, 5, 6, 16, 14, 2, 1, 7, 9, 4, 8, 0}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	207
<pre>int arr[] = {20, 19, 3, 4, 2, 12, 7, 16, 18, 1, 5, 15, 0, 6, 8, 14, 10, 13, 11, 17}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	201

**Answer:** (penalty regime: 0 %)

Reset answer

```

1 | int calcSum(int *ptr, int n)
2 | {
3 |     int sum = 0;
4 |     for (int i = 0; i < n; i++) {
5 |         sum += *ptr;
6 |         ptr++;
7 |     }
8 |     return sum;
9 | }
```

	Test	Expected	Got	
✓	<pre>int arr[] = {1, 2, 3, 4, 5}; cout &lt;&lt; calcSum(arr, sizeof(arr) / sizeof(arr[0]));</pre>	15	15	✓

Passed all tests! ✓

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

## Câu hỏi 2

Chính xác

Điểm 1,00 của  
1,00

Hiện thực hàm sau:

**void add(int \*ptr, int n, int k);**

Thực hiện thêm phần tử vào cuối của mảng 1 chiều được cho bởi con trỏ.

Trong đó:

**ptr** là con trỏ tới phần tử đầu tiên trong mảng.

**n, k** lần lượt là kích thước của mảng và phần tử cần được thêm vào.

Implement the following function:

**void add(int \*ptr, int n, int k);**

Insert element to the end of the 1-dimension array given by a pointer.

Where:

**ptr** is a pointer to the first element in the array.

**n, k** respectively is the size of the array and the element that need to be added.

**For example:**

Test	Result
<pre>int arr[100] = {1, 2, 3, 4, 5}; int n = 5; int k = 7; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	7
<pre>int arr[100] = {3, 9, 20, 6, 18, 0, 16, 8, 15, 14}; int n = 10; int k = 0; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	0

Test	Result
<pre>int arr[100] = {3, 9, 20, 6, 18, 0, 16, 8, 15, 14}; int n = 10; int k = 2; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	2
<pre>int arr[100] = {3, 9, 20, 6, 18, 0, 16, 8, 15, 14, 12, -10}; int n = 11; int k = -8; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	-8
<pre>int arr[100] = {3, 9, 20, 6, 18, 0, 16, 8, 15, 14, 12, -10}; int n = 8; int k = -10; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	-10
<pre>int arr[100] = {15, 8, -14, 13, -17, -12, 10, -15, -9, 4, 1, 0, 16, -11, -5, 19, 17, -13, -18, -4, 0}; int n = 22; int k = -100; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	-100
<pre>int arr[100] = {15, 8, -14, 13, -17, -12, 10, -15, -9, 4, 1, 0, 16, -11, -5, 19, 17, -13, -18, -4, 0}; int n = 21; int k = -110; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	-110
<pre>int arr[100] = {1}; int n = 1; int k = -1; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	-1

Test	Result
<pre>int arr[100] = {}; int n = 0; int k = -1; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	-1
<pre>int arr[100] = {5}; int n = 0; int k = -1; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	-1

**Answer:** (penalty regime: 0 %)

Reset answer

```
1 void add(int *ptr, int n, int k)  
2 {  
3     ptr += n;  
4     *ptr = k;  
5 }
```

	Test	Expected	Got	
✓	<pre>int arr[100] = {1, 2, 3, 4, 5}; int n = 5; int k = 7; add(arr, n, k); cout &lt;&lt; arr[n];</pre>	7	7	✓

Passed all tests! ✓

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

## BÁCH KHOA E-LEARNING



## WEBSITE

HCMUT

MyBK

BKSI

## LIÊN HỆ

📍 268 Lý Thường Kiệt, P.14, Q.10, TP.HCM

☎ (028) 38 651 670 - (028) 38 647 256 (Ext: 5258, 5234)

✉ elearning@hcmut.edu.vn

Copyright 2007-2022 BKEL - Phát triển dựa trên Moodle