

## Contents

Bài thực hành số 1 – Tuần 7.....	2
<b>Bài 1.1.</b> Viết một chương trình C nhập vào 3 số nguyên. Thiết lập một con trỏ để lần lượt trỏ tới từng số nguyên và hiển thị kết quả giá trị tham chiếu ngược của con trỏ. ....	2
<b>Bài 1.2.</b> Viết chương trình in ra địa chỉ của 5 phần tử đầu tiên trong mảng được định nghĩa sau đây: <code>int a[7]= {13, -355, 235, 47, 67, 943, 1222};</code> .....	3
<b>Bài 1.3.</b> Viết chương trình yêu cầu nhập giá trị cho 3 biến số nguyên x, y, z kiểu int. Sau đó sử dụng duy nhất một con trỏ để cộng giá trị của mỗi biến thêm 100. ....	4
<b>Bài 1.4.</b> Viết hàm <code>countEven(int*, int)</code> nhận một mảng số nguyên và kích thước của mảng, trả về số lượng số chẵn trong mảng???	6
<b>Bài 1.5.</b> Viết hàm trả về con trỏ trỏ tới giá trị lớn nhất của một mảng các số double. Nếu mảng rỗng hãy trả về NULL. ....	7
<b>Bài 1.6.</b> Viết hàm đảo ngược một mảng các số nguyên theo hai cách: dùng chỉ số và dùng con trỏ. ....	8
<b>Bài 1.7.</b> Viết chương trình nhập vào một mảng các số nguyên với số lượng các phần tử nhập từ bàn phím. Sau đó sắp xếp mảng theo thứ tự tăng dần. Hiển thị danh sách mảng trước và sau khi sắp xếp. .	9
<b>Bài 1.8.</b> Viết chương trình nhập vào một ma trận 2 chiều kích thước $m \times n$ với m và n nhập từ bàn phím. Sau đó đưa ra tổng các phần tử chẵn của ma trận đó. ....	11
<b>Bài 1.9.</b> Viết chương trình in ra tất cả các dãy con của một dãy cho trước. ....	12
<b>Bài 1.10.</b> Viết chương trình nhập vào 2 ma trận vuông cùng kích thước $n \times n$ , trong đó n nhập từ bàn phím. Sau đó tính tổng và tích của hai ma trận đó và đưa kết quả ra màn hình. Yêu cầu sử dụng cấp phát động để cấp phát bộ nhớ cho các ma trận. ....	16

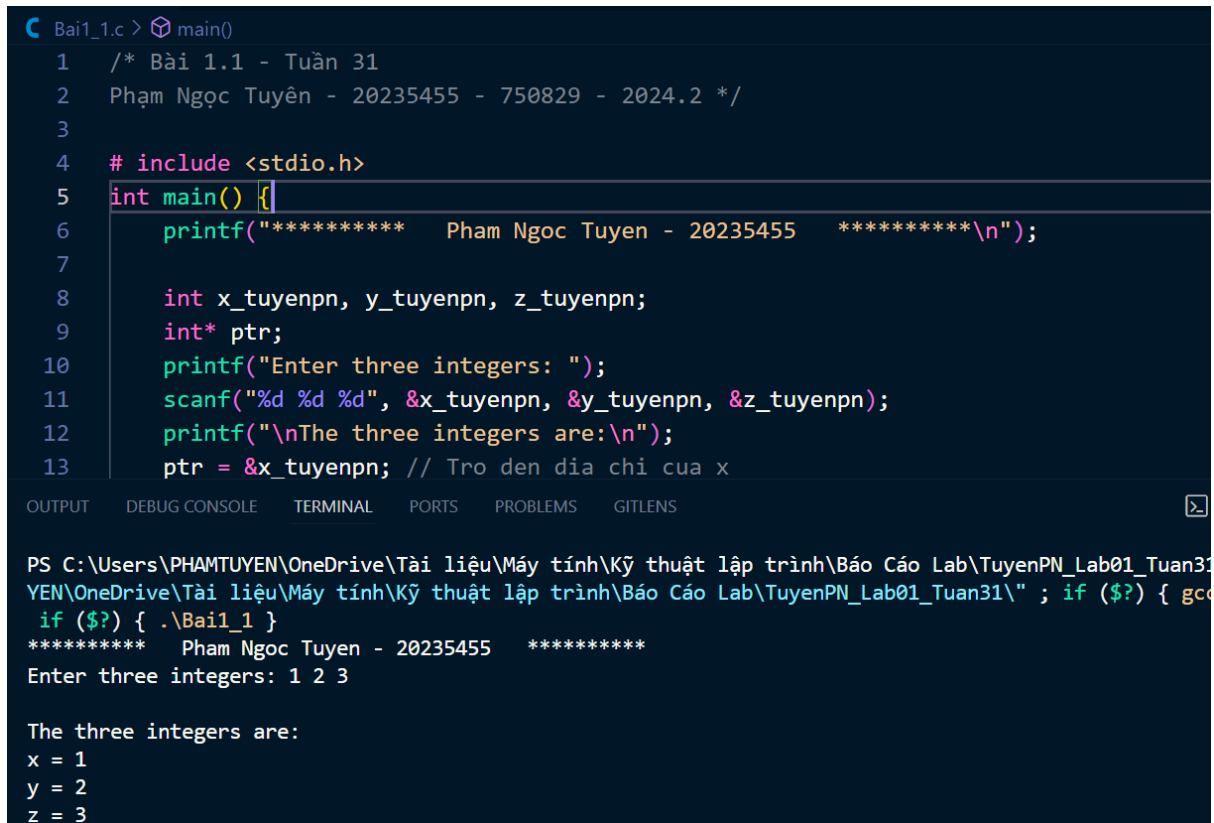
## Figures

<b>Hình 1. 1:</b> Đề bài bài 1.1 .....	<b>Error! Bookmark not defined.</b>
<b>Hình 1. 2:</b> Code bài 1.2 .....	2
<b>Hình 1.3:</b> Test case bài 1.3 .....	3
<b>Hình 2. 1:</b> Code và testcase bài 2 .....	4
<b>Hình 3. 1:</b> Code bài 1.3 .....	5
<b>Hình 3. 2:</b> Testcase bài 1.3 .....	5
<b>Hình 4. 1:</b> Code bài 1.4 .....	6
<b>Hình 4. 2:</b> Testcase bài 1.4 .....	6
<b>Hình 5. 1:</b> code bài 1.5`7 .....	
<b>Hình 5. 2:</b> Test case bài 1.5 .....	7
<b>Hình 6. 1:</b> Code bài 1.6 .....	8
<b>Hình 6. 2:</b> Test case bài 1.6 .....	9
<b>Hình 7.1:</b> Code bài 1.7 .....	10
<b>Hình 7. 2:</b> Test case bài 1.7 .....	10
<b>Hình 8. 1:</b> Code bài 1.8 .....	11
<b>Hình 8. 2:</b> Test case bài 1.8. ....	12
<b>Hình 9 1:</b> Code và test case bài 1.9 .....	15
<b>Hình 10. 1:</b> Code bài 1.10 .....	<b>Error! Bookmark not defined.</b>
<b>Hình 10. 2:</b> Test case bài 1.10. ....	18

## Bài thực hành số 1 – Tuần 7

**Bài 1.1.** Viết một chương trình C nhập vào 3 số nguyên. Thiết lập một con trỏ để lần lượt trỏ tới từng số nguyên và hiển thị kết quả giá trị tham chiếu ngược của con trỏ.

**Lưu ý:** Phép toán & trả về địa chỉ của biến.



```
Bai1_1.c > main()
1  /* Bài 1.1 - Tuần 31
2  Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
3
4  # include <stdio.h>
5  int main() {
6      printf("*****   Pham Ngoc Tuyen - 20235455   *****\n");
7
8      int x_tuyenpn, y_tuyenpn, z_tuyenpn;
9      int* ptr;
10     printf("Enter three integers: ");
11     scanf("%d %d %d", &x_tuyenpn, &y_tuyenpn, &z_tuyenpn);
12     printf("\nThe three integers are:\n");
13     ptr = &x_tuyenpn; // Trỏ đến địa chỉ của x

OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  PROBLEMS  GITLENS

PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_Lab01_Tuan31\
YEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_Lab01_Tuan31> gcc Bai1_1.c
if ($?) { gcc Bai1_1.c }
*****   Pham Ngoc Tuyen - 20235455   *****
Enter three integers: 1 2 3

The three integers are:
x = 1
y = 2
z = 3
```

Hình 1. 1: Code bài 1.2

## Phạm Ngọc Tuyên – 20235455

```
15 ptr = &y_tuyenpn; // Tro den dia chi cua y
16 printf("y = %d\n", *ptr);
17 ptr = &z_tuyenpn; // Tro den dia chi cua z
18 printf("z = %d\n", *ptr);
19 return 0;
20 }
21
22
23 /* Bài 1.1 - Tuần 9
24 Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
25
26
```

Thời gian còn lại 23:56:49

Chấm thử Chấm điểm

	Dữ liệu đầu vào	Kết quả đúng	Kết quả chương trình	
✓	3 4 5	Enter three integers: The three integers are: x = 3 y = 4 z = 5	Enter three integers: The three integers are: x = 3 y = 4 z = 5	✓
✓	133 24 5	Enter three integers: The three integers are: x = 133 y = 24 z = 5	Enter three integers: The three integers are: x = 133 y = 24 z = 5	✓

Hoàn thành được tất cả các bộ mẫu ✓

Hình 1.2: Test case bài 1.3

**Bài 1.2.** Viết chương trình in ra địa chỉ của 5 phần tử đầu tiên trong mảng được định nghĩa sau đây: `int a[7]= {13, -355, 235, 47, 67, 943, 1222};`

**Lưu ý:**

**Để in địa chỉ con trỏ các bạn sử dụng ký tự định dạng %p**

**Để lấy địa chỉ của một biến ta có thể dùng phép toán &**

```

C Bai1_2.c > main()
1  /* Bài 1.2 - Tuần 31
2  Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
3
4  #include <stdio.h>
5  int main() {
6      printf("*****  Pham Ngoc Tuyen - 20235455  *****\n");
7
8      int a_tuyenpn[7] = { 13, -355, 235, 47, 67, 943, 1222 };
9      printf("address of first five elements in memory.\n");
10     for (int i=0; i<5;i++) printf("\ta[%d] \t",i);
11     printf("\n");
12
13     /*****  Phạm Ngọc Tuyên - 20235455  *****/

```

OUTPUT    DEBUG CONSOLE    **TERMINAL**    PORTS    PROBLEMS    GITLENS

```

PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN
YEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_Lab01_Tuan31\>
if ($?) { .\Bai1_2 }
*****  Pham Ngoc Tuyen - 20235455  *****
address of first five elements in memory.
a[0]      a[1]      a[2]      a[3]      a[4]
0061FEFC  0061FF00  0061FF04  0061FF08  0061FF0C

```

Hình 2. 1: Code và testcase bài 2

**Bài 1.3.** Viết chương trình yêu cầu nhập giá trị cho 3 biến số nguyên x, y, z kiểu int. Sau đó sử dụng duy nhất một con trỏ để cộng giá trị của mỗi biến thêm 100.

**Bài 1.3.** Viết chương trình yêu cầu nhập giá trị cho 3 biến số nguyên x, y, z kiểu int. Sau đó sử dụng duy nhất một con trỏ để cộng giá trị của mỗi biến thêm 100.

```
Bai1_3.c > main()
1  /* Bài 1.3 - Tuần 31
2  Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
3
4  #include <stdio.h>
5  int main()
6  {
7      printf("*****   Pham Ngoc Tuyen - 20235455   *****\n");
8
9      int x_tuyenpn, y_tuyenpn, z_tuyenpn;
10     int *ptr;
11     scanf("%d %d %d", &x_tuyenpn, &y_tuyenpn, &z_tuyenpn);
12     printf("Here are the values of x, y, and z:\n");
13     printf("Terminal (Ctrl+)", x_tuyenpn, y_tuyenpn, z_tuyenpn);
```

OUTPUT    DEBUG CONSOLE    **TERMINAL**    PORTS    PROBLEMS    GITLENS

PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN\_Lab01\_Tuan31\YEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN\_Lab01\_Tuan31\" ; if (\$?) { g  
if (\$?) { .\Bai1\_3 }  
\*\*\*\*\* Pham Ngoc Tuyen - 20235455 \*\*\*\*\*  
25 50 75  
Here are the values of x, y, and z:  
25 50 75  
Once again, here are the values of x, y, and z:  
125 150 175

Hình 3. 1: Code bài 1.3

25

26 ▾ /\* Bài 1.3 - Tuần 31

27 Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 \*/

28

29

30

Chấm thử

Chấm điểm

	Dữ liệu đầu vào	Kết quả đúng	Kết quả chương trình
✓	25 50 75	Here are the values of x, y, and z: 25 50 75 Once again, here are the values of x, y, and z: 125 150 175	Here are the values of x, y, and z: 25 50 75 Once again, here are the values of x, 125 150 175
✓	125 150 185	Here are the values of x, y, and z: 125 150 185 Once again, here are the values of x, y, and z: 225 250 285	Here are the values of x, y, and z: 125 150 185 Once again, here are the values of x, 225 250 285
Hoàn thành được tất cả các bộ mẫu ✓			

Hình 3. 2: Test case bài 1.3

**Bài 1.4.** Viết hàm `countEven(int*, int)` nhận một mảng số nguyên và kích thước của mảng, trả về số lượng số chẵn trong mảng???

```

G+ Bai1_4.cpp > main()
1  /* Bài 1.4 - Tuần 31
2  Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
3  #include <bits/stdc++.h>
4  using namespace std;
5  int counteven(int* arr, int size) {
6      int count = 0;
7      /***** Phạm Ngọc Tuyên - 20235455 *****/
8      int* ptr_tuyenpn = arr;
9      for (int i_tuyenpn = 0; i_tuyenpn < size; i_tuyenpn++) {
10         if(*(ptr_tuyenpn + i_tuyenpn) % 2 == 0) count++;
11     }
12     return count;
13 }

OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  PROBLEMS  GITLENS

PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_Lab01_Tuan31\
YEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_Lab01_Tuan31\" ; if ($?) { g++
; if ($?) { .\Bai1_4 }
*****  Phạm Ngọc Tuyên - 20235455  *****
5
    
```

Hình 4. 1: Code bài 1.4

14  
15  
16

/\* Bài 1.4 - Tuần 31  
Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 \*/

Chấm thử
Chấm điểm

	Test	Kết quả đúng	Kết quả chương trình	
✓	int arr[] = {1, 5, 4, 8, 10, 6, 7, 2}; cout << counteven(arr, 8);	5	5	✓
✓	int arr[] = {1, 5, 4, 0, 10, 7}; cout << counteven(arr, 6);	3	3	✓

Hoàn thành được tất cả các bộ mẫu ✓

Hình 4. 2: Testcase bài 1.4

**Bài 1.5.** Viết hàm trả về con trỏ tới giá trị lớn nhất của một mảng các số double. Nếu mảng rỗng hãy trả về NULL.

```

C++ Bai1_5.cpp > maximum(double *, int)
1  /* Bài 1.5 - Tuần 31
2  Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
3  #include <bits/stdc++.h>
4  using namespace std;
5  double* maximum(double* a, int size_tuyenpn) {
6      double *max;
7      max = a;
8      if (a==NULL) return NULL;
9
10     /***** Phạm Ngọc Tuyên - 20235455 *****/
11
12     for(int i_tuyenpn = 0; i_tuyenpn < size_tuyenpn; i_tuyenpn++) {
13         if (*(max + i_tuyenpn) > *max) max = max + i_tuyenpn;
14     }
15     return max;
16 }
17 /* Bài 1.5 - Tuần 31
18 Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */

```

Hình 5. 1: code bài 1.5

11 for(int i\_tuyenpn = 0; i\_tuyenpn < size\_tuyenpn; i\_tuyenpn++)  
12 | if (\*(max + i\_tuyenpn) > \*max) max = max + i\_tuyenpn;  
13 }  
14 return max;  
15 }  
16 /\* Bài 1.5 - Tuần 31  
17 Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 \*/

Thời gian còn lại 23:20:49

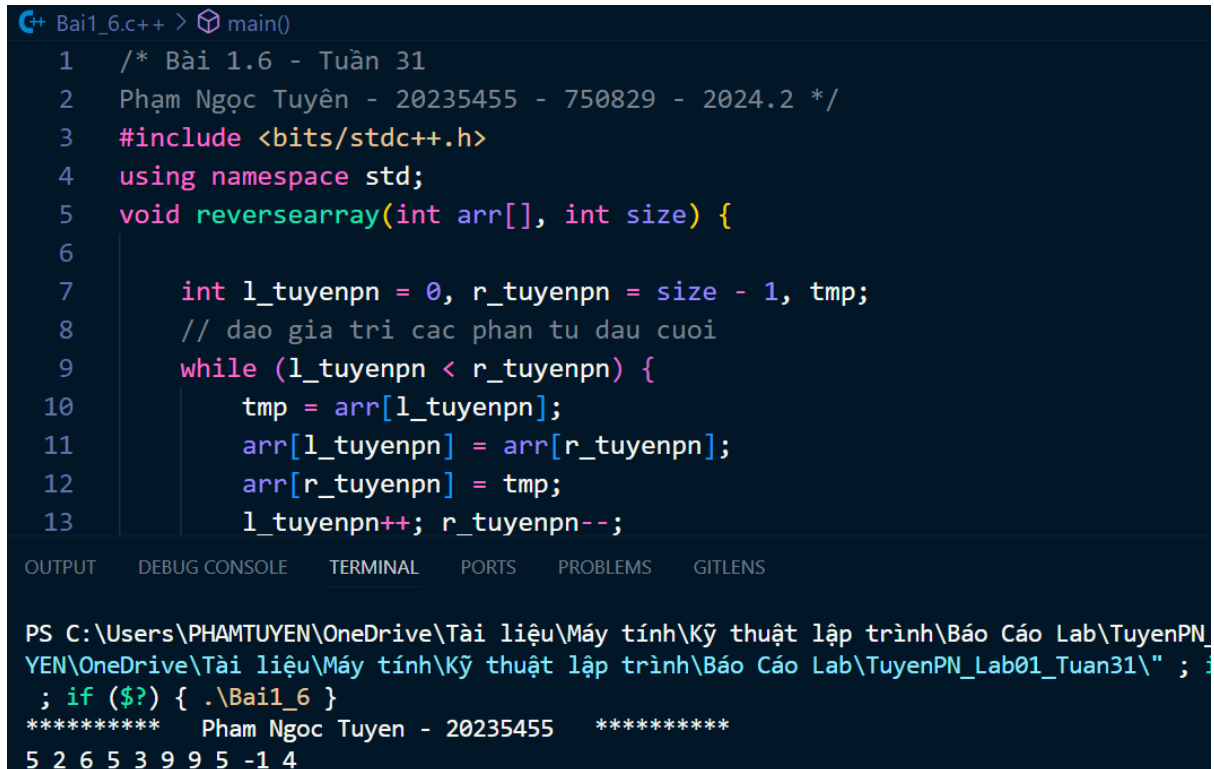
Chấm thử
Chấm điểm

	Test	Kết quả đúng	Kết quả chương trình	
✓	double arr[] = {1., 10., 2., -7., 25., 3.}; double* max = maximum(arr, 6); printf("%.0f", *max);	25	25	✓
✓	double* arr = NULL; double* max = maximum(arr, 6); printf("%.d", max==NULL?1:0);	1	1	✓
✓	double arr[] = {12.}; double* max = maximum(arr, 1); printf("%.0f", *max);	12	12	✓

Hoàn thành được tất cả các bộ mẫu ✓

Hình 5. 2: Test case bài 1.5

**Bài 1.6.** Viết hàm đảo ngược một mảng các số nguyên theo hai cách: dùng chỉ số và dùng con trỏ.



```
1  /* Bài 1.6 - Tuần 31
2  Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
3  #include <bits/stdc++.h>
4  using namespace std;
5  void reversearray(int arr[], int size) {
6
7      int l_tuyenpn = 0, r_tuyenpn = size - 1, tmp;
8      // dao gia tri cac phan tu dau cuoi
9      while (l_tuyenpn < r_tuyenpn) {
10         tmp = arr[l_tuyenpn];
11         arr[l_tuyenpn] = arr[r_tuyenpn];
12         arr[r_tuyenpn] = tmp;
13         l_tuyenpn++; r_tuyenpn--;
14     }
15 }
```

OUTPUT DEBUG CONSOLE TERMINAL PORTS PROBLEMS GITLENS

```
PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_YEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_Lab01_Tuan31\" ; if ($?) { .\Bai1_6 }
***** Phạm Ngọc Tuyên - 20235455 *****
5 2 6 5 3 9 9 5 -1 4
```

Hình 6. 1: Code bài 1.6



```

30 }
31 /* Bài 1.6 - Tuần 31
32 Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
33
34

```

Chấm thử Chấm điểm

	Test	Kết quả đúng	Kết quả chương trình	
✓	<pre> int arr[] = {9, 3, 5, 6, 2, 5}; reversearray(arr, 6); for(int i = 0; i &lt; 6; i++) cout &lt;&lt; arr[i] &lt;&lt; " "; int arr2[] = {4, -1, 5, 9}; ptr_reversearray(arr2, 4); for(int i = 0; i &lt; 4; i++) cout &lt;&lt; arr2[i] &lt;&lt; " "; </pre>	5 2 6 5 3 9 9 5 -1 4	5 2 6 5 3 9 9 5 -1 4	✓
✓	<pre> int arr[] = {9, 6, 2, 5}; reversearray(arr, 4); for(int i = 0; i &lt; 4; i++) cout &lt;&lt; arr[i] &lt;&lt; " "; int arr2[] = {12, 4, -1, 5, 9}; ptr_reversearray(arr2, 5); for(int i = 0; i &lt; 5; i++) cout &lt;&lt; arr2[i] &lt;&lt; " "; </pre>	5 2 6 9 9 5 -1 4 12	5 2 6 9 9 5 -1 4 12	✓
Hoàn thành được tất cả các bộ mẫu ✓				

Hình 6. 2: Test case bài 1.6

**Bài 1.7.** Viết chương trình nhập vào một mảng các số nguyên với số lượng các phần tử nhập từ bàn phím. Sau đó sắp xếp mảng theo thứ tự tăng dần. Hiển thị danh sách mảng trước và sau khi sắp xếp.

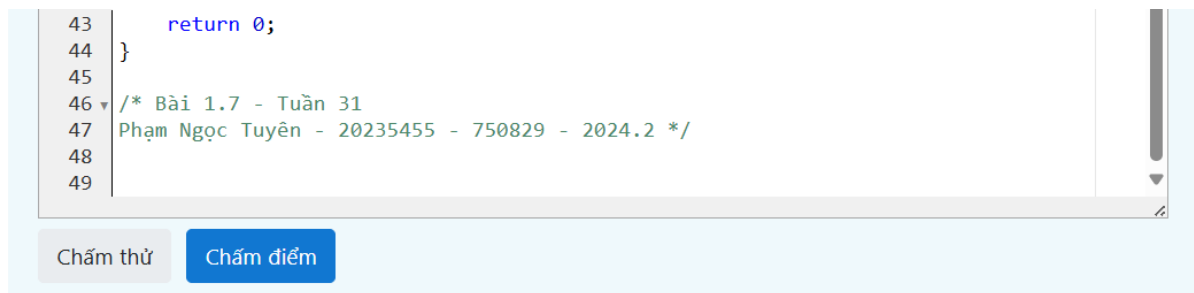
Yêu cầu chỉ sử dụng con trỏ để truy cập mảng, không truy cập theo index mảng.

```

C Bai1_7.c > ...
1  /* Bài 1.7 - Tuần 31
2  Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
3
4  #include <stdio.h>
5  #include <stdlib.h>
6  int *a;
7  int n, tmp;
8
9  int main() {
10     printf("*****   Phạm Ngọc Tuyên - 20235455   *****\n");
11     printf("Enter the number of elements: ");
12     scanf("%d", &n);
13
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  PROBLEMS  GITLENS
PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_Lab0
YEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_Lab01_Tuan31\" ; if ($
if ($?) { .\Bai1_7 }
*****   Phạm Ngọc Tuyên - 20235455   *****
Enter the number of elements: 5
9 -2 1 2 16
The input array is:
9 -2 1 2 16
The sorted array is:
-2 1 2 9 16

```

Hình 7.1: Code bài 1.7



	Dữ liệu đầu vào	Kết quả đúng	Kết quả chương trình
✓	5 9 -2 1 2 15	Enter the number of elements: The input array is: 9 -2 1 2 15 The sorted array is: -2 1 2 9 15	Enter the number of elements: The input array is: 9 -2 1 2 15 The sorted array is: -2 1 2 9 15
✓	10 9 -2 1 2 15 3 7 8 20 -5	Enter the number of elements: The input array is: 9 -2 1 2 15 3 7 8 20 -5 The sorted array is: -5 -2 1 2 3 7 8 9 15 20	Enter the number of elements: The input array is: 9 -2 1 2 15 3 7 8 20 -5 The sorted array is: -5 -2 1 2 3 7 8 9 15 20
Hoàn thành được tất cả các bộ mẫu ✓			

Hình 7. 2: Test case bài 1.7

**Bài 1.8.** Viết chương trình nhập vào một ma trận 2 chiều kích thước  $m \times n$  với  $m$  và  $n$  nhập từ bàn phím. Sau đó đưa ra tổng các phần tử chẵn của ma trận đó.

**Lưu ý:** Khi viết hàm cấp phát bộ nhớ cho một ma trận hai chiều biểu diễn bởi con trỏ `int **mt`, nếu ta truyền con trỏ theo kiểu địa chỉ `void allocate_mem(int **mt, int m, int n)` sẽ dẫn tới việc cấp phát bộ nhớ cho một bản sao của con trỏ `**mt`. Do đó, sau khi gọi hàm thì con trỏ `**mt` gốc vẫn không được cấp phát bộ nhớ. Để cấp phát thành công cần truyền con trỏ theo dạng địa chỉ, ví dụ sử dụng con trỏ cấp 3 dạng `int ***mt`.

```
C Bai1_8.c > main()
1  /* Bài 1.8 - Tuần 31
2  Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
3
4  #include <stdio.h>
5  #include <stdlib.h>
6
7  void allocate_mem(int ***mt, int m, int n){
8      // #Allocate memory for the matrix
9
10     /***** Phạm Ngọc Tuyên - 20235455 *****/
11     // Cấp phát bộ nhớ chung cho ma trận
12     *mt = (int**) malloc (m * sizeof(int*));
13
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  PROBLEMS  GITLENS
YEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\TuyenPN_Lab01_Tuan31\" ; if ($?) { gcc B
if ($?) { .\Bai1_8 }
***** Phạm Ngọc Tuyên - 20235455 *****
Enter m, n = 2 2
1 2
3 4
mt[0][0] = 1
mt[0][1] = 2
mt[1][0] = 3
mt[1][1] = 4
The sum of all even elements is 6
```

Hình 8. 1: Code bài 1.8

```

93     free_mem(mt, m, n);
94     return 0;
95 }
96 /* Bài 1.8 - Tuần 31
97 Phạm Ngọc Tuyên - 20235455 - 750829 - 2024.2 */
98

```

Chấm thử Chấm điểm

	Dữ liệu đầu vào	Kết quả đúng	Kết quả chương trình
✓	2 2 1 2 4 5	Enter m, n = mt[0][0] = mt[0][1] = mt[1][0] = mt[1][1] = 1 2 4 5 The sum of all even elements is 6	Enter m, n = mt[0][0] = mt[0][1] 4 5 The sum of all even elements is
Hoàn thành được tất cả các bộ mẫu ✓			

Hình 8. 2: Test case bài 1.8.

**Bài 1.9.** Viết chương trình in ra tất cả các dãy con của một dãy cho trước.

## Phạm Ngọc Tuyền – 20235455

```
Bai1_9.c > main()
6 void process(int* array_tuyenpn, int n_tuyenpn) {
  for(int j_tuyenpn = 1_tuyenpn; j_tuyenpn < n_tuyenpn; j_tuyenpn++)
  14 }
15 /***** Phạm Ngọc Tuyền - 20235455 *****/
16 int main() {
17 // freopen("input.txt", "r", stdin);
18 freopen("output.txt", "w", stdout);
19 printf("***** Phạm Ngọc Tuyền - 20235455 *****\n");
20 int test_tuyenpn;
21 scanf("%d", &test_tuyenpn);
22 printf("%d\n", test_tuyenpn);
23 int n;
24 scanf("%d", &n);
25 printf("input = %d\n", n);
26 printf("output : \n");
27 int* a = (int*)malloc(n * (sizeof(int)));
28 for (int i = 0; i < n; i++) scanf("%d", (a + i));
29 process(a,n);
30 return 0;
31 }
32
```

```
output.txt
1 ***** Phạm Ngọc Tuyền - 20235455 *****
2 case = 1
3 input = 2
4 output :
5 3
6 3 2
7 2
8
```

PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\sourcecode> cd "c:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\sourcecode"; if (\$?) { gcc Bai1\_9.c -o Bai1\_9 }; if (\$?) { .\Bai1\_9 }

```
1
2
3 2
```

```
Bai1_9.c > main()
6 void process(int* array_tuyenpn, int n_tuyenpn) {
  for(int j_tuyenpn = 1_tuyenpn; j_tuyenpn < n_tuyenpn; j_tuyenpn++)
  14 }
15 /***** Phạm Ngọc Tuyền - 20235455 *****/
16 int main() {
17 // freopen("input.txt", "r", stdin);
18 freopen("output.txt", "w", stdout);
19 printf("***** Phạm Ngọc Tuyền - 20235455 *****\n");
20 int test_tuyenpn;
21 scanf("%d", &test_tuyenpn);
22 printf("%d\n", test_tuyenpn);
23 int n;
24 scanf("%d", &n);
25 printf("input = %d\n", n);
26 printf("output : \n");
27 int* a = (int*)malloc(n * (sizeof(int)));
28 for (int i = 0; i < n; i++) scanf("%d", (a + i));
29 process(a,n);
30 return 0;
31 }
32
```

```
output.txt
1 ***** Phạm Ngọc Tuyền - 20235455 *****
2 case = 2
3 input = 3
4 output :
5 5
6 5 1
7 5 1 4
8 1
9 1 4
10 4
11
```

PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\sourcecode> cd "c:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\sourcecode"; if (\$?) { gcc Bai1\_9.c -o Bai1\_9 }; if (\$?) { .\Bai1\_9 }

```
2
3
5 1 4
PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\sourcecode>
```

## Phạm Ngọc Tuyên – 20235455

```
C Bai_9.c > main()
6 void process(int* array_tuyenpn, int n_tuyenpn) {
  int i;
  for (i = 0; i < n_tuyenpn; i++) {
    array_tuyenpn[i] = array_tuyenpn[i] * 2;
  }
}
15 /***** Phạm Ngọc Tuyên - 20235455 *****/
16 int main() {
17   // freopen("input.txt", "r", stdin);
18   freopen("output.txt", "w", stdout);
19   printf("***** Phạm Ngọc Tuyên - 20235455 *****\n");
20   int test_tuyenpn;
21   scanf("%d", &test_tuyenpn);
22   printf("%d\n", test_tuyenpn);
23   int n;
24   scanf("%d", &n);
25   printf("input = %d\n", n);
26   printf("output : \n");
27   int* a = (int*)malloc(n * (sizeof(int)));
28   for (int i = 0; i < n; i++) scanf("%d", (a + i));
29   process(a, n);
30   return 0;
31 }
32
```

```
output.txt
1 ***** Phạm Ngọc Tuyên - 20235455 *****
2 case = 3
3 input = 10
4 output :
5 1
6 1 4
7 1 4 -2
8 1 4 -2 4
9 1 4 -2 4 65
10 1 4 -2 4 65 8
11 1 4 -2 4 65 8 -9
12 1 4 -2 4 65 8 -9 12
13 1 4 -2 4 65 8 -9 12 5
14 1 4 -2 4 65 8 -9 12 5 67
15 4
16 4 -2
17 4 -2 4
18 4 -2 4 65
19 4 -2 4 65 8
20 4 -2 4 65 8 -9
21 4 -2 4 65 8 -9 12
```

```
PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\sourcecode> cd "c:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\sourcecode\"; if ($?) { gcc Bai_9.c -o Bai_9 }; if ($?) { .\Bai_9 }
3
10
1 4 -2 4 65 8 -9 12 5 67
```

```
C Bai_9.c > main()
6 void process(int* array_tuyenpn, int n_tuyenpn) {
  int i;
  for (i = 0; i < n_tuyenpn; i++) {
    array_tuyenpn[i] = array_tuyenpn[i] * 2;
  }
}
15 /***** Phạm Ngọc Tuyên - 20235455 *****/
16 int main() {
17   // freopen("input.txt", "r", stdin);
18   freopen("output.txt", "w", stdout);
19   printf("***** Phạm Ngọc Tuyên - 20235455 *****\n");
20   int test_tuyenpn;
21   scanf("%d", &test_tuyenpn);
22   printf("%d\n", test_tuyenpn);
23   int n;
24   scanf("%d", &n);
25   printf("input = %d\n", n);
26   printf("output : \n");
27   int* a = (int*)malloc(n * (sizeof(int)));
28   for (int i = 0; i < n; i++) scanf("%d", (a + i));
29   process(a, n);
30   return 0;
31 }
32
```

```
output.txt
1 ***** Phạm Ngọc Tuyên - 20235455 *****
2 case = 4
3 input = 20
4 output :
5 92
6 92 94
7 92 94 85
8 92 94 85 15
9 92 94 85 15 57
10 92 94 85 15 57 12
11 92 94 85 15 57 12 18
12 92 94 85 15 57 12 18 19
13 92 94 85 15 57 12 18 19 18
14 92 94 85 15 57 12 18 19 18 45
15 92 94 85 15 57 12 18 19 18 45 49
16 92 94 85 15 57 12 18 19 18 45 49 55
17 92 94 85 15 57 12 18 19 18 45 49 55 28
18 92 94 85 15 57 12 18 19 18 45 49 55 28 36
19 92 94 85 15 57 12 18 19 18 45 49 55 28 36 16
```

```
PS C:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\sourcecode> cd "c:\Users\PHAMTUYEN\OneDrive\Tài liệu\Máy tính\Kỹ thuật lập trình\Báo Cáo Lab\sourcecode\"; if ($?) { gcc Bai_9.c -o Bai_9 }; if ($?) { .\Bai_9 }
4
20
92 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70 39 39
```

## Phạm Ngọc Tuyen – 20235455

```
output.txt
1 ***** Pham Ngoc Tuyen - 20235455 *****
2 case = 3
3 input = 10
4 output :
5 1
6 1 4
7 1 4 -2
8 1 4 -2 4
9 1 4 -2 4 65
10 1 4 -2 4 65 8
11 1 4 -2 4 65 8 -9
12 1 4 -2 4 65 8 -9 12
13 1 4 -2 4 65 8 -9 12 5
14 1 4 -2 4 65 8 -9 12 5 67
15 4
16 4 -2
17 4 -2 4
18 4 -2 4 65
19 4 -2 4 65 8
20 4 -2 4 65 8 -9
21 4 -2 4 65 8 -9 12
22 4 -2 4 65 8 -9 12 5
23 4 -2 4 65 8 -9 12 5 67
24 -2
25 -2 4
26 -2 4 65
27 -2 4 65 8
28 -2 4 65 8 -9
29 -2 4 65 8 -9 12
30 -2 4 65 8 -9 12 5
31 -2 4 65 8 -9 12 5 67
32 4
33 4 65
34 4 65 8
35 4 65 8 -9
36 4 65 8 -9 12
37 4 65 8 -9 12 5
38 4 65 8 -9 12 5 67
39 65
40 65 8
41 65 8 -9
42 65 8 -9 12
43 65 8 -9 12 5
44 65 8 -9 12 5 67
45 8
46 8 -9
47 8 -9 12
48 8 -9 12 5
49 8 -9 12 5 67
50 -9
51 -9 12
52 -9 12 5
53 -9 12 5 67
54 12
55 12 5
56 12 5 67
57 5
58 5 67
59 67
60
```

```
output.txt
1 ***** Pham Ngoc Tuyen - 20235455 *****
2 case = 4
3 input = 20
4 output :
5 92
6 92 94
7 92 94 85
8 92 94 85 15
9 92 94 85 15 57
10 92 94 85 15 57 12
11 92 94 85 15 57 12 18
12 92 94 85 15 57 12 18 19
13 92 94 85 15 57 12 18 19 18
14 92 94 85 15 57 12 18 19 18 45
15 92 94 85 15 57 12 18 19 18 45 49
16 92 94 85 15 57 12 18 19 18 45 49 55
17 92 94 85 15 57 12 18 19 18 45 49 55 28
18 92 94 85 15 57 12 18 19 18 45 49 55 28 36
19 92 94 85 15 57 12 18 19 18 45 49 55 28 36 16
20 92 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100
21 92 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90
22 92 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70
23 92 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70 39
24 92 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70 39 39
25 94
26 94 85
27 94 85 15
28 94 85 15 57
29 94 85 15 57 12
30 94 85 15 57 12 18
31 94 85 15 57 12 18 19
32 94 85 15 57 12 18 19 18
33 94 85 15 57 12 18 19 18 45
34 94 85 15 57 12 18 19 18 45 49
35 94 85 15 57 12 18 19 18 45 49 55
36 94 85 15 57 12 18 19 18 45 49 55 28
37 94 85 15 57 12 18 19 18 45 49 55 28 36
38 94 85 15 57 12 18 19 18 45 49 55 28 36 16
39 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100
40 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90
41 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70
42 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70 39
43 94 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70 39 39
44 85
45 85 15
46 85 15 57
47 85 15 57 12
48 85 15 57 12 18
49 85 15 57 12 18 19
50 85 15 57 12 18 19 18
51 85 15 57 12 18 19 18 45
52 85 15 57 12 18 19 18 45 49
53 85 15 57 12 18 19 18 45 49 55
54 85 15 57 12 18 19 18 45 49 55 28
55 85 15 57 12 18 19 18 45 49 55 28 36
56 85 15 57 12 18 19 18 45 49 55 28 36 16
57 85 15 57 12 18 19 18 45 49 55 28 36 16 100
58 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90
59 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70
60 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70 39
61 85 15 57 12 18 19 18 45 49 55 28 36 16 100 90 70 39 39
62 15
```

Hình 9 1: Code và test case bài 1.9

**Bài 1.10.** Viết chương trình nhập vào 2 ma trận vuông cùng kích thước  $n \times n$ , trong đó  $n$  nhập từ bàn phím. Sau đó tính tổng và tích của hai ma trận đó và đưa kết quả ra màn hình. Yêu cầu sử dụng cấp phát động để cấp phát bộ nhớ cho các ma trận.

The screenshot shows a C++ IDE with a code editor, a terminal, and an output window. The code defines a `multMatrix` function and a `main` function. In `main`, `n` is set to 2, and two 2x2 matrices are input. The program calculates the sum and product of these matrices. The terminal shows the execution command, and the output window displays the results for `case = 1` with `input = 2`.

```
1 ***** Pham Ngoc Tuyen - 20235455 *****
2 case = 1
3 input = 2
4 output =
5 ----- Sum of Matrix -----
6 1 5
7 8 9
8 ----- Mult of Matrix -----
9 1 14
10 4 38
11
```

The screenshot shows the same C++ IDE with `n` set to 3. The code is identical to the previous one, but the input and output are for 3x3 matrices. The terminal shows the execution command, and the output window displays the results for `case = 2` with `input = 3`.

```
1 ***** Pham Ngoc Tuyen - 20235455 *****
2 case = 2
3 input = 3
4 output =
5 ----- Sum of Matrix -----
6 8 0 15
7 3 7 10
8 8 18 14
9 ----- Mult of Matrix -----
10 45 63 89
11 19 36 40
12 62 113 143
13
```



## Phạm Ngọc Tuyên – 20235455

```
C Bai1_10.c > ...
47 int** multMatrix(int **mt1, int **mt2, int n){
62
63 void free_mem(int **mt, int n){
64     for (int i_tuyenpn = 0; i_tuyenpn < n; i_tuyenpn++)
65         free(mt[i_tuyenpn]);
66     free(mt);
67 }
68 /***** Phạm Ngọc Tuyên - 20235455 *****/
69 int main() {
70     freopen("output.txt", "w", stdout);
71     printf("***** Phạm Ngọc Tuyên - 20235455 *****\n");
72     int n, ** mt1, ** mt2, test_tuyenpn;
73     scanf("%d", &test_tuyenpn);
74     printf("%d\n", test_tuyenpn);
75     scanf("%d", &n);
76     printf("input = %d\n", n);
77     allocate_mem(&mt1, n);
78     allocate_mem(&mt2, n);
79
80
81     input(mt1, n);
82
83     output(mt1, mt2, n);
84 }

output.txt
1 ***** Phạm Ngọc Tuyên - 20235455 *****
2 case = 3
3 input = 7
4 output =
5 ----- Sum of Matrix -----
6 5 4 1 5 3 6 2
7 2 4 8 6 7 5 6
8 5 5 2 10 4 6 7
9 5 4 5 4 6 4 5
10 6 5 7 8 3 4 5
11 5 4 5 2 6 5 6
12 2 3 4 5 3 1 7
13 ----- Mult of Matrix -----
14 17 19 41 25 15 19 34
15 62 47 78 83 32 54 83
16 41 29 58 53 19 31 61
17 34 26 47 44 15 30 55
18 50 29 47 53 18 42 48
19 45 38 56 49 19 32 62
20 27 21 39 39 15 25 42
21
```

```
C Bai1_10.c > ...
47 int** multMatrix(int **mt1, int **mt2, int n){
62
63 void free_mem(int **mt, int n){
64     for (int i_tuyenpn = 0; i_tuyenpn < n; i_tuyenpn++)
65         free(mt[i_tuyenpn]);
66     free(mt);
67 }
68 /***** Phạm Ngọc Tuyên - 20235455 *****/
69 int main() {
70     freopen("output.txt", "w", stdout);
71     printf("***** Phạm Ngọc Tuyên - 20235455 *****\n");
72     int n, ** mt1, ** mt2, test_tuyenpn;
73     scanf("%d", &test_tuyenpn);
74     printf("%d\n", test_tuyenpn);
75     scanf("%d", &n);
76     printf("input = %d\n", n);
77     allocate_mem(&mt1, n);
78     allocate_mem(&mt2, n);
79
80
81     input(mt1, n);
82
83     output(mt1, mt2, n);
84 }

output.txt
1 ***** Phạm Ngọc Tuyên - 20235455 *****
2 case = 4
3 input = 10
4 output =
5 ----- Sum of Matrix -----
6 75 137 68 92 65 100 145 99 103 182
7 44 150 134 45 120 150 72 95 133 164
8 22 65 144 129 74 62 129 140 41 48
9 67 102 59 75 24 114 163 77 66 151
10 118 122 60 166 100 106 116 99 143 115
11 87 103 37 39 115 172 156 75 110 164
12 46 117 126 114 136 52 123 88 180 107
13 103 149 110 112 54 96 110 49 96 120
14 78 110 38 90 111 53 41 134 90 132
15 105 116 153 79 101 141 112 92 94 122
16 ----- Mult of Matrix -----
17 21382 29723 24124 20954 26397 29918 27265 25457 28350 3246
18 21683 29432 22452 24803 29076 34127 27543 26666 31160 3194
19 16399 21213 21643 22126 20399 24994 24038 20295 22911 2437
20 21480 30091 22331 21421 25670 34932 27251 23402 29239 3877
21 18763 29027 23508 21882 24115 29492 29341 24285 25016 3537
```

```

output.txt X
output.txt
1  ***** Pham Ngoc Tuyen - 20235455 *****
2  case = 4
3  input = 10
4  output =
5  ----- Sum of Matrix -----
6  75 137 68 92 65 100 145 99 103 182
7  44 150 134 45 120 150 72 95 133 164
8  22 65 144 129 74 62 129 140 41 48
9  67 102 59 75 24 114 163 77 66 151
10 118 122 60 166 100 106 116 99 143 115
11 87 103 37 39 115 172 156 75 110 164
12 46 117 126 114 136 52 123 88 180 107
13 103 149 110 112 54 96 110 49 96 120
14 78 110 38 90 111 53 41 134 90 132
15 105 116 153 79 101 141 112 92 94 122
16 ----- Mult of Matrix -----
17 21382 29723 24124 20954 26397 29918 27265 25457 28350 32407
18 21683 29432 22452 24803 29076 34127 27543 26666 31160 31948
19 16399 21213 21643 22126 20399 24994 24038 20295 22911 24376
20 21480 30091 22331 21421 25670 34932 27251 23402 29239 38729
21 18763 29027 23508 21882 24115 29492 29341 24285 25016 35322
22 24484 34041 24058 26743 30527 38822 25919 25793 35666 37604
23 22509 29970 24095 27208 27943 30250 25051 25812 30481 31168
24 18088 27900 28864 24619 25210 30139 28012 25932 28379 29771
25 14822 17333 15409 15845 15503 19723 16568 15089 18312 18700
26 21932 28325 31183 27088 27763 36325 34976 27260 32277 34374
27

```

Hình 10. 1: Test case bài 1.10.