

# Programming I – Assignment 07

## Arrays

#	Student ID	Student Name	Grade (10)
-			

Delivery Date	
---------------	--

١. يتم تسليم التمرين محلولا في خلال أسبوعا من تاريخ التمرين، و يتم حذف درجتين من التمرين عن كل أسبوع تأخير
٢. يتم التسليم لمعيد المقرر مباشرة
٣. تتم أجابه التمرين في نفس ورق الأسئلة

**Highlight the box that contains the correct answer**

#	a	b	c	d	comment
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					

#	Question
1	<p>What will happen if in a C program you assign a value to an array element whose subscript exceeds the size of array?</p> <p>A. The element will be set to 0.            B. The compiler would report an error.            C. The program may crash if some important data gets overwritten.            D. The array size would appropriately grow.</p>
2	<p>What does the following declaration mean?  <code>int (*ptr)[10];</code></p> <p>A. ptr is array of pointers to 10 integers            B. ptr is a pointer to an array of 10 integers            C. ptr is an array of 10 integers            D. ptr is an pointer to array</p>
3	<p>In C, if you pass an array as an argument to a function, what actually gets passed?</p> <p>A. Value of elements in array            B. First element of the array            C. Base address of the array            D. Address of the last element of array</p>
4	<p>What will be the output of the program ?</p> <pre>#include&lt;stdio.h&gt; int main() {     int a[5] = {5, 1, 15, 20, 25};     int i, j, m;     i = ++a[1];     j = a[1]++;     m = a[i++];     printf("%d, %d, %d", i, j, m);     return 0; }</pre> <p>A. 2, 1, 15      B. 1, 2, 5            C. 3, 2, 15      D. 2, 3, 20</p>
5	<p>What will be the output of the program ?</p>

```
#include<stdio.h>
int main()
{
    static int a[2][2] = {1, 2, 3, 4};
    int i, j;
    static int *p[] = {(int*)a, (int*)a+1, (int*)a+2};
    for(i=0; i<2; i++)
    {
        for(j=0; j<2; j++)
        {
            printf("%d, %d, %d, %d\n", (*(p+i)+j), (*(j+p)+i),
                (*(i+p)+j), (*(p+j)+i));
        }
    } return 0;}

```

A. 1, 1, 1, 1

2, 3, 2, 3

3, 2, 3, 2

4, 4, 4, 4

B. 1, 2, 1, 2

2, 3, 2, 3

3, 4, 3, 4

4, 2, 4, 2

C. 1, 1, 1, 1

2, 2, 2, 2

2, 2, 2, 2

3, 3, 3, 3

D. 1, 2, 3, 4

2, 3, 4, 1

3, 4, 1, 2

4, 1, 2, 3

6 What will be the output of the program ?

	<pre>#include&lt;stdio.h&gt;  int main() {     void fun(int, int[]);     int arr[] = {1, 2, 3, 4};     int i;     fun(4, arr);     for(i=0; i&lt;4; i++)         printf("%d,", arr[i]);     return 0; }  void fun(int n, int arr[]) {     int *p=0;     int i=0;     while(i++ &lt; n)         p = &amp;arr[i];     *p=0; }</pre> <p>A. 2, 3, 4, 5    B. 1, 2, 3, 4 C. 0, 1, 2, 3    D. 3, 2, 1 0</p>
7	<p>What will be the output of the program ?</p> <pre>#include&lt;stdio.h&gt; void fun(int **p);  int main() {     int a[3][4] = {1, 2, 3, 4, 4, 3, 2, 8, 7, 8, 9, 0};     int *ptr;     ptr = &amp;a[0][0];     fun(&amp;ptr);     return 0;</pre>

	<pre> } void fun(int **p) {     printf("%d\n", **p); } </pre> <p>A. 1      B. 2 C. 3      D. 4</p>
8	<p>What will be the output of the program ?</p> <pre> #include&lt;stdio.h&gt;  int main() {     static int arr[] = {0, 1, 2, 3, 4};     int *p[] = {arr, arr+1, arr+2, arr+3, arr+4};     int **ptr=p;     ptr++;     printf("%d, %d, %d\n", ptr-p, *ptr-arr, **ptr);     *ptr++;     printf("%d, %d, %d\n", ptr-p, *ptr-arr, **ptr);     *++ptr;     printf("%d, %d, %d\n", ptr-p, *ptr-arr, **ptr);     ++*ptr;     printf("%d, %d, %d\n", ptr-p, *ptr-arr, **ptr);     return 0; } </pre> <p>A. 0, 0, 0 1, 1, 1 2, 2, 2 3, 3, 3</p> <p>B. 1, 1, 2 2, 2, 3 3, 3, 4</p>

	<p>4, 4, 1</p> <p>C. 1, 1, 1 2, 2, 2 3, 3, 3 3, 4, 4</p> <p>D. 0, 1, 2 1, 2, 3 2, 3, 4 3, 4, 5</p>
9	<p>What will be the output of the program if the array begins at 65472 and each integer occupies 2 bytes?</p> <pre>#include&lt;stdio.h&gt;  int main() {     int a[3][4] = {1, 2, 3, 4, 4, 3, 2, 1, 7, 8, 9, 0};     printf("%u, %u\n", a+1, &amp;a+1);     return 0; }</pre> <p>A. 65474, 65476      B. 65480, 65496 C. 65480, 65488      D. 65474, 65488</p>
10	<p>What will be the output of the program in Turb C (under DOS)?</p> <pre>#include&lt;stdio.h&gt;  int main() {     int arr[5], i=0;     while(i&lt;5)         arr[i]=++i;      for(i=0; i&lt;5; i++)</pre>

	<pre>printf("%d, ", arr[i]);  return 0; }</pre> <p>A. 1, 2, 3, 4, 5,      B. Garbage value, 1, 2, 3, 4, C. 0, 1, 2, 3, 4,      D. 2, 3, 4, 5, 6,</p>
11	<p>What will be the output of the program ?</p> <pre>#include&lt;stdio.h&gt;  int main() {     int arr[1]={10};     printf("%d\n", 0[arr]);     return 0; }</pre> <p>A. 1      B. 10 C. 0      D. 6</p>
12	<p>What will be the output of the program if the array begins at address 65486?</p> <pre>#include&lt;stdio.h&gt;  int main() {     int arr[] = {12, 14, 15, 23, 45};     printf("%u, %u\n", arr, &amp;arr);     return 0; }</pre> <p>A. 65486, 65488      B. 65486, 65486 C. 65486, 65490      D. 65486, 65487</p>
13	



	<p>What will be the output of the program ?</p> <pre>#include&lt;stdio.h&gt;  int main() {     float arr[] = {12.4, 2.3, 4.5, 6.7};     printf("%d\n", sizeof(arr)/sizeof(arr[0]));     return 0; }</pre> <p>A. 5      B. 4 C. 6      D. 7</p>
14	<p>What will be the output of the program if the array begins 1200 in memory?</p> <pre>#include&lt;stdio.h&gt;  int main() {     int arr[]={2, 3, 4, 1, 6};     printf("%u, %u, %u\n", arr, &amp;arr[0], &amp;arr);     return 0; }</pre> <p>A. 1200, 1202, 1204    B. 1200, 1200, 1200 C. 1200, 1204, 1208    D. 1200, 1202, 1200</p>
15	<p>Which of the following is correct way to define the function fun() in the below program?</p> <pre>#include&lt;stdio.h&gt;  int main() {     int a[3][4];</pre>



	<pre>fun(a); return 0; }  A.  void fun(int p[][4]) { }  B.  void fun(int *p[4]) { }  C.  void fun(int *p[][4]) { }  D.  void fun(int *p[3][4]) { }</pre>
16	<p>Which of the following statements mentioning the name of the array begins DOES NOT yield the base address?</p> <ol style="list-style-type: none"><li>1: When array name is used with the sizeof operator.</li><li>2: When array name is operand of the &amp; operator.</li><li>3: When array name is passed to scanf() function.</li><li>4: When array name is passed to printf() function.</li></ol> <p>A. A    B. A, B C. B    D. B, D</p>
17	<p>Which of the following statements are correct about the program</p>

	<p>below?</p> <pre>#include&lt;stdio.h&gt;  int main() {     int size, i;     scanf("%d", &amp;size);     int arr[size];     for(i=1; i&lt;=size; i++)     {         scanf("%d", arr[i]);         printf("%d", arr[i]);     }     return 0; }</pre> <p>A. The code is erroneous since the subscript for array used in for loop is in the range 1 to size.</p> <p>B. The code is erroneous since the values of array are getting scanned through the loop.</p> <p>C. The code is erroneous since the statement declaring array is invalid.</p> <p>D. The code is correct and runs successfully.</p>
18	<p>Which of the following statements are correct about 6 used in the program?</p> <pre>int num[6]; num[6]=21;</pre> <p>A. In the first statement 6 specifies a particular element, whereas in the second statement it specifies a type.</p> <p>B. In the first statement 6 specifies a array size, whereas in the second statement it specifies a particular element of array.</p> <p>C. In the first statement 6 specifies a particular element, whereas in the second statement it specifies a array size.</p> <p>D. In both the statement 6 specifies array size.</p>
19	<p>Which of the following statements are correct about an array?</p>

- |    |  |    |     |
|----|--|----|-----|
| 1: | The array <code>int num[26]</code> ; can store 26 elements.                        |    |     |
| 2: | The expression <code>num[1]</code> designates the very first element in the array. |    |     |
| 3: | It is necessary to initialize the array at the time of declaration.                |    |     |
| 4: | The declaration <code>num[SIZE]</code> is allowed if <code>SIZE</code> is a macro. |    |     |
| A. | 1  | B. | 1,4 |
| C. | 2,3  | D. | 2,4 |