

LAB 12: ONE-DIMENSIONAL ARRAY

OBJECTIVES FOR STUDENTS

1. Write and create one-dimensional array.
[Menulis dan membina tatasusunan 1 dimensi.]
2. Write and use one-dimensional array with functions.
[Menulis tatasusunan 1 dimensi bersama fungsi.]

ASSUMPTIONS

1. Students should have knowledge in using loop statement.
[Pelajar sepatutnya mempunyai pengetahuan dalam menggunakan pernyataan gelung.]
2. Students should have knowledge in passing arguments to functions.
[Pelajar sepatutnya mempunyai pengetahuan dalam menghantar parameter kepada fungsi.]

LAB EXERCISES

EXERCISE 1:

[LATIHAN 1]

1. Write Program 12.1 and Program 12.2. Differentiate both programs.
[Tulis Program 12.1 dan Program 12.2. Huraikan perbezaan kedua-dua program.]

```
1 //Program 12.1
2 #include <stdio.h>
3 #include <conio.h>
4
5 int main()
6 {
7     float score1, score2, score3, score4;
8     printf ("Enter four floats: \n");
9     scanf ("%f", &score1);
10    scanf ("%f", &score2);
11    scanf ("%f", &score3);
12    scanf ("%f", &score4);
13    printf ("The scores in reverse order are: \n");
14    printf ("%f %f %f %f", score4, score3, score2,
15    score1;
```

```

15   getch();
16   return 0;
17 }

```

```

1 //Program 12.2
2 #include <stdio.h>
3 #include <conio.h>
4 #define SIZE 4
5
6 int main()
7 {
8     float score[SIZE];
9     int i;
10    printf ("Enter %f floats: ", SIZE);
11    for(i = 0; i <= (SIZE - 1); i = i + 1)
12        scanf ("%f", &score[i]);
13    printf ("The scores in reverse order are: \n");
14    for(i = SIZE - 1; i >= 0; i = i - 1){
15        printf ("%f", score[i]);
16        printf ("\n");}
17    getch();
18    return 0;
19 }

```

2. Write and run the following Program 12.3, then answer the following questions.

[Tulis dan laksanakan Program 12.3, kemudian jawab soalan berikutnya.]

```

1 //Program 12.3
2 #include <stdio.h>
3 #include <conio.h>
4
5 int main()
6 {
7     char a[11] = "Boring";
8     printf ("Index 0 has %c\n", a[0]);
9     printf ("Index 1 has %c\n", a[1]);
10    printf ("Index 2 has %c\n", a[2]);
11    printf ("Index 3 has %c\n", a[3]);
12    printf ("Index 4 has %c\n", a[4]);
13    printf ("Index 5 has %c\n", a[5]);
14    printf ("Index 6 has %c\n", a[6]);
15    printf ("Index 7 has %c\n", a[7]);
16    printf ("Numerically, the a[6] is %c ",a[6]);
17    getch();
18    return 0;
19 }

```

- i. How many indexes do array named, **a**, have?
[Berapakah bilangan indeks yang terdapat pada tatasusunan bernama a?]

- ii. How many characters (including the null character at the end of the array), do this array hold?
[Berapakah bilangan bilangan aksara (termasuk aksara null yang terdapat di akhir tatasusunan) yang dipegang oleh tatasusunan?]
 - iii. The number of an index is also called an index. What is the lowest index?
[Bilangan indeks turut dikenali sebagai indeks. Apakah nilai bagi indeks yang terendah?]
 - iv. What is the highest index for this array?
[Apakah indeks yang tertinggi bagi tatasusunan ini?]
 - v. What is the character value that is stored in the index number 6?
[Apakah nilai aksara yang terdapat pada indeks ke-6?]
3. Run the following Program 12.4 with the first **printf** commented out. Then answer the following questions.
[Laksanakan Program 12.4 berikut dengan pernyataan pertama **printf** dikomenkan. Seterusnya jawab soalan-soalan berikut.]

```

1 //Program 12.4
2 #include <stdio.h>
3 #include <conio.h>
4
5 int main()
6 {
7     char a[16] = "Programming C";
8     int i;
9     for(i = 0; i <= 9; i = i + 1)
10    {
11        a[i] = a[9 - i];
12        // printf ("i = %d%s\n", i, a);
13    }
14    printf ("%s", a);
15    getch();
16    return 0;
17 }

```

- i. When **i** was 0, which index of **a** was changed? To which value?
[Apabila **i** ialah 0, indeks tatasusunan **a** manakah yang berubah? Perubahan kepada nilai apa?]
- ii. When **i** was 2, which index of **a** was changed? To which value?
[Apabila **i** ialah 2, indeks tatasusunan **a** manakah yang berubah? Perubahan kepada nilai apa?]
- iii. When **i** was 9, which index of **a** was changed? To which value?

[Apabila i ialah 9, indeks tatasusunan a manakah yang berubah? Perubahan kepada nilai apa?]

- iv. After removing the comment at line 12, show the output of the program.
[Selepas menghapuskan symbol komen pada baris 12, tunjukkan output bagi aturcara.]

4. Run the following Program 12.5 and answer the following questions.
[Laksanakan Program 12.5 berikut dan jawab soalan-soalan berikut.]

```

1 //Program 12.5
2 #include <stdio.h>
3 #include <conio.h>
4
5 int main()
6 {
7     int i, x[6], y[6] = {3, 8, 2, 9, 4, 1};
8     for(i = 0; i <= 5; i = i + 1)
9         x[i] = y[i];
10    for(i = 0; i <= 5; i = i + 1)
11        printf ("%d", x[i]);
12    getch();
13    return 0;
14 }
```

- i. What are the names of the arrays?
[Apakah nama-nama tatasusunan tersebut?]
- ii. Which array is initialized during its declaration?
[Tatasusunan manakah yang dinilaiawalkan semasa penakrifan?]
- iii. Which array is assigned values in a loop?
[Tatasusunan manakah yang mengumpukkan nilai di dalam gelung?]
- iv. Determine the output if statement in line 9 is changed with $x[i] = y[i] + 1$.
[Tentukan output jika pernyataan pada baris 9 ditukarkan kepada $x[i] = y[i] + 1$.]
- v. Determine the output if statement in line 9 is changed with $x[i] = y[5 - i]$.
[Tentukan output jika pernyataan pada baris 9 ditukarkan kepada $x[i] = y[5 - i]$.]

5. Run Program 12.6 below with the statement at line 9 being replaced by suggested statement below. Show the output of this program.
 [Laksanakan Program 12.6 dengan pernyataan pada baris 9 digantikan dengan pernyataan-pernyataan yang dicadangkan berikut. Tunjukkan output bagi aturcara ini.]

```

1 //Program 12.6
2 #include <stdio.h>
3 #include <conio.h>
4
5 int main()
6 {
7     int i, a[] = {40, 20, 70, 10, 80, 30, 90};
8     for(i = 1; i <= 5; i = i + 1)
9         //_____
10        printf ("\n ");
11    getch();
12    return 0;
13 }
```

- i. `printf ("%d\t", a[i]);`
- ii. `printf ("%d\t", a[i+1]);`
- iii. `printf ("%d\t", a[i-1]);`
- iv. `printf ("%d\t", a[i] + a[i+1]);`
- v. `printf ("%d\t", a[i] + i);`

EXERCISE 2:

[LATIHAN 2]

1. The following program segment in Program 12.7 is an implementation of a sorting algorithm that sorts a list of numbers in an array. Understand the program and assume that a call to the function `sort` is as follows.
 [Segmen aturcara pada Program 12.7 berikut merupakan implementasi bagi algoritma isihan yang mengisih satu senarai nombor dalam tatasusunan. Fahamkan aturcara tersebut dan andaikan panggilan kepada fungsi `sort` adalah seperti di bawah.]

```

int a[7]={8,6,5,3,9,2,7};
sort(a,6);
```

```

1 //Program 12.7
2 void swap_values(int &v1, int &v2)
3 {
4     int temp;
5     temp = v1;
6     v1 = v2;
7     v2 = temp;
8 }
9
10 int index_of_smallest(const int a[], int begin_index,
    int end_index)
```

```

11 {
12     int min = a[end_index],
13         index_of_min = end_index;
14
15     for (int index=begin_index; index < end_index;
16         index++)
17         if (a[index] < min )
18         {
19             min = a[index];
20             index_of_min = index;
21         }
22     return index_of_min;
23 }
24
25 void sort(int a[], int last_index)
26 {
27     int index_of_next_smallest;
28
29     for (int index=last_index; index>=0; index--)
30     {
31         index_of_next_smallest = index_of_smallest(a, 0,
32         index);
33         swap_values(a[index], a[index_of_next_smallest]);
34     }
35 }

```

- i. Write the contents of the array **a** for each loop when the function **sort** is being executed. (Write your answers in a table as shown below).
 [Tuliskan kandungan tatasusunan **a** bagi setiap ulangan semasa fungsi **sort** dilaksanakan. (Tuliskan jawapan anda menggunakan jadual di bawah).]

a[0]	a[1]	a[2]	a[3]	a[4]	a[5]	a[6]
8	6	5	3	9	2	7

2. Write C statements to perform each of the following:
 [Tulis pernyataan C untuk melaksanakan setiap satu yang berikut:]
- i. Declare an array of variable **number** to allocate 10 elements.
 [Isytihar satu tatasusun bagi pembolehubah **number** untuk menyediakan 10 elemen.]

- ii. Read 10 data then assign the values into array **number**.
[Membaca 10 data untuk dimasukkan ke dalam tatasusun bagi **number**.]
- iii. Assign value of **number[3]** into **number[4]** and **number[4]** takes value of **number[5]**.
[Mengumpulkan nilai **number[3]** kepada **number[4]** dan **number[4]** mengambil nilai **number[5]**.]

EXERCISE 3:

[LATIHAN 3]

1. Given the formula for converting Fahrenheit (F) to Celcius (C):
[Diberi rumus penukaran dari unit Farenheit (F) kepada unit Selsius (C):]

$$C = 5/9 \times (F - 32)$$

Where **C** is the unit of temperature in Celcius and **F** is the unit of temperature in Farenheit

[Dimana **C** ialah unit suhu di dalam Celcius dan **F** ialah uni suhu di dalam Farenheit.]

Write a complete C program that reads in a list of data **F** from a text file, then calculates the values of **C** using the formula given. The program should use an array to store the values of **F** as example shown in Figure 12.1

[Tuliskan satu aturcara C lengkap yang membaca satu senarai data **F** daripada sebuah fail teks, kemudian mengira nilai-nilai **C** menggunakan formula yang diberi. Aturcara tersebut perlu menggunakan tatasusunan untuk menyimpan nilai-nilai **F** seperti contoh pada Rajah 12.1.]

```

13.29
29.76
14.81
23.78
29.37
..
..
..

```

Figure 12.1: input file of data **F**
[Rajah 12.1: fail input bagi data **F**]

```

Average of the temperature : 32.3
Number of high temperature : 2
Number of medium temperature : 20
Number of low temperature : 8

```

Figure 12.2: Example of output skrin
[Rajah 12.2: example of screen output]

The program then prints a **summary output** onto the screen and the **detail output** into a text file as shown in Figures 12.2 and 12.3. Grades 'H' mean high temperature; 'M' is medium temperature and 'L' is low temperature [Seterusnya, aturcara tadi akan mencetak **output rumusan** ke skrin dan **output terperinci** ke dalam sebuah fail teks seperti yang ditunjukkan masing-masing pada Rajah 12.2 dan 12.3. Gred 'H' mewakili suhu tinggi; 'M' pula mewakili suhu sederhana dan 'L' mewakili suhu rendah.]

C(Celcius)	F (Fahrenheit)	Description
=====	=====	=====
54.94	130.89	H
19.86	67.75	L
93.70	200.67	H
13.77	56.78	L
:	:	:
:	:	:
:	:	:

Figure 12.3: example of output file
[Rajah 12.3: contoh fail output]

Your program must define several functions at least as listed in the Table 12.1. You are also required to apply the concept of parameter passing to these functions.

[Aturcara anda perlu mendefinisikan beberapa fungsi sekurang-kurangnya seperti yang disenaraikan dalam jadual berikut. Anda juga dikehendaki melaksanakan konsep penghantaran parameter terhadap fungsi-fungsi tersebut.]

Table 12.1

Fungsi [<i>Function</i>]	Keterangan [<i>Description</i>]
readFile	<p>This function reads in a list of numbers from a text file and stores them into a one-dimensional array. It receives the following parameters:</p> <p>[Fungsi ini digunakan untuk membaca senarai nombor daripada sebuah fail teks dan menyimpan nombor-nombor tersebut ke dalam sebuah tatasusunan. Ia perlu menerima parameter-parameter berikut:]</p> <ul style="list-style-type: none"> • The name of the text file to be read from [Nama bagi fail teks yang hendak dibaca] • An array to store the list of numbers read [Sebuah tatasusunan untuk menyimpan senarai nombor tersebut] • A variable to store the number of data read [Pembolehubah untuk menyimpan bilangan data yang telah dibaca]
computeC	<p>This function computes the values of C. It receives the following parameters:</p> <p>[Fungsi ini digunakan untuk mengira nilai-nilai C. Ia perlu menerima parameter-parameter berikut:]</p> <ul style="list-style-type: none"> • An array that contains data F [Tatasusunan yang mengandungi senarai data F] • An array to store the calculated values of C [Tatasusunan yang akan menyimpan senarai nilai C yang telah dikira] • The number of data [Bilangan data]
average	<p>This function computes the average of a list of numbers stored in an array.</p> <p>[Fungsi ini digunakan untuk mengira purata bagi senarai nombor dalam sebuah tatasusunan.]</p>
grade	This function determines either temperature (C) is high or

	<p>medium or low. This function will return 'H' if $C \geq 35$, 'M' if $C < 35$ and $C \geq 20$ and 'L' if $C < 20$.</p> <p>[Fungsi ini digunakan untuk menentukan sama ada suhu itu tinggi, sederhana atau rendah dalam pengujian kategori suhu tersebut. Fungsi ini memulangkan nilai 'H' sekiranya $C \geq 35$, 'M' jika $C < 35$ atau $C \geq 20$ dan 'L' jika $C < 20$]</p>
writeFile	<p>This function prints the output file as in Figure 12.3. It receives the following parameters:</p> <p>[Fungsi ini digunakan untuk mencetak fail output seperti dalam Rajah 12.3. Ia perlu menerima parameter-parameter seperti di bawah:]</p> <ul style="list-style-type: none"> • An array that contains data F [Tatasusunan yang mengandungi senarai data F] • An array that contains data C [Tatasusunan yang mengandungi senarai data C] • The number of data [Bilangan data]

For printing summary output onto the screen, you may define another function or you may just put the code into the function **main**.

[Untuk cetakan **output rumusan** ke skrin, anda juga boleh mendefinisikan satu fungsi yang lain atau anda boleh menulis kod-kod yang berkenaan terus ke dalam fungsi **main**.]