To:
Tenouk

C LAB WORKSHEET 9_1
C & C++ Array Data Type Part 2

1. More questions and answers.
2. Array and loops
3. Tutorial references that should be used together with this worksheet are array part 1 and array part 2.

- First run the following program with the first printf() commented out then answer the questions. Finally check your answers by removing the // (uncomment).

```
#include <stdio.h>

void main()
{
    char a[11] = "sweet girl";
    int i;

    for(i = 0; i <= 9; i = i + 1)
    {
        a[i + 1] = a[i];
        // printf("i = %d %s\n", i, a);
    }
    printf("%s\n", a);
}
```

a. When i was 0, which index of a was changed? To which value?
b. When i was 1, which index of a was changed? To which value?
c. When i was 2, which index of a was changed? To which value?
d. When i was 9, which index of a was changed? To which value?

a. Index 1 was changed to 0. a[1]=a[0]. Assign a[0] to a[1].
b. Index 2 was changed to 1. a[2]=a[1]. Assign a[1] to a[2].
c. Index 3 was changed to 2. a[3]=a[2]. Assign a[2] to a[3].
d. Index 10 was changed to 9. a[10]=a[9]. Assign a[9] to a[10].

When the first printf() uncommented the following is the output.

- First run the following program with the first printf() commented out then answer the questions. Finally check your answers by removing the // (uncomment).

```
#include <stdio.h>

void main()
{
    char a[11] = "sweet girl";
    int i;

    for(i = 0; i <= 9; i = i + 1)
    {
        a[i] = a[9 - i];
        // printf("i = %d %s\n", i, a);
    }
    printf("%s\n", a);
}
```

a. When i was 0, which index of a was changed? To which value?
b. When i was 1, which index of a was changed? To which value?
c. When i was 2, which index of a was changed? To which value?
d. When i was 9, which index of a was changed? To which value?

a. Index 0 was changed to 9. a[0]=a[9], assign a[9] to a[0].
b. Index 1 was changed to 8. a[1]=a[8], assign a[8] to a[1].
c. Index 2 was changed to 7. a[2]=a[7], assign a[7] to a[2].
d. Index 9 was changed to 0. a[9]=a[0], assign a[0] to a[9].

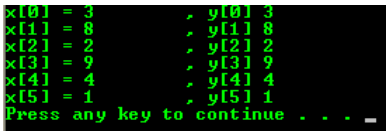
When the first printf() uncommented the following is the output.

- So far we have been looking only at character arrays that terminate with a null character or simply strings. Now let us turn our attention to numeric arrays. Here, the first loop completes before the second one starts.

```
#include <stdio.h>

void main()
{
    int i, x[6], y[6] = {3, 8, 2, 9, 4, 1};
    for(i = 0; i <= 5; i = i + 1)
        x[i] = y[i];
    for(i = 0; i <= 5; i = i + 1)
        printf("x[%d] = %d\t, y[%d] %d\n", i, x[i], i, y[i]);
}
```

a. What are the names of the arrays?
b. What are the names of the scalars? Scalars are the variables that are not arrays.
c. How many indexes do each of the arrays have?
d. What is the lowest index and the highest index for the arrays?
e. Which array is initialized during its declaration?
f. Which array is assigned values in a loop?
g. Are the arrays printed horizontally or vertically?
h. How would you write the code if you had wanted to write the arrays the opposite way?



a. x and y.
b. i.
c. 6.
d. The lowest index is 0 and the highest is 5.
e. Array y.
f. Array x.
g. Vertically.
h. Change the for loop components to the following for both arrays.

```
for(i = 5; i >= 0; i = i - 1)
```

```
#include <stdio.h>

void main()
{
    int i, x[6], y[6] = {3, 8, 2, 9, 4, 1};
    for(i = 5; i >= 0; i = i - 1)
        x[i] = y[i];
    for(i = 5; i >= 0; i = i - 1)
        printf("x[%d] = %d,\t y[%d] %d\n", i, x[i], i, y[i]);
}
```

- Run the following program and answer the questions. It is the previous program with some code has been changed.

```
#include <stdio.h>

void main()
{
    int i, x[6], y[7] = {3, 8, 2, 9, 4, 1, 0};
    for(i = 0; i <= 5; i = i + 1)
        x[i] = y[i + 1];  // Statement 1
    for(i = 0; i <= 5; i = i + 1)
        printf("x[%d] = %d\t y[%d] = %d\n", i, x[i], i, y[i]);
}
```

  a. When i is 0, which index of y is used? Which index of x is changed?
  b. When i is 1, which index of y is used? Which index of x is changed?
  c. When i is 5, which index of y is used? Which index of x is changed?
  d. Can you determine what would be printed if Statement 1 were changed to: x[i] = y[i] + 1;?
  e. Can you determine what would be printed if Statement 1 were changed to: x[i] = y[6 -i];?
  f. Can you determine what would be printed if Statement 1 were changed to: x[i] = y[1];?

```
x[5] = 1,       y[5] 1
x[4] = 4,       y[4] 4
x[3] = 9,       y[3] 9
x[2] = 2,       y[2] 2
x[1] = 8,       y[1] 8
x[0] = 3,       y[0] 3
Press any key to continue . . .
```

```
x[0] = 8        y[0] = 3
x[1] = 2        y[1] = 8
x[2] = 9        y[2] = 2
x[3] = 4        y[3] = 9
x[4] = 1        y[4] = 4
x[5] = 0        y[5] = 1
Press any key to continue . . .
```

  a. Index 1 of y was used. No change on index x.
  b. Index 2 of y was used. No change on index x.
  c. Index 6 of y was used. No change on index x.
  d.

```
x[0] = 4        y[0] = 3
x[1] = 9        y[1] = 8
x[2] = 3        y[2] = 2
x[3] = 10       y[3] = 9
x[4] = 5        y[4] = 4
x[5] = 2        y[5] = 1
Press any key to continue . . .
```

  e.

```
x[0] = 0        y[0] = 3
x[1] = 1        y[1] = 8
x[2] = 4        y[2] = 2
x[3] = 9        y[3] = 9
x[4] = 2        y[4] = 4
x[5] = 8        y[5] = 1
Press any key to continue . . .
```

  f.

```
x[0] = 8        y[0] = 3
x[1] = 8        y[1] = 8
x[2] = 8        y[2] = 2
x[3] = 8        y[3] = 9
x[4] = 8        y[4] = 4
x[5] = 8        y[5] = 1
Press any key to continue . . .
```

- When running the following program, enter the following input: 6, 4.0, 5.0, 2.0, 55.0, 8.0, 1.0. The first number is the count and the rest are the items that go in the array. When scanning a float, use %f instead of %.2f.

```
#include <stdio.h>

void main()
{
    int count, i;
    float a[6];
    printf("How many numbers do you have?\n");
    scanf_s("%d", &count);
    printf("Enter the number(s): \n");
    for(i = 0; i < count; i = i + 1)
        scanf_s("%f", &a[i]);
    for(i = 0; i < count; i = i + 1)
        printf("a[%d] = %.2f\n", i, a[i]);
}
```

  a. What was the value of count?
  b. In the scanf_s() and the printf(), did i ever reach the value of count? Why or why not?
  c. The 4.0 was read into which index of the array?
  d. The 5.0 was read into which index of the array?

```
How many numbers do you have?
6
Enter the number(s):
4.0 5.0 2.0 55.0 8.0 1.0
a[0] = 4.00
a[1] = 5.00
a[2] = 2.00
a[3] = 55.00
a[4] = 8.00
a[5] = 1.00
Press any key to continue . . .
```

  a. 6.
  b. i did not reach the value of count because the terminal condition is i < count and an array index start from 0 instead of 1.
  c. Index 0.
  d. Index 1.

- Enter the following input: 4.0, 5.0, 2.0, 55.0 for the below program example.

```
#include <stdio.h>

void main()
{
    int i, j;
    float a[30];
    printf("Enter floats that are < 10, 10 and above to stop:\n");
    scanf_s("%f", &a[0]);
    for(i = 0; a[i] <= 10.0; i = i + 1)
       scanf_s("%f", &a[i + 1]);
    // printf("Last value of i was %d\n", i);
    for(j = 0; j < i; j = j + 1)
       printf("a[%d] = %.2f\n", j, a[j]);
}
```

  a. Does the scanf_s() here change the value of i?
  b. The first data item was read into which index of the array?
  c. The first time the condition a[i] <= 10.0 was checked, i was 0. Was the condition true or false? During the first time through the loop, into which index did the scanf_s() place a data item?
  d. The second time that the condition a[i] <= 10.0 was checked, i was 1? Was the condition true or false? Into which index did the next scanf_s() place a new data item during the second iteration of the

---

```
Enter floats that are < 10, 10 and above to stop:
4.0 5.0 2.0 55.0 10
a[0] = 4.00
a[1] = 5.00
a[2] = 2.00
Press any key to continue . . .
```

  a. No.
  b. Index 0.
  c. True. Index 0.
  d. True. Index 2.
  e. i was 3.

loop?

e. When the condition a[i] <= 10.0 finally became false, what was the value of i? Run the program again by removing the // (uncomment) to check your answer.

f. In the second loop does j ever reach this last value of i in the printf ()?

```
Enter floats that are < 10, 10 and above to stop:
4.0 5.0 2.0 55.0
Last value of i was 3
 a[0] = 4.00
 a[1] = 5.00
 a[2] = 2.00
Press any key to continue . . .
```

f. No, at this stage i = 3 and j = 3.

More Question and Activities

1. Show the output from each code using the given array and for loop. Remember that the first index is always 0 and not 1. Use the same for loop with each printf (). Study the source code and the output.

```
#include <stdio.h>

void main()
{
    int i, a[ ] = {40, 20, 70, 10, 80, 30, 90};
    for(i = 1; i <= 5; i = i + 1)
        ----------------------------
     printf("\n");
}
```

Example:

```
#include <stdio.h>

void main()
{
    int i, a[ ] = {40, 20, 70, 10, 80, 30, 90};
    for(i = 1; i <= 5; i = i + 1)
        printf("%d\t",  a[i]);
    printf("\n");
}
```

a. printf("%d\t", i);
b. printf("%d\t", a[i + 1]);
c. printf("%d\t", a[i -1]);
d. printf("%d\t", a[i] + a[i + 1]);
e. printf("%d\t", a[i] + 1);
f. printf("%d\t", a[i] + i);

---

a.
```
C:\WINDOWS\system32\cmd.exe
20      70      10      80      30
Press any key to continue . . .
```

b.
```
70      10      80      30      90
Press any key to continue . . .
```

c.
```
40      20      70      10      80
Press any key to continue . . .
```

d.
```
90      80      90      110     120
Press any key to continue . . .
```

e.
```
21      71      11      81      31
Press any key to continue . . .
```

f.
```
21      72      13      84      35
Press any key to continue . . .
```

```
C:\WINDOWS\system32\cmd.exe
a[1] = 0 a[2] = 0 a[3] = 0 a[4] = 0 a[5] = 0
Press any key to continue . . .
```

2. Show the contents of the new array after it is changed by each assignment statement and complete the code. Use the same for loop with each assignment statement. Start with the given array for each. When doing these, remember that during each iteration of the loop, array elements are changing. This is especially important in question 2.d, 2.f and 2.h. Study the source code and the output.

```
#include <stdio.h>

void main()
{
    int i, a[ ] = {40, 20, 70, 10, 80, 30, 90};
    for(i = 1; i <= 5; i = i + 1)
    {
        ---------------------------------------
        ---------------------------------------
    }
}
```

Example:

```
#include <stdio.h>

void main()
{
    int i, a[ ] = {40, 20, 70, 10, 80, 30, 90};
    for(i = 1; i <= 5; i = i + 1)
    {
        a[i] = 0;
        printf("a[%d] = %d ", i, a[i]);
    }
    printf("\n");
}
```

a. a[i] = i;
b. a[i] = a[i + 1];
c. a[i + 1] = a[i];
d. a[i] = a[i] + a[i] + a[i + 1];
e. a[i + 1] = a[i] + a[i + 1];

a.
```
a[1] = 1 a[2] = 2 a[3] = 3 a[4] = 4 a[5] = 5
Press any key to continue . . .
```

b.
```
a[1]=70, a[i + 1]=70
a[2]=10, a[i + 1]=10
a[3]=80, a[i + 1]=80
a[4]=30, a[i + 1]=30
a[5]=90, a[i + 1]=90

Press any key to continue . . .
```

c.
```
i=1, a[1 + 1]=20, a[i]=20
i=2, a[2 + 1]=20, a[i]=20
i=3, a[3 + 1]=20, a[i]=20
i=4, a[4 + 1]=20, a[i]=20
i=5, a[5 + 1]=20, a[i]=20

Press any key to continue . . .
```

d.
```
i=1, a[i]=110, a[i] + a[i] + a[i + 1]=290
i=2, a[i]=150, a[i] + a[i] + a[i + 1]=310
i=3, a[i]=100, a[i] + a[i] + a[i + 1]=280
i=4, a[i]=190, a[i] + a[i] + a[i + 1]=410
i=5, a[i]=150, a[i] + a[i] + a[i + 1]=390

Press any key to continue . . .
```

e.
```
i=1, a[i + 1]=90, a[i] + a[i + 1]=110
i=2, a[i + 1]=100, a[i] + a[i + 1]=190
i=3, a[i + 1]=180, a[i] + a[i + 1]=280
i=4, a[i + 1]=210, a[i] + a[i + 1]=390
i=5, a[i + 1]=300, a[i] + a[i + 1]=510

Press any key to continue . . .
```

f. a[6 – i] = a[5 – i];
g. a[5 – i] = a[6 – i];

f.
```
i=1, a[6 - i]=80, a[5 - i]=80
i=2, a[6 - i]=10, a[5 - i]=10
i=3, a[6 - i]=70, a[5 - i]=70
i=4, a[6 - i]=20, a[5 - i]=20
i=5, a[6 - i]=40, a[5 - i]=40

Press any key to continue . . .
```

g.
```
i=1, a[5 - i]=30, a[6 - i]=30
i=2, a[5 - i]=30, a[6 - i]=30
i=3, a[5 - i]=30, a[6 - i]=30
i=4, a[5 - i]=30, a[6 - i]=30
i=5, a[5 - i]=30, a[6 - i]=30

Press any key to continue . . .
```