Introduction to C Programming

— Structures —

Waseda University

Today's Topics

• How to learn Structure

- What is Structure
- Usage of Structures
- How to initialize and access members in a Structure variable.

What's difference between Array and Structure?

[Array]

- An array is a set of variables with the same name.
- All elements in an array have the same data type.



[Structure]

• Structure is user defined data type which is used to store heterogeneous data under unique name.



Usage of Structures (Defining a Structure)

- A structure can be defined as a new data type.
- To define a structure, you must use the struct statement.

The general form of a structure definition

- *structure tag* is name of the structure.
- The variable named in a structure are called *member*.

Usage of Structures (Defining a Structure)

- A structure can be defined as a new data type.
- To define a structure, you must use the struct statement.

Example

struct	studer	nt	{
	char		name[10];
	int	s	core;
	doubl	e	devi;
};			

- *structure tag* is name of the structure.
- The variable named in a structure are called *member*.

Usage of Structures (Declaring Structure Variables)

• You need to declare variables with new data type defined by **struct** statement.

```
struct student{
             name[10];
        char
        int
             score;
        double devi;
};
int main(void){
    struct student a, b;
    •
}
```

• Each structure variables **a** and **b** with **student** data type have three members.

Accessing Members of Structure

• Structure members are accessed using dot(.) operator.

```
struct student{
       char name[10];
       int score;
       double devi;
};
int main(void){
    struct student a, b;
    strcpy(a.name,"Aki"); //Return a copy of the string.
    a.score = 90:
   a.devi = 64.2:
}
```

Initializing Members of Structure

• You can initialize some or all members in a structure as following:

```
struct student{
       char name[10];
        int score;
       double devi;
};
int main(void){
    struct student a ={"Aki", 90, 64.2};
    :
}
```

Example 1 Structure assignment

- You can copy a structure variable using assignment operator (=).
- You have to write the same data type with left and right hand side variables.
- Write a following program "struct1.c".

```
#include<stdio.h>
#include<string.h>
struct student{
       char name[10]:
       int score;
       double devi:
}:
int main(void){
    struct student a, b;
    strcpy(a.name,"Aki");
    a.score
               = 90:
    a.devi = 64.2:
    printf("Name: %s¥n", a.name);
    printf("Score: %3d¥n", a.score);
    printf("Devi: %.1f\n", a.devi);
   : (Continue to the right)
```

```
:
b=a; /*Substituting b for a*/
printf("Name: %s¥n", b.name);
printf("Score: %3d¥n", b.score);
printf("Devi: %.1f¥n", b.devi);
```

```
return 0;
```

}

```
[Results]
```

Name:	Aki
Score:	90
Devi:	64.2
Name:	Aki
Score:	90
Devi:	64.2

Example 2

- The structure variable can be used as a return value of a function.
- Let's copy "struct1.c" to "struct2.c".
- Write the following program and run it.



Exercise

Exercise 1 : Write the following program and run it (TriArea.c)

- Let O be the origin.
- Let A_1 and A_2 be points of Cartesian coordinate system.
- Write the program that calculate a measure of triangle OA_1A_2 .
- Here, x and y coordinates of each point A1 and A2 are input number.
- x and y coordinates are members in a structure.
- You may not use the function (You will have to use the function in Exercise 2)
- You may use the following formula to calculate a measure of triangle: For points $A_1(x_1, y_1), A_2(x_2, y_2)$, the measure S of triangle OA_1A_2 is

$$S = \frac{1}{2}|x_1y_2 - x_2y_1|.$$

• Example:

```
Input x1 = 1 [Enter]

y1 = 1.5 [Enter]

Input x2 = 3.5 [Enter]

y2 = 1.5 [Enter]

Measure of triangle is 1.875
```

Exercise Hints

- You can use the function fabs() which returns the absolute value.
- The required header for the fabs function is **math.h**.
- To use math.h, you have to add the "-Im" option for gcc.
- For example, you may define a structure as follows :

```
struct point{
    double x;
    double y;
};
```

Exercise

Exercise 2 : Let's copy "TriArea.c" to "TriArea1.c" and edit it

- Write the function "Measure" which calculate the area of a triangulation.
- Arguments of the function "Measure" are two structure variables which have points.
- A return value of the function "Measure" is area of an triangle.

Summary

• How to learn Structure

- What is Structure
- Usage of Structures
- How to initialize and access members in a Structure variable.