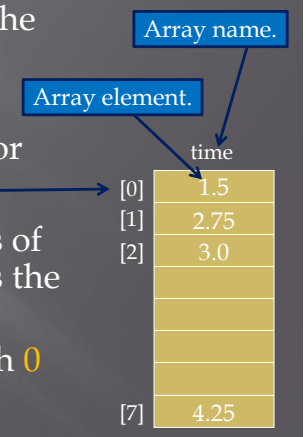
C PROGRAMMING FOR ENGINEERS

One-Dimensional Arrays

Facts

- An array is an indexed data structure.
- All values stored in an array are of the same data type.
- An element of an array is accessed using the array name and an index or subscript.
- The name of the array is the address of the first element and the subscript is the offset.
- In C, the subscripts always start with 0 and increments by 1.



Definition and Initialization

- An array is defined using a declaration statement.

  General form: data type array_name[size];
  - allocates memory for size elements.
  - subscript of first element is 0.
  - subscript of last element is size-1.
  - size must be an integer constant.

Example

Notice that square brackets are required.

int temp[10];

- allocates memory for 10 integer values
- subscript of first element is 0
- subscript of last element is 9

The array temp can be assigned values by assignment operations.
Examples:

```
temp[0] = 32;
temp[1] = 35.5;
temp[9] = 150;
```

What happens with this assignment operation?

Initializing Arrays

- Arrays can be initialized at the time they are declared.

Examples:

```
double taxrate[3] ={0.15, 0.25, 0.3};
char list[5] = {'h', 'e', 'l', 'l', 'o'};
double vector[100] = {0.0};
int stats[ ] = {5,0,-5};
```

NOTE: It is recommended to not use these techniques for data sets larger than 5 values.

Assigns zero to all 100 memory positions of the array.

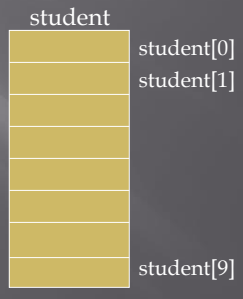
The size of the array is set to 3 when the initialization is performed.

Assigning values to an array

- **for** loops are most often used to assign values to an array.

Example:

```
int student[10], i;
for(i=0; i<=9; i++)
{
    student[i] = i+1;
}
```



Input from a data file

- Arrays are often used to store information from an input data file.

Example:

```
double time[10], motion[10];
int i, ndata;
FILE *quake;
quake = fopen(inputfile, "r");
fscanf(quake,"%i",&ndata);
for(i=0; i<=ndata-1; i++)
{
    fscanf(quake, "%lf  %lf",&time[i], &motion[i]);
}
```