Arrays

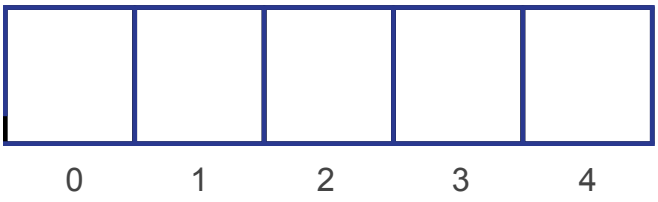
Array, A ray, Air ray, Hairy, Hooray!

Arrays

- An array is a data structure that contains a number of data values all of which have the same type. (e.g. all *int*)
- These values are also known as elements.

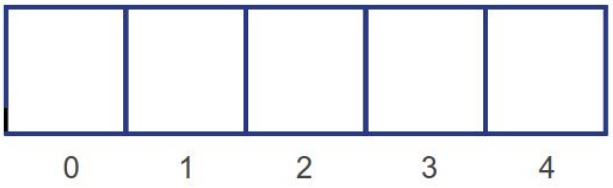
One-Dimensional Arrays

- The elements in the array are arranged one after another in a single row or column, just like boxes lined up, or a set of lockers in the hall.
- Each box will hold only one piece of data.



One-Dimensional Arrays:

- With the array, notice that the indexes start at 0.
- The upper limit is the number of elements minus 1.
- cars[5] would have subscripts 0, 1, 2, 3, 4
  the lower limit is 0 and the upper limit is 4



Declaring an array

- First we must indicate what type of data we are going to put into the array.
- Second we must name the array.
- Third we must decide how many elements are in the array.

```
int point[50];
```

  - Note: an array of size n, is indexed from 0 to n-1 the above array has 50 elements with indexes: 0, 1, 2, …49

Declaring an Array

- `int class[5];`
- `float sales[20];`
- The square brackets let the program know how big the array will be.
- Once declared, the size of the array can't be changed.

Initializing Arrays using For Loop

- A useful way of initializing (populating) an array is to use a for loop.

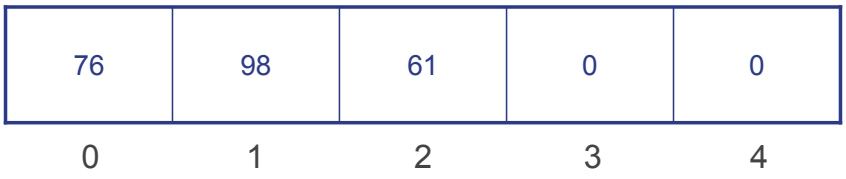
```
int class[5];

for( i=0; i<5; i++) {
   class[i] = -1;
}
```

- This is the most usual and straightforward way to initialize an array.

Initializing Arrays

- There are other ways of initializing arrays. This is the easiest way, but less useful.

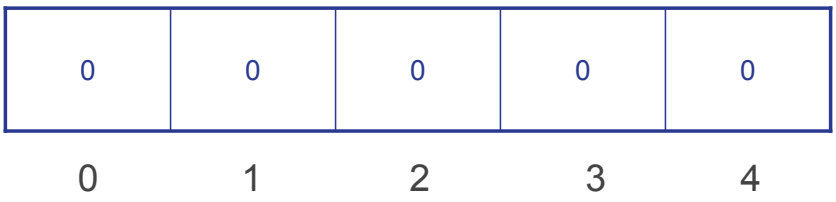
```
int class[5] = {76, 98, 61};
```



Notice elements 3 and 4 have the value of zero. If you do not put enough values in the rest will fill in with 0.

Initializing to all Zeros

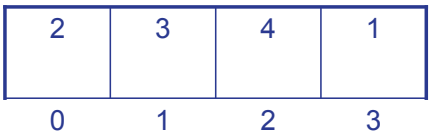
```
int class[5]= {0};
```



Initializing Arrays

- If you do not enter a size of array, but you do enter elements into your array in the declaration of the array then the size of your array will be equal to the number of elements in the array.

- `int array[] ={2, 3, 4, 1};`
- Arrays size is 4



Examples

```
int combination[] = { 36, 24, 12};

printf(The combination is:\n”);
printf(Turn left to %d\n”, combination[0]);
printf(Turn right to %d\n”, combination[1]);
printf(Turn left to %d, open\n”, combination[2]);
```

Print out an array using loop

```
char grades[] = { ‘A’, ‘B’, ‘C’, ‘B’, ‘F’,
                  ‘A’, ‘B’, ‘C’, ‘B’, ‘A’,
                  ‘A’, ‘A’, ‘B’, ‘B’, ‘C’,
                  ‘A’, ‘A’, ‘B’, ‘B’, ‘C’,
                  ‘A’, ‘A’, ‘B’, ‘B’, 'A’ };

int student;
for (student = 0; student < 25; student++)
   printf("Student %d, %c\n", student+1,grades[student]);
```