1) How to access the elements in arrays?

Method 1: use indices

```
// Delare an int array with 3 elements
// and initialize it to {1, 2, 3}
int someArray[3] = {1, 2, 3};

// Access the first element in someArray
int temp = someArray[0];
// Will print 1
cout<< temp << endl;

// Access the second element in someArray
temp = someArray[1];
// Will print 2
cout<< temp << endl;
```

Method 2: use pointers

```
// Delare an int array with 3 elements
// and initialize it to {1, 2, 3}
int someArray[3] = {1, 2, 3};

// Get the address of the first element
int* temp = &someArray[0];
// Will print 1
cout<< *temp << endl;

// Get the address of the first element
temp = someArray;
// Will print 1
cout<< *temp << endl;
// Will print 2
cout<< *(temp+1) << endl;
```

2) How to specify array parameters in function declarations?

Method 1: specify array parameters as arrays

```
// Specify someArray as an int array
void someFunction(int someArray[]){
    ...
}
```

Method 2: specify array parameters as pointers

```
// Specify someArray as an int pointer
void someFunction(int *someArray){
    ...
}
```

---

3) How to pass arrays to functions?

Method 1: pass arrays by their names

```
int main()
{
    int array1[3];
    // Pass array1 by its name
    someFunction(array1);
    ...
}
```

Method 2: pass pointers to arrays

```
int main()
{
    int array1[3];
    // Pass a pointer to array1's first element
    someFunction(&array1[0]);
    ...
}
```