

# Arrays and function calls

## Passing arrays to function calls

- There are some limitations
  - Arrays cannot be copied, so they cannot be passed by value
  - Arrays can only be passed by reference if the size is known at compile time
- Arrays are usually passed by address
  - This takes advantage of their property to "decay" into a pointer

```
void somefunc(int *pi);           // Function that takes pointer to int
int pa[] = {4, 3, 2, 1};
somefunc(pa);                     // &pa[0] will be passed to function
```

## Arrays and function calls

- If we want to emphasize that the function is meant to take an array, we can use a different syntax

```
void somefunc(int pi[]);
```

```
// Function that takes array of int
```

```
void somefunc(int pi[4]);
```

```
// Function that takes array of int (the '4' is ignored)
```

- Inside somefunc, pi can be accessed as a pointer to the first element of an array

```
void somefunc(int *pi) {
```

```
    cout << *pi << endl;
```

```
// First element of argument array
```

```
    cout << pi[1] << endl;
```

```
// Second array element of argument array
```

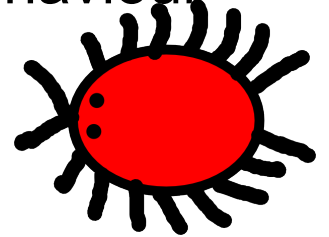
```
    cout << *(pi + 2) << endl;
```

```
// Third element of argument array
```

```
}
```

- With built-in arrays, there is no way of knowing how many elements are in the array
  - Easy to make a mistake that leads to undefined behaviour

```
void somefunc(int *pi) {  
    cout << pi[5] << endl;    // No compiler warning!  
}
```



- To be safe, we should pass in the number of elements

```
void somefunc(int *pi, int n) {  
    for (int i = 0; i < n; ++i)  
        cout << pi[i] << endl;    // Safe  
}
```

## Passing arrays by reference

- To pass an array by reference, we need to know the number of elements at compile time
  - We also need to get the syntax right

```
void func3(int& pi[4]);           // Array of 4 references to int - error!
```

```
void func3(int (&pi)[4]);        // Reference to array of 4 ints - OK
```

- Unlike passing by address, the compiler will check whether we have the right number of elements

```
int pa3[] = {5, 4, 3, 2, 1};    // Array of 5 ints
```

```
func3(pa3);                     // Compiler error
```