



COMP1511

Tutorial 5

2D arrays | array functions | strings

2D Arrays - farm.c





Passing arrays into functions

A function that takes an array:

```
void print_array(int array[SIZE]);
```

Passing an array into this function:

```
print_array(my_array);
```

Strings Recap



String Functions



```
// Returns : 1 if `c` is a lowercase letter
//           : 0 otherwise.
int is_lowercase(char c);

// Returns : 1 if `c` is an uppercase letter
//           : 0 otherwise.
int is_uppercase(char c);

// Returns : 1 if `c` is a letter
//           : 0 otherwise.
int is_letter(char c);

// Returns : `c` converted to lowercase, if
//           it was an uppercase letter
//           : `c` unmodified, otherwise
char to_lowercase(char c);

// Returns : `c` converted to uppercase, if
//           it was a lowercase letter
//           : `c` unmodified, otherwise
char to_uppercase(char c);

// Returns : 1 if `c` is an uppercase or
//           lowercase vowel
//           : 0 otherwise.
int is_vowel(char c);
```

// Functions to implement:

// 1.

// returns the number of lowercase letters
// in `char *string`

int count_lowercase(char *string);

// 2.

// modifies `char *string` by converting all
// its vowels to uppercase

void make_vowels_uppercase(char *string);

// 3..

// shortens a string so that it ends after
// the first word

// e.g. "This is a sentence" should turn into:
// "This"

//

// (hint. what defines when a string ends?)

void delete_following_words(char *string);

Side Notes





C Expressions

You can think of C expressions in a similar way to maths expressions:

```
return 'A' <= c && c <= 'Z'; // Let c = 'E'
```

```
return 'A' <= 'E' && 'E' <= 'Z';
```

```
return 1 && 1; // True is represented by 1 in C
```

```
return 1;
```