# Tute 4

## COMP1511 23T1

### Jack Robbers

# content

- functions
- scanning in loops
- arrays

# assignment 1

- cs_defence, out now
- due Monday Week 7 (27th of March)

# scanning in loops

```c
int main(void) {
    int number;
    while (scanf("%d", &number) == 1) {
        printf("%d\n", number)
    }
    return 0;
}
```

what happens when you type:

- a number?
- a letter?
- numbers with spaces between them?
- a double?
- just press enter?

# arrays

arrays store things of the same type sequentially

# arrays and functions activity

in groups, write the following functions, swap who is holding the pen after each instruction

# instructions

### Odd Only - `void odd_only(int array[SIZE])`

e.g - `odd_only({3, 2, 3, 4, 5, -9});` (SIZE is 6 in this case)

1. Create a while loop which loops through every element of the array.
2. Write an if statement which adds 1 to each even value. Do this within the while loop.

### Copy Array - `void copy_array(double from[SIZE], double to[SIZE])`

e.g - `copy_array({3.1415, 2.71828, 1.4142}, {0.0, 0.0, 0.0});` (SIZE is 3 in this case)

1. Create a while loop that loops through every element of the first array.
2. Copy the elements of the first array into the second array

### Print Array - `void print_array(int array[SIZE])`

e.g `print_array({5, 10, 15, 20})` (SIZE is 4 in this case)

1. Create a while loop that loops through each element in the array.
2. Print out each element
3. Modify your code so that the output is of the form "[5, 10, 15, 20]".

### Largest Character - `char largest_character(char array[SIZE])`

e.g - `printf("%c\n", largest_character({'C', 'O', 'M', 'P', '1', '5', '1', '1'}));` (SIZE is 8 in this case)

1. Create a character variable called `largest_character`, equal to the first character of the array.
2. Create a while loop to loop through the character array.
3. Create an if statement to check if the current character has a higher ascii value than `largest_character`
4. Return the largest character you've found.

# odd_only

```c
void odd_only(int array[SIZE]) {
    // 1. Create a while loop which loops through every element of the array.
    int i = 0;
    while (i < SIZE) {
        // 2. Write an if statement which adds 1 to each even value. Do this within the while loop.
        if (array[i] % 2 == 0) {
            array[i] = array[i] + 1;
        }
        i++;
    }
}
```

# copy_array

```
void copy_array(double from[SIZE], double to[SIZE]) {
    // 1. Create a while loop that loops through every element of the first array.
    int i = 0
    while (i < SIZE) {
        // 2. Copy the elements of the first array into the second array (leave 0's at the end)
        to[i] = from[i];
        i++;
    }
}
```

# print_array

```c
void print_array(int array[SIZE])
    int i = 0;
    // 3. open brackets
    printf("[")
    // 1. Create a while loop that loops through each element in the array.
    while (i < SIZE) {
        // 2. Print out each element
        printf("%d", array[i]);
        i++;

        // 3. Modify your code so that the output is of the form "[5, 10, 15, 20]".
        if (i != SIZE) {
            printf(", ")
        }
    }
    // 3. end the line
    printf("]\n");
```

# largest_character

```
char largest_character(char array[SIZE]) {
    // 1. Create a character variable called largest, equal to the first character of the array.
    char largest = array[0];

    // 2. Create a while loop to loop through the character array.
    int i = 0;
    while (i < SIZE) {
        // 3. Create an if statement to check if the current character
        // has a higher ascii value than "largest_character"
        if (array[i] > largest) {
            largest = array[i]
        }

        i++
    }

    // 4. Return the largest character you've found.
    return largest
}
```

# functions

live coding

- make_colour()
- get_main_colour()
- invert_colours()