



https://folk.ntnu.no/michaeng/tdt4205_21/
https://folk.ntnu.no/michaeng/tdt4186_21/
michael.engel@ntnu.no

Theoretical exercises
Spring 2021

Theoretical Exercises 1

Introduction to C programming

Please submit solutions on Blackboard by Friday, 29.01.2021 14:00h

Notice: Please submit solutions on Blackboard in groups of two or three students.

You will receive feedback on your submission, but the theoretical exercises are **not** part of your final grade.

1.1 Parameter passing

Consider the following C program:

```
1 #include <stdio.h>
2
3 int a = 23;
4 void increment_with_value (int a, int b) {
5     a += b;
6 }
7
8 int main(void) {
9     increment_with_value(a, 1);
10    return a;
11 }
```

Without compiling and running the program, indicate which value is returned by the `main` function?

Briefly explain your answer.

1.2 Symbols

If we compile the program shown above using `gcc -std=c11 -Wall -o test test.c` and execute `nm test` afterwards, the `nm` output does not contain a memory address for variable `b`.

Briefly explain why `b` is not listed.

1.3 C arrays

Consider the following C program:

```
1 #include <stdio.h>
2 #include <string.h>
3
4 int main(void) {
5     int foo = 0;
6     char s[12];
7     char *t = "01234567890123";
8
9     printf("foo %p\n s %p\n", &foo, s);
10    strcpy(s, t);
11    printf("foo = %d\n", foo);
12 }
```

- Without compiling and running the program, give the value printed for `foo`.
- Describe briefly the problem that shows up in the given code which results in this output.
- Modern C compilers protect against the problems shown in this example. For `gcc` or `clang`, find out which command line option can be used to enable this protection.
- What would the output be if line 5 was replaced by

```
static int foo = 0;
```

Briefly explain whether this change would solve the underlying problem.

1.4 Functions and variables

Consider the following C program:

```
1 #include <stdio.h>
2
3 const int c = 1;
4 int d, counter = 0;
5
6 unsigned int rec(unsigned int number) {
7     counter++;
8     return rec(counter);
9 }
10
11 int main(void) {
12     int a = rec(c);
13     printf("%d\n", a);
14     return 0;
15 }
```

- Which memory segments are the function `rec()`, variables `c`, `d`, `counter`, and `a` as well as parameter `a` located in?
- What happens if you execute the compiled program? What changes if you add a local variable `char array[1000]` to function `rec`?