

Electrical Engineering and Information Technology, B.Eng.
Introduction to the C Programming Language: Exercises

Exercise Sheet 6

1. Exercise. Try out the lines of E.5-1:

```
int  x, y;
int  z[10];
int  *ipoint; /* ipoint is a pointer to int */

x = 1;
y = 2;
ipoint = &x; /* ipoint points to x */
y = *ipoint; /* y is 1 */
*ipoint = 0; /* x is 0 */
ipoint = &z[0]; /* ipoint points to z[0] */
```

Surround the lines with a main() and add some printf()'s to verify the pointer mechanism.

2. Exercise. Define an int array numbers[10] and initialize the array with

```
numbers[i] = i;
```

Define an int pointer *finger and let finger point to numbers[0]. Now use finger and printf() to output the values of numbers[0], numbers[1] and numbers[3].

Print also the values of finger, finger+1, finger+2 with %d. What is the difference between the printed values?. How many bytes are therefor allocated for an int?

3. Exercise. Define 3 arrays a[10], b[10], c[10]. Initilize all components of the first array with 1, of the second array with 2, of the third array with 3. Then set

```
b[14] = 2;
```

Print the values of all components of a and c to the screen. What is the result of writing a 2 into b[14]?

Reset a[] and c[], define an int pointer *finger which shall point to b. Write a 5 into b[15] with the help of finger and print all values of a and c to the screen. Is there a difference to the result before?

4. Exercise. Write a simple copy program that reads with getchar() from the keyboard until '\n' and writes with putchar() to the screen substituting a sequence of blanks by a single blank.