

Informatik für Mathematiker und Physiker HS15

Exercise Sheet 8

Submission deadline: 15:15 - Tuesday 10th November, 2015

Course URL: <http://lec.inf.ethz.ch/ifmp/2015/>

Assignment 1 (4 points)

On 8th June 2012, Neue Zürcher Zeitung went completely digital, and what they did to visualize this was to encode the whole cover page in binary (see image to the right) in the way that each 8-bit binary number represented a single ASCII character (e.g. 01001110 01011010 01011010 encodes NZZ, since for example 01001110 is 78 which is the ASCII code for N). The nzz.in is a plain text input file that contains the transcript of the NZZ binary cover page. Your task is to write a program `nzz_decoder.cpp` that decodes this file and outputs the decoded text.



Hint: You can input the file `nzz.in` to your program `nzz_decoder` by calling it as follows:
`./nzz_decoder < nzz.in`
 (`nzz.in` has to be in the same folder as `nzz_decoder`.)

Judge Examples	(Explanation: http://lec.inf.ethz.ch/ifmp/2015/judge_boxes.html)
<pre>01000101 01010100 01001000 ETH</pre>	
<pre>01001000 01101001 00100000 01100001 01101100 01101100 00100001 Hi all!</pre>	
<p>Submission: https://challenge.inf.ethz.ch/team/websubmit.php?cid=5&problem=MP15081</p>	

Assignment 2 – Skript-Aufgabe 107 (4 points)

For larger floors, `shortest_path.cpp` from the lecture can become quite inefficient, since every step *i* examines *all* cells of the floor in order to find the (possibly very few) ones that have to be labeled with

i in that step. A better solution would be to examine only the neighbors of the cells that are already labeled with $i - 1$, since only these are candidates for getting label i .

Write a program `shortest_path_fast.cpp` that realizes this idea, and measure the performance gain on some larger floors. (You find a few example-floors on the course webpage).

Judge Examples	(Explanation: http://lec.inf.ethz.ch/ifmp/2015/judge_boxes.html)
<pre> 2 12 S--X-----XX XX---XX-----T SooXoooooXX XXoooXX--ooT </pre>	
Submission:	https://challenge.inf.ethz.ch/team/websubmit.php?cid=5&problem=MP15082

Assignment 3 – Skript-Aufgabe 115a (4 points)

Consider the following program:

```

#include<iostream>
int main() {
    int a[] = {5, 6, 2, 3, 1, 4, 0};
    int* p = a;
    do {
        std::cout << *p << " ";
        p = a + *p;
    } while (p != a);

    return 0;
}

```

What does this program output? Explain how the program works.

Assignment 4 (6 points)

- a) On the course website you find a template `iteration_template_a.cpp`. Extend it with a code snippet which outputs every second element (starting from index 0) in the given array using pointers. **You are not allowed to use `[]` to access elements of the array.**

Judge Examples	(Explanation: http://lec.inf.ethz.ch/ifmp/2015/judge_boxes.html)
<pre> 0 1 2 3 4 5 6 7 8 9 0 2 4 6 8 </pre>	

Submission: <https://challenge.inf.ethz.ch/team/websubmit.php?cid=5&problem=MP15084a>

- b) Extend `iteration_template_b.cpp` with a code snippet which outputs the given array in reverse order using pointers. **You are not allowed to use `[]` to access elements of the array.**

Judge Examples

(Explanation: http://lec.inf.ethz.ch/ifmp/2015/judge_boxes.html)

```
0 1 2 3 4 5 6 7 8 9
9 8 7 6 5 4 3 2 1 0
```

Submission: <https://challenge.inf.ethz.ch/team/websubmit.php?cid=5&problem=MP15084b>

- c) Extend `iteration_template_c.cpp` with a code snippet so that after your code snipped the pointer `max_ptr` points to the largest array element (the first one if the maximum is not unique). **You are not allowed to use `[]` to access elements of the array.**

Judge Examples

(Explanation: http://lec.inf.ethz.ch/ifmp/2015/judge_boxes.html)

```
0 1 2 3 2 1 -1 2 2 1
0 0 0 3 0 0 0 0 0 0
```

Submission: <https://challenge.inf.ethz.ch/team/websubmit.php?cid=5&problem=MP15084c>

Challenge – Skript-Aufgabe 106 (Lindenmayer Systems) (8 points)

Did you always dream of producing a picture that will be shown in front of 400 students, and that has the chance to become famous through its appearance in the lecture's podcast? Well, now you can! This challenge asks you to make beautiful fractal images using Lindenmayer systems. All submissions will be made into a video to be shown during a later lecture.

Don't forget to write some recommended parameter settings (for instance the number of iterations) as a comment in your `cpp`-files, and also include your name. Hand in your `cpp`-files directly to your exercise class teacher. You can also hand in multiple submissions!