

Homework 2 of CSC I0600 Fundamental Algorithms, Fall 2020

given October 6, 2020, due October 31

The aim of this homework is to create an approximate solution for the zookeeper problem. The zookeeper has to visit all cages in the zoo (rectangles) but does not enter any cage. You write a function that gets an array of rectangles, and produces a closed polygonal path that touches each rectangle, but does not enter the interior of any rectangle. You write a function

```
int zoopath( int *rectangles, int n, int *path, int m)
```

The rectangles are given as an $n \times 4$ -array, the i -th rectangle being

```
[rectangles[i][0],rectangles[i][1]]  $\times$  [rectangles[i][2],rectangles[i][3]]
```

For the polygonal path you receive a $m \times 2$ -array, which you fill in up to whatever path length you require; the return value of the function is the length of the path. I will provide again an xlib-based main function which calls your function, shows the rectangles and the path.

Your code needs to be C or C++, I need to be able to compile it on my laptop, and test it. Do not share code, and do not use code you found on the web. It is essential that what you submit is your code, anything else is cheating.