



Algorithms and Datastructures

Sample Solution Exercise Sheet 12

Exercise 1: Rabin-Karp Algorithm

(10 Points)

- (a) Implement the Rabin-Karp algorithm. You may use the template `StringMatching.py`. The algorithm should return a Python-list containing all starting points of the pattern. That is, for each time the pattern is recognized, the list should contain the position of the first letter of this appearance. *(5 Points)*
- (b) Run your algorithm on the text and pattern given in `input.txt`. Write down your output. *(5 Points)*

Remark: When choosing the parameters b and M , consider that the procedure `read_input` used on `input.txt` creates an array with values from `ord(' ') = 32` (whitespace) to `ord('z') = 122`.

Sample Solution

- (a) C.f. `StringMatching.py`.
- (b) The desired output is:
 [212, 2194, 2604, 5208, 7193, 7443, 7939, 10245, 11594, 13544, 14276, 22354, 25024, 28735, 39999, 40835, 46199].

Exercise 2: Knuth-Morris-Pratt Algorithmus

(10 Points)

Consider the pattern $P = AABAAA$ and the text $T = BAABAABAAABAAABBA$.

- (a) Compute the array S of the Knuth-Morris-Pratt algorithm. *(5 Points)*
- (b) Use the Knuth-Morris-Pratt algorithm to find all appearances of P in T . Document the steps analogously to the lecture. *(5 Points)*

Sample Solution

- (a) $S = [-1, 0, 1, 0, 1, 2, 2]$

- (b)
- | | | | | | | | | | | | | | | | | | |
|----------|---|---|---|---|---|----------|---|---|---|---|---|---|----------|---|---|---|---|
| B | A | A | B | A | A | B | A | A | A | B | A | A | A | B | B | A | A |
| <u>A</u> | A | B | A | A | A | | | | | | | | | | | | |
| | A | A | B | A | A | <u>A</u> | | | | | | | | | | | |
| | | | | A | A | B | A | A | A | | | | | | | | ✓ |
| | | | | | | | A | A | B | A | A | A | | | | | ✓ |
| | | | | | | | | | | A | A | B | <u>A</u> | A | A | | |
| | | | | | | | | | | | | | <u>A</u> | A | B | | |
| | | | | | | | | | | | | | | A | A | | |