UNIVERSITY OF OSLO

Faculty of Mathematics and Natural Sciences

Exam in:	**INF 4130/9135:** ***Algoritmer: Design og effektivitet***
Date of exam:	**14th December 2012**
Exam hours:	**09:00 – 13:00 (4 hours)**
Exam paper consists of:	**4 pages**
Appendices:	**None**
Permitted materials:	**All written and printed**

Make sure that your copy of this examination paper is complete before answering.

Read the text carefully, and good luck!

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.a (6%)

We will work with an alphabet that consists of the nine first letters of the standard a-z alphabet: { a, b, c, d, e, f, g, h, i }. We are given a string P = "abcifbei" to search for in a longer string T. Set up the table of Shift values for the Boyer-Moore algorithm (the simplified version discussed in the textbook, which is called Horspool on the slides). You can use the letters themselves as indices to the Shift table/array.

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.a (6%)

We will work with an alphabet that consists of the nine first letters of the standard a-z alphabet: { a, b, c, d, e, f, g, h, i }. We are given a string P = "abcifbei" to search for in a longer string T. Set up the table of Shift values for the Boyer-Moore algorithm (the simplified version discussed in the textbook, which is called Horspool on the slides). You can use the letters themselves as indices to the Shift table/array.

P = a b c i f b e i

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.a (6%)

We will work with an alphabet that consists of the nine first letters of the standard a-z alphabet: { a, b, c, d, e, f, g, h, i }. We are given a string P = "abcifbei" to search for in a longer string T. Set up the table of Shift values for the Boyer-Moore algorithm (the simplified version discussed in the textbook, which is called Horspool on the slides). You can use the letters themselves as indices to the Shift table/array.

P	=	a	b	c	i	f	b	e	i
		7	6	5	4	3	2	1	0

← Distance from the end of P

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.a (6%)

We will work with an alphabet that consists of the nine first letters of the standard a-z alphabet: { a, b, c, d, e, f, g, h, i }. We are given a string *P* = "abcifbei" to search for in a longer string *T*. Set up the table of Shift values for the Boyer-Moore algorithm (the simplified version discussed in the textbook, which is called Horspool on the slides). You can use the letters themselves as indices to the Shift table/array.

P	=	a	b	c	i	f	b	e	i
		7	6	5	4	3	2	1	0

Shift values	
a	**7**
b	***2***
c	**5**
d	8
e	**1**
f	**3**
g	8
h	8
i	***4***

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.b (6%)

Suppose that we do string search with the Boyer-Moore algorithm (the same as in 1.a), and that m is the length of the pattern P and n is the length of the string T (and that n is much larger than m). By coincidence in a search, the last symbol of P does not occur elsewhere in P and not at all in T. Estimate the number of tests for symbol equality that will be made during the search.

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.b (6%)

Suppose that we do string search with the Boyer-Moore algorithm (the same as in 1.a), and that m is the length of the pattern P and n is the length of the string T (and that n is much larger than m). By coincidence in a search, the last symbol of P does not occur elsewhere in P and not at all in T. Estimate the number of tests for symbol equality that will be made during the search.

There is a slight inaccuracy in this question. The intention was that the pattern should be shifted m steps each time. For this to occur with the Horspool algorithm, the last symbol of every *window* cannot be present in P. (The window is the m symbols of T that we currently compare with P.) In other words, the symbols $T[m]$, $T[2m]$, …, $T[\lfloor n/m \rfloor \cdot m]$ can not occur in P. Remember that we shift based on the last symbol of the window, not the pattern, when we get a mismatch.

The string T = "abcdefghijklmn" and the pattern P = "gedx" fit with the description in the question (x only occurs at end of P, and nowhere in T). We get the following shift values: Shift(g) = 3, Shift (e) = 2, Shift (d) = 1, and Shift(*) = 4, where * indicates the remaining symbols in our alphabet. Running the algorithm gives us shifts of 1 (when x ≠ d), 2 (when x ≠ e), 3 (when x ≠ g), and 4 (when x ≠ j).

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.b (6%)

Suppose that we do string search with the Boyer-Moore algorithm (the same as in 1.a), and that m is the length of the pattern P and n is the length of the string T (and that n is much larger than m). By coincidence in a search, the last symbol of P does not occur elsewhere in P and not at all in T. Estimate the number of tests for symbol equality that will be made during the search.

Keeping with the spirit of the question, we assume $T[i \cdot m] \notin P$, for $i = 1, 2, \ldots, \lfloor n/m \rfloor$, so that we always get a shift of m.

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.b (6%)

Suppose that we do string search with the Boyer-Moore algorithm (the same as in 1.a), and that m is the length of the pattern P and n is the length of the string T (and that n is much larger than m). By coincidence in a search, the last symbol of P does not occur elsewhere in P and not at all in T. Estimate the number of tests for symbol equality that will be made during the search.

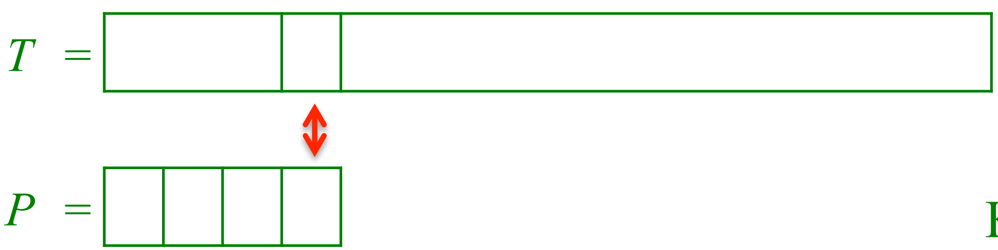
T =

Keeping with the spirit of the question, we assume $T[i \cdot m] \notin P$, for $i = 1, 2, \ldots, \lfloor n/m \rfloor$, so that we always get a shift of m.

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.b (6%)

Suppose that we do string search with the Boyer-Moore algorithm (the same as in 1.a), and that m is the length of the pattern P and n is the length of the string T (and that n is much larger than m). By coincidence in a search, the last symbol of P does not occur elsewhere in P and not at all in T. Estimate the number of tests for symbol equality that will be made during the search.

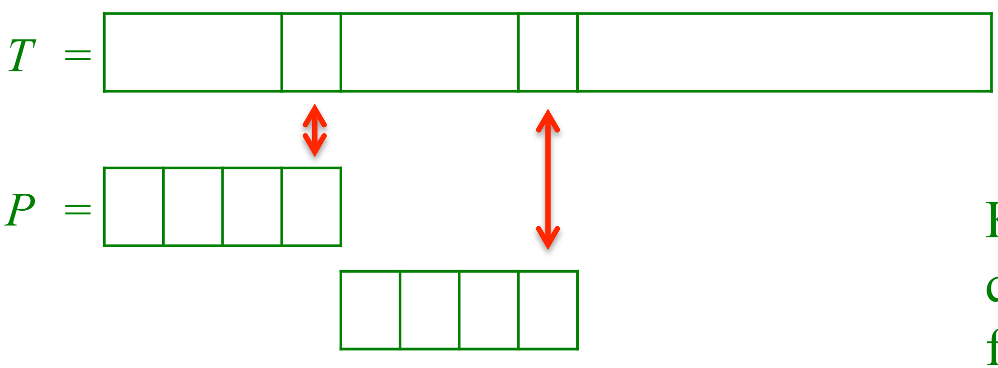


Keeping with the spirit of the question, we assume $T[i \cdot m] \notin P$, for $i = 1, 2, \ldots, \lfloor n/m \rfloor$, so that we always get a shift of m.

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.b (6%)

Suppose that we do string search with the Boyer-Moore algorithm (the same as in 1.a), and that m is the length of the pattern P and n is the length of the string T (and that n is much larger than m). By coincidence in a search, the last symbol of P does not occur elsewhere in P and not at all in T. Estimate the number of tests for symbol equality that will be made during the search.

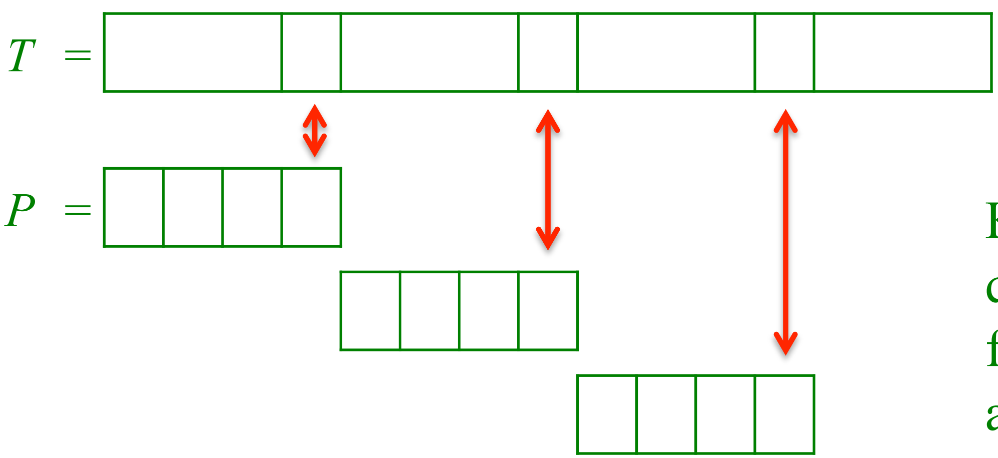


Keeping with the spirit of the question, we assume $T[i \cdot m] \notin P$, for $i = 1, 2, \ldots, \lfloor n/m \rfloor$, so that we always get a shift of m.

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.b (6%)

Suppose that we do string search with the Boyer-Moore algorithm (the same as in 1.a), and that m is the length of the pattern P and n is the length of the string T (and that n is much larger than m). By coincidence in a search, the last symbol of P does not occur elsewhere in P and not at all in T. Estimate the number of tests for symbol equality that will be made during the search.

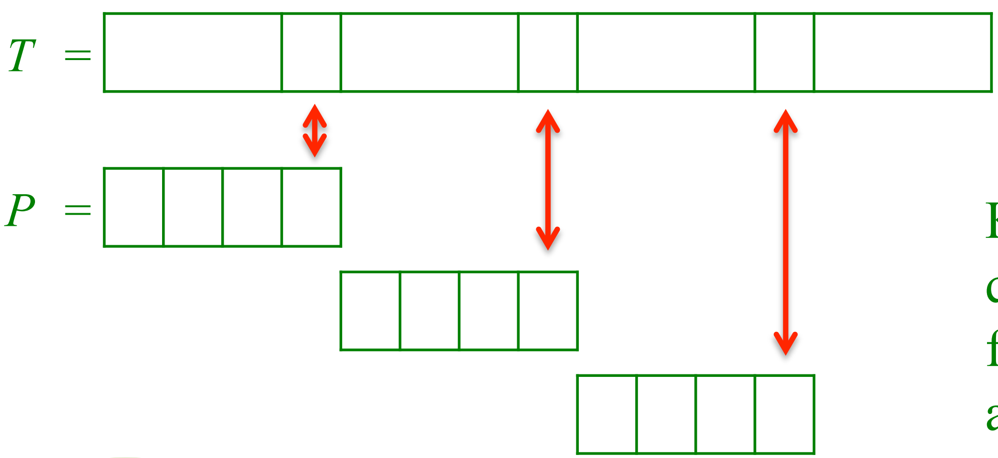


Keeping with the spirit of the question, we assume $T[i \cdot m] \notin P$, for $i = 1, 2, \ldots, \lfloor n/m \rfloor$, so that we always get a shift of m.

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.b (6%)

Suppose that we do string search with the Boyer-Moore algorithm (the same as in 1.a), and that m is the length of the pattern P and n is the length of the string T (and that n is much larger than m). By coincidence in a search, the last symbol of P does not occur elsewhere in P and not at all in T. Estimate the number of tests for symbol equality that will be made during the search.



Keeping with the spirit of the question, we assume $T[i \cdot m] \notin P$, for $i = 1, 2, \ldots, \lfloor n/m \rfloor$, so that we always get a shift of m.

$O(\lfloor n/m \rfloor)$

Assignment 1: Search in strings with Boyer-Moore (18%)

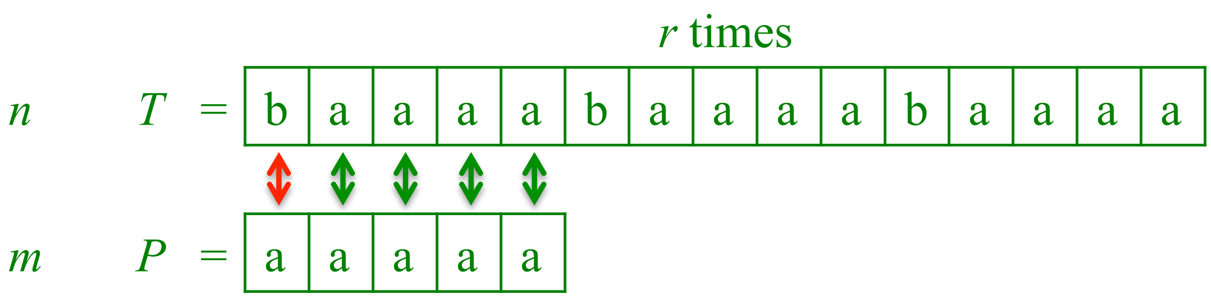
Question 1.c (6%)

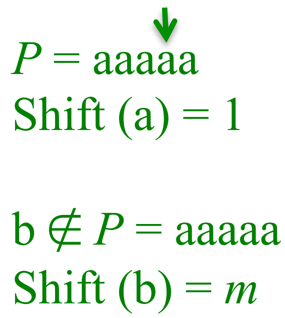
We will study what will happen if P consists of the letter ‘a’ repeated m times (that is: a^m), and T is a string that is a repetition r times of the string $b\,a^{m-1}$. How many test for symbol equality will be made during the search?

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.c (6%)

We will study what will happen if P consists of the letter 'a' repeated m times (that is: a^m), and T is a string that is a repetition r times of the string $b\,a^{m-1}$. How many test for symbol equality will be made during the search?

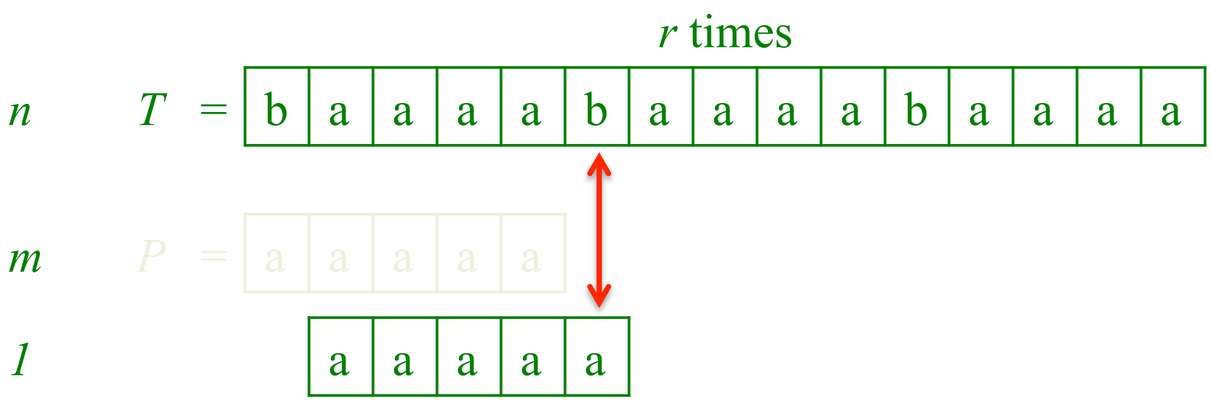




Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.c (6%)

We will study what will happen if P consists of the letter ‘a’ repeated m times (that is: a^m), and T is a string that is a repetition r times of the string $b\,a^{m\text{-}1}$. How many test for symbol equality will be made during the search?



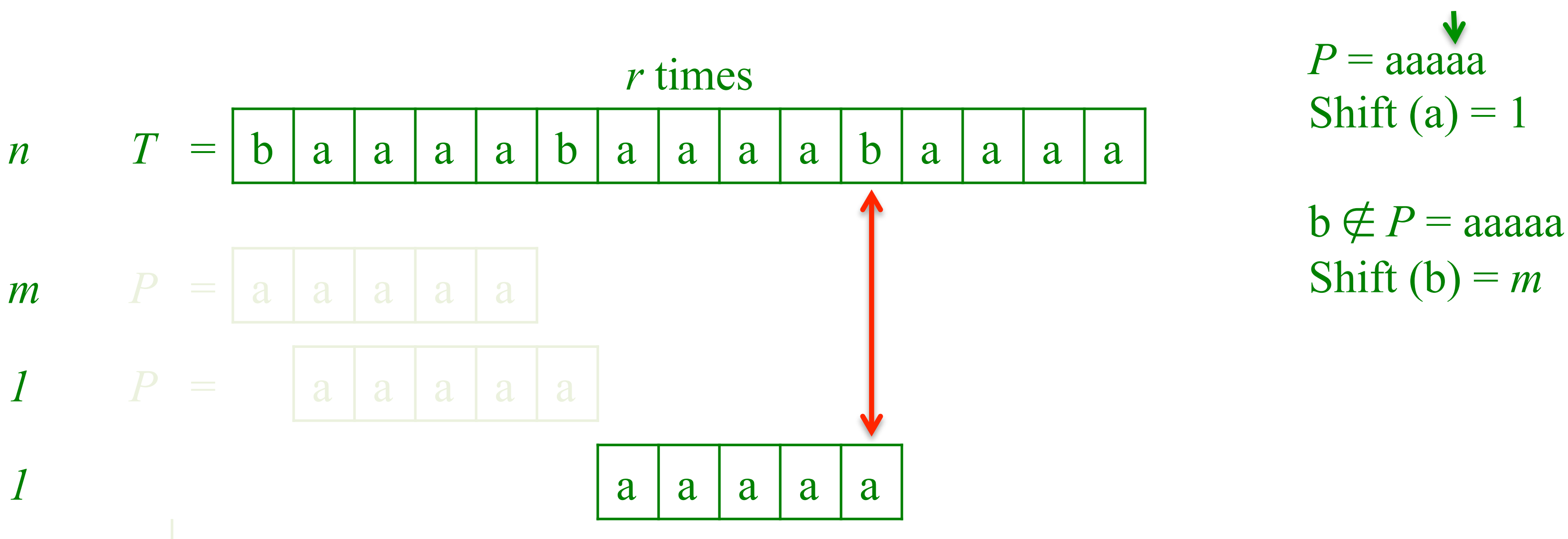
P = aaaaa
Shift (a) = 1

b $\notin$ P = aaaaa
Shift (b) = m

Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.c (6%)

We will study what will happen if P consists of the letter 'a' repeated m times (that is: a^m), and T is a string that is a repetition r times of the string $b\,a^{m-1}$. How many test for symbol equality will be made during the search?



Assignment 1: Search in strings with Boyer-Moore (18%)

Question 1.c (6%)

We will study what will happen if P consists of the letter 'a' repeated m times (that is: a^m), and T is a string that is a repetition r times of the string $b\,a^{m\text{-}1}$. How many test for symbol equality will be made during the search?