String Algorithms and Data Structures
The Z-algorithm

CS 199-225
Brad Solomon

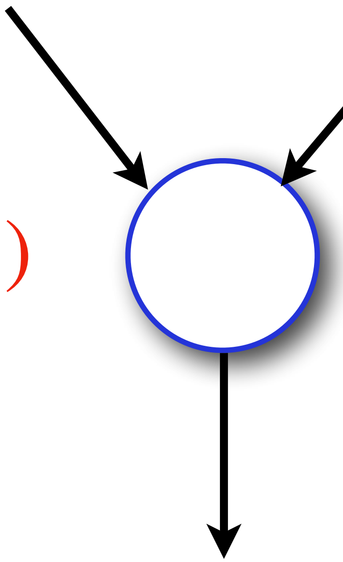
February 12, 2023

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Department of Computer Science

Exact Pattern Matching w/ Z-algorithm

Pattern, P *Text, T*



Naive $\approx \theta(|P||T|)$

Z-Algorithm $\approx \theta(|P| + |T|)$

Find instances of *P* in *T*

‘instances’: An exact, full length copy

The Z-value [$Z_i(S)$]

Given a string S, $Z_i(S)$ is the length of the longest substring in S, starting at position $i > 0$, that matches a prefix of S.

```
    0 1 2 3 4 5 6 7 8 9
S:  A B C D A B C D A B
```

$Z_4(S) =$

```
S:  C G C G A ? ? ? ? ?
```

$Z_5(S) = 3$

```
S:  A ? ? ? ? ? ? ? ? ?
```

$Z_1(S) = 7$

The Z-Algorithm

```
S: 101$101011
   01$101011
   1$101011
   $101011
   101011
   01011
   1011
   011
   11
   1
```

The Z-Algorithm

$Z_1 = 3$

$Z_2 =$

0	1	2	3	4	5	6	7
A	A	A	A	B	B	B	B
A	A	A	A	B	B	B	B

We track our current knowledge of *S* using three values: i, r, l

i gets updated every iteration (as we compute Z_i)

r gets updated when $Z_i > 0$ *AND* $r_{new} > r_{old}$

l gets updated whenever r is updated (it stores the index of r's Z-value)

The Z-Algorithm

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

The Z-Algorithm

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

The Z-Algorithm

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

The Z-Algorithm

0	1	2	3	4	5	6	7
A	A	A	B	B	A	A	A
A	A	A	B	B	A	A	A

The values of i, r, l tell us how much work we need to do to compute Z_i

Case 1: $i > r$

Ex: $i = 1, r = 0, l = 0$

We must compute Z_i explicitly!

The Z-Algorithm

0	1	2	3	4	5	6	7
A	A	A	B	B	A	A	A
A	A	A	B	B	A	A	A

The values of i, r, l tell us how much work we need to do to compute Z_i

Case 1: $i > r$

Ex: $i = 5, r = 2, l = 1$

We must compute Z_i explicitly!

The Z-Algorithm

0	1	2	3	4	5	6	7
A	A	A	B	B	A	A	A
A	A	A	B	B	A	A	A

The values of i, r, l tell us how much work we need to do to compute Z_i

Case 2: $i \leq r$

Ex: $i = 6, r = 7, l = 5$

To find Z_6, we can save time by looking up the value ______________

The Z-Algorithm

0	1	2	3	4	5	6	7
A	B	C	B	A	B	C	A
A	B	C	B	A	B	C	A

The values of i, r, l tell us how much work we need to do to compute Z_i

Case 2: $i \leq r$

Ex: $i = 5, r = 6, l = 4$

To find Z_5, we can save time by looking up the value ______________

The Z-Algorithm

0	1	2	3	4	5	6	7
A	A	B	A	A	A	B	C
A	A	B	A	A	A	B	C

The values of i, r, l tell us how much work we need to do to compute Z_i

Case 2: $i \leq r$

Ex: $i = 4, r = 4, l = 3$

To find Z_4, we can save time by looking up the value ______________

The Z-Algorithm

Let $l = 0, r = 0$, for $i = [1, \ldots, |S| - 1]$:

Compute Z_i using irl:

Case 1 ($i > r$): Compute explicitly; update irl

Case 2 ($i \leq r$):

- Use previous Z-values to avoid work
- Explicitly compute only *'new'* characters
- How can we tell the difference between cases?

The Z-Algorithm

$i = 6, r = 7, l = 5$

0	1	2	3	4	5	6	7	8
A	**A**	**A**	**A**	**C**	**A**	**A**	**A**	**B**
A	A	A	A	C	A	A	A	B
A	A	A	A	C	A	A	A	B

The amount of work required depends on two pieces of information

1. # of characters at or after i that we have seen before

2. The Z-value that matches part or all of the string starting at i

The Z-Algorithm

$i = 6, r = 7, l = 5$

0	1	2	3	4	5	6	7	8
A	**A**	**A**	**A**	**C**	**A**	**A**	**A**	**B**
A	A	A	A	C	A	A	A	B
A	A	A	A	C	A	A	A	B

The amount of work required depends on two pieces of information

1. # of characters at or after i that we have seen before

Call this value $|\beta|$. What is $|\beta|$ in terms of i, r, l?

The Z-Algorithm

$i = 6, r = 7, l = 5$

0	1	2	3	4	5	6	7	8
A	**A**	**A**	**A**	**C**	**A**	**A**	**A**	**B**
A	A	A	A	C	A	A	A	B
A	A	A	A	C	A	A	A	B

The amount of work required depends on two pieces of information

2. The Z-value that matches part or all of the string starting at i

Call this value Z_k. What is k in terms of i, r, l ?

The Z-Algorithm

$i = 6, r = 7, l = 5$

	0	1	2	3	4	5	6	7	8
	A	**A**	**A**	**A**	**C**	**A**	**A**	**A**	**B**
	A	A	A	A	C	A	A	A	B
$Z_k = Z_1 = 3$	A	A	A	A	C	A	A	A	B

The amount of work required depends on two pieces of information

1. # of characters at or after i that we have seen before

$$|\beta| = 7 - 6 + 1 = 2$$

2. The Z-value that matches part or all of the string starting at i

$$k = 6 - 5 = 1$$

The Z-Algorithm

$i = 5, r = 7, l = 4$

0	1	2	3	4	**5**	6	7
A	A	A	B	A	A	A	B
A	A	A	B	A	A	A	B

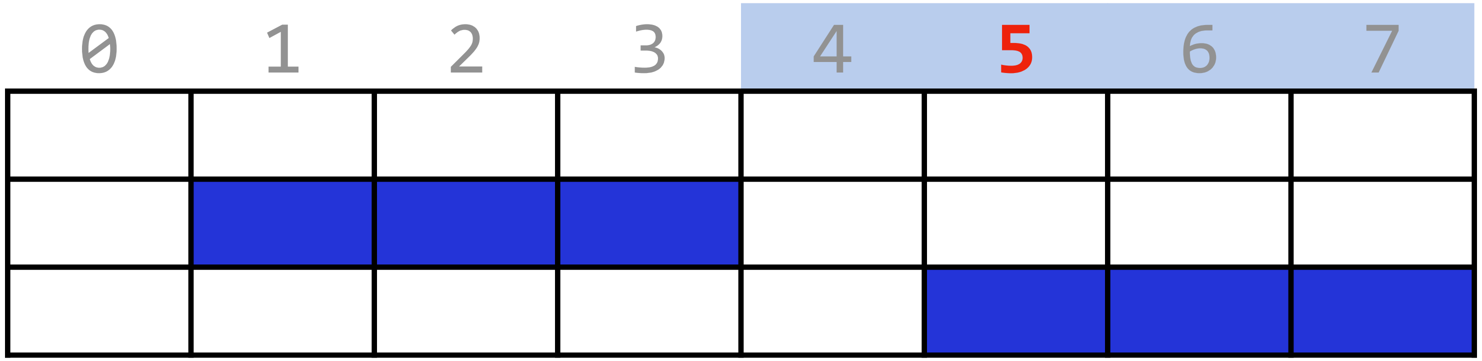
Case 2a: $i \leq r, Z_k < |\beta|$

$|\beta| =$ __________, $k =$ __________, $Z_k =$ __________

$Z_i =$ __________

The Z-Algorithm

$i = 5, r = 7, l = 4$

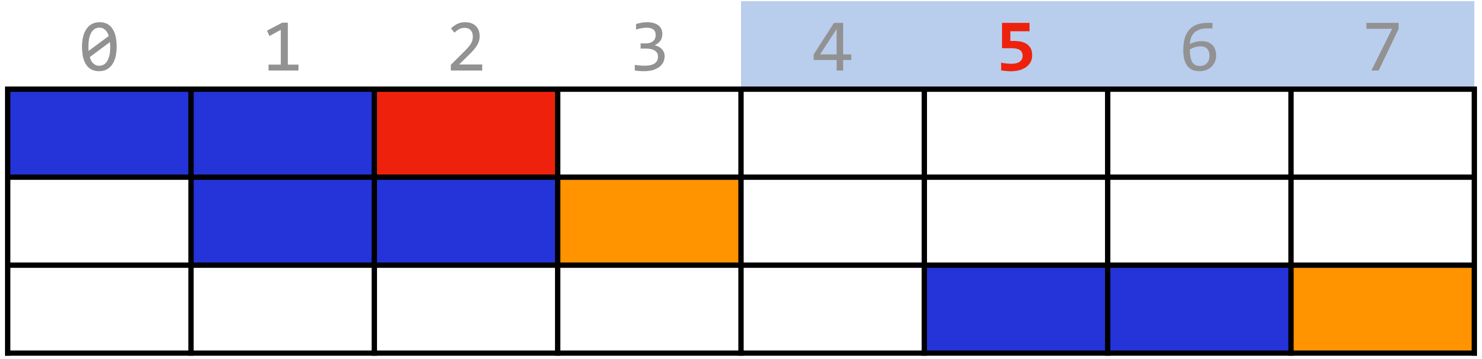


Case 2a: $i \leq r, Z_k < \beta$

Z_l (defined by r, l) tells us that β matches earlier.

The Z-Algorithm

$i = 5, r = 7, l = 4$

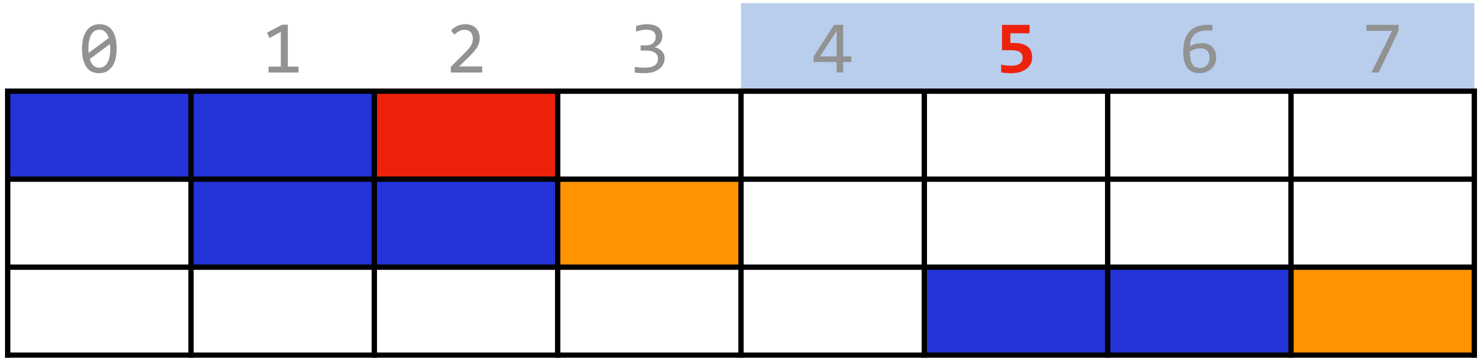


Case 2a: $i \leq r$, $Z_k < |\beta|$

Z_l tells us that β matches earlier. Z_k tells us how much matches the prefix.

The Z-Algorithm

$i = 5, r = 7, l = 4$



Case 2a: $i \leq r$, $Z_k < |\beta|$

Z_l tells us that β matches earlier. Z_k tells us how much matches the prefix.

Because $Z_k < |\beta|$, $Z_i =$ ___________

The Z-Algorithm

$i = 4, r = 4, l = 3$

0	1	2	3	4	5	6	7
A	A	B	A	A	A	B	C
A	A	B	A	A	A	B	C

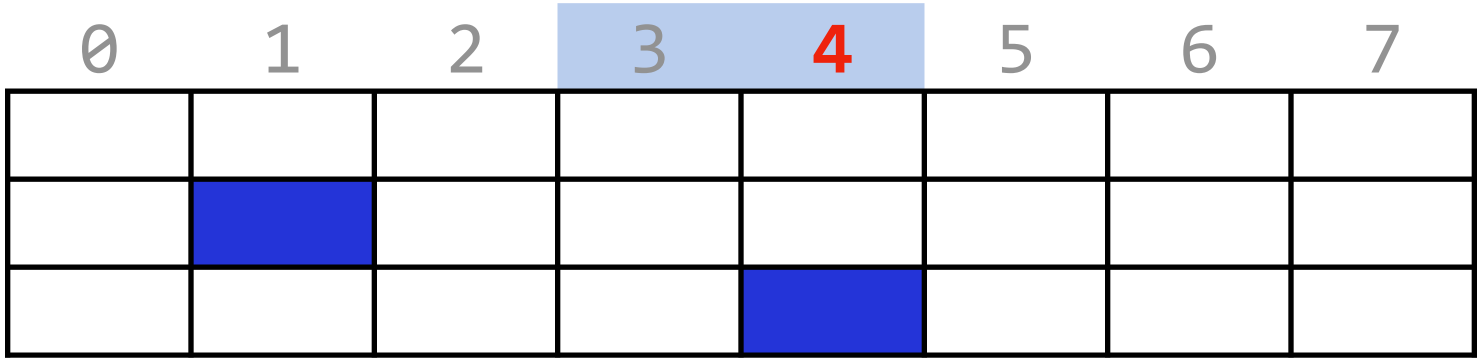
Case 2b: $i \le r, Z_k = |\beta|$

$|\beta| =$ __________, $k =$ __________, $Z_k =$ __________

$Z_i =$ __________

The Z-Algorithm

$i = 4, r = 4, l = 3$

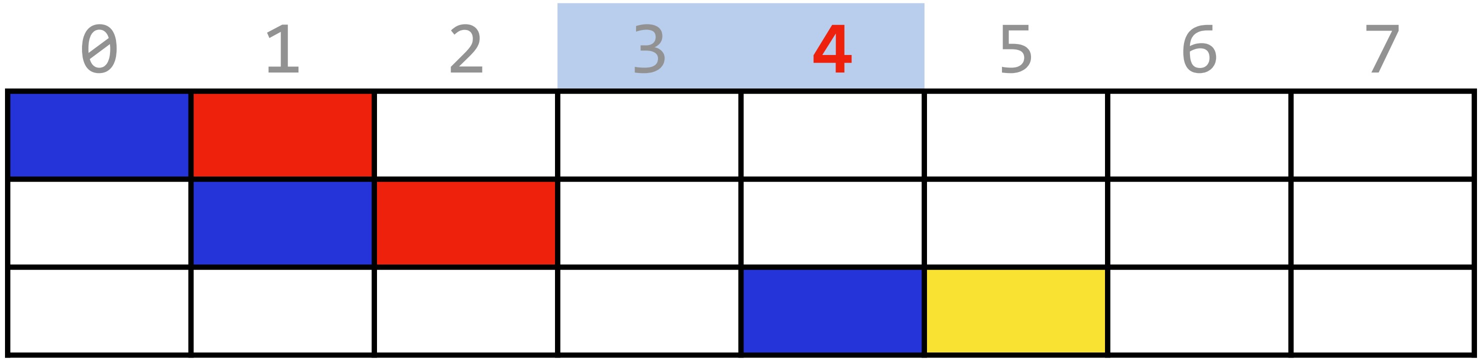


Case 2b: $i \leq r$, $Z_k = |\beta|$

Z_l (defined by r, l) tells us that β matches earlier.

The Z-Algorithm

$i = 4, r = 4, l = 3$



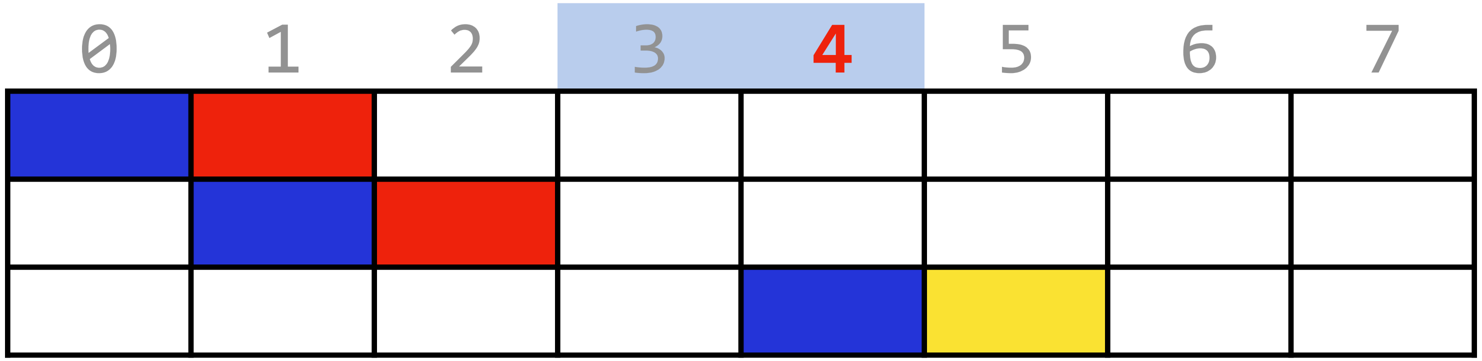
Case 2b: $i \leq r$, $Z_k = |\beta|$

Z_l (defined by r, l) tells us that β matches earlier.

Z_k tells us how much matches the prefix… but not everything!

The Z-Algorithm

$i = 4, r = 4, l = 3$



Case 2b: $i \leq r, Z_k = |\beta|$

We have all the same info as before but we have unseen characters!

Because $Z_k = |\beta|, Z_i =$ ___________

The Z-Algorithm

$i = 3, r = 5, l = 1$

0	1	2	3	4	5	6	7
A	A	A	A	A	A	B	C
A	A	A	A	A	A	B	C

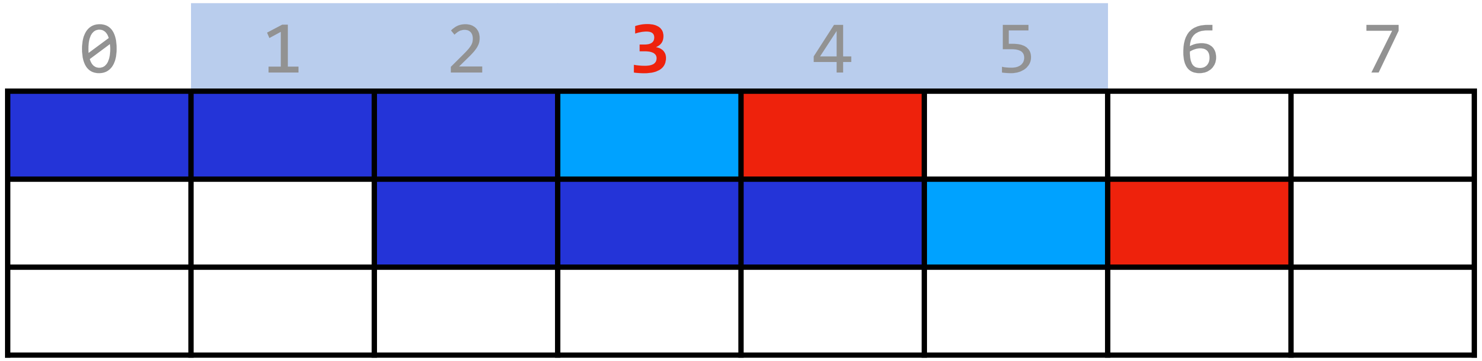
Case 2c: $i \leq r, Z_k > |\beta|$

$|\beta| =$ __________, $k =$ __________, $Z_k =$ __________

$Z_i =$ __________

The Z-Algorithm

$i = 3, r = 5, l = 1$

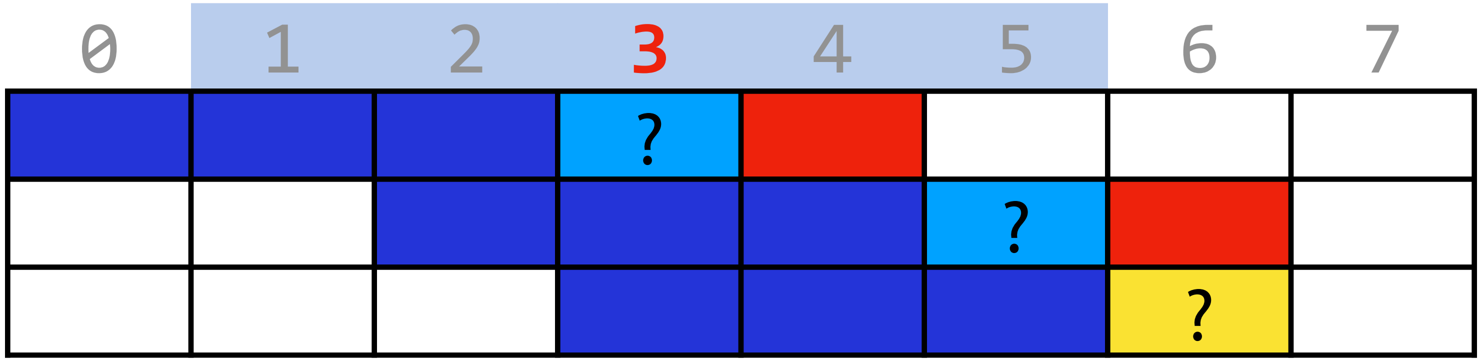


Case 2c: $i \leq r, Z_k > |\beta|$

Z_k tells us how much matches the prefix.

The Z-Algorithm

$i = 3, r = 5, l = 1$



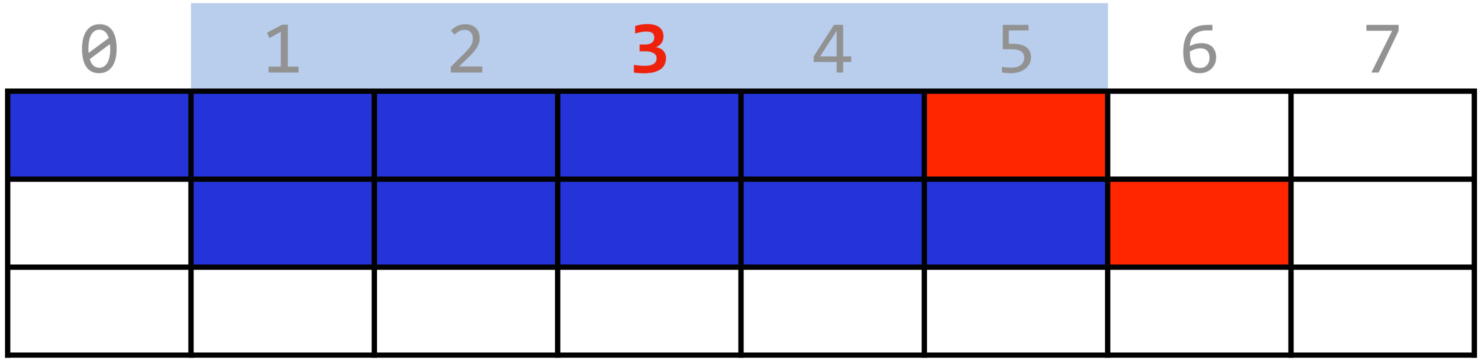
Case 2c: $i \leq r, Z_k > |\beta|$

Z_l tells us that β matches earlier. Z_k tells us how much matches the prefix.

What do we know about yellow?

The Z-Algorithm

$i = 3, r = 5, l = 1$



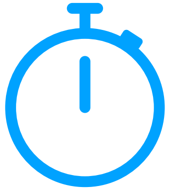
Case 2c: $i \leq r, Z_k > |\beta|$

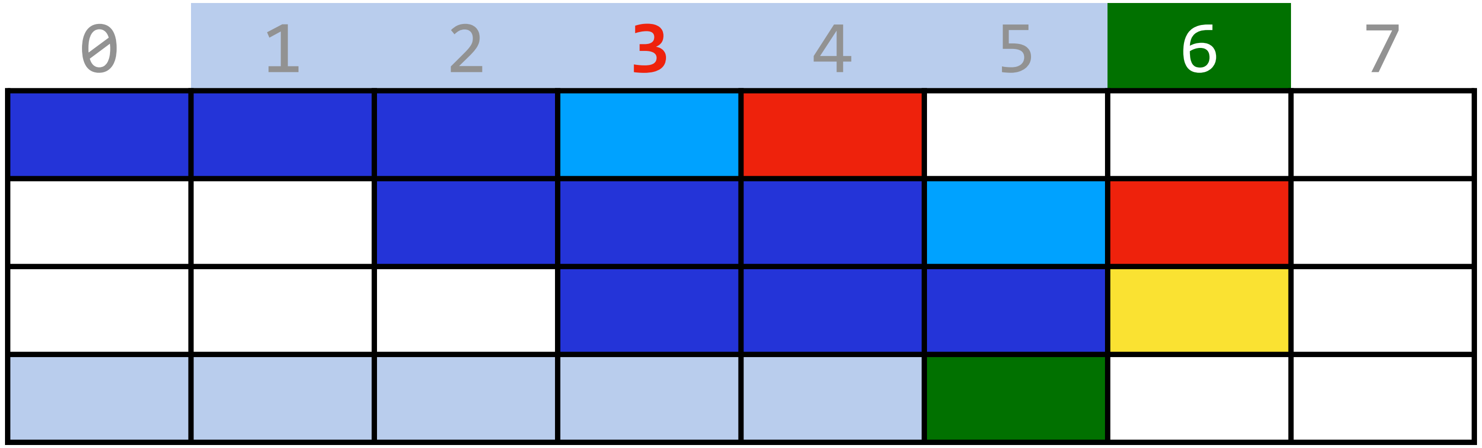
Z_l tells us that our entire range (β included) matches earlier

… and that it failed to match the next character.

The Z-Algorithm

$i = 3, r = 5, l = 1$





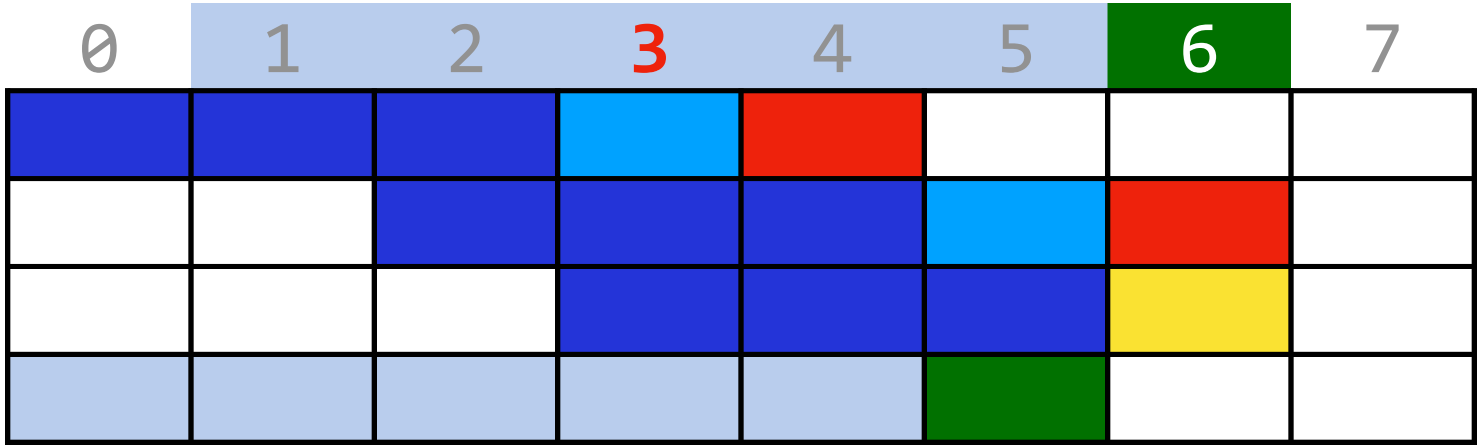
Case 2c: $i \leq r$, $Z_k > |\beta|$

Z_l tells us that β matches earlier. Z_k tells us how much matches the prefix.

Z_l also tells us that yellow and green can't be equal!

The Z-Algorithm

$i = 3, r = 5, l = 1$



Case 2c: $i \leq r, Z_k > |\beta|$

Z_l tells us that β is our prefix. Z_k is also a previously computed prefix.

Because $Z_k > |\beta|, Z_i =$ ___________

The Z-Algorithm

Let $l = 0, r = 0$, for $i = [1, \ldots, |S| - 1]$:

Compute Z_i using irl:

Case 1 ($i > r$): Compute explicitly; update irl

Case 2 ($i \leq r$):

2a: ($Z_k < |\beta|$): $Z_i = Z_k$

2b: ($Z_k = |\beta|$): $Z_i = Z_k$+ *explicit*(r+1); update irl

2c: ($Z_k > |\beta|$): $Z_i = |\beta|$

Assignment 3: a_zalg

Learning Objective:

Construct the full Z-algorithm and measure its efficiency

Demonstrate use of Z-algorithm in pattern matching

Consider: Our goal is $\theta(|P| + |T|)$. Does Z-alg search match this?

Next week:

If I gave you the pattern I was interested in ahead of time, what could you pre-compute to speed up search?

Ex: I’m going to try to look up the word **‘arrays’** — but you don’t know what text I’m going to search through.