



# String Matching

By Joshua Yudaken

# Terms

## ◆ Haystack

- A string in which to search

## ◆ Needle

- The string being searched for

- ◆ find the needle in the haystack

# Basic Algorithm

- ◆ For each letter in the haystack
  - Check if the needle is present

```
for (int k = 0; k < (strlen(haystack) - strlen(needle) + 1); k++)  
    if (memcmp(haystack, needle, strlen(needle)) == 0)  
        return true;  
return false;
```

# Boyer-Moore Algorithm

Haystack = "jim saw it at a barbershop"

Needle = "barber"

JIM SAW IT AT THE BARBERSHOP

BARBER

# Boyer-Moore Algorithm

Haystack = "jim saw it at a barbershop"

Needle = "barber"

JIM SAW IT AT THE BARBERSHOP

BARBER

BARBER

# Boyer-Moore Algorithm

Haystack = "jim saw it at a barbershop"

Needle = "barber"

JIM SAW IT AT THE BARBERSHOP

BARBER

BARBER

BARBER

# Boyer-Moore Algorithm

Haystack = "jim saw it at a barbershop"

Needle = "barber"

JIM SAW IT AT THE BARBERSHOP

BARBER

BARBER

BARBER

BARBER

# Boyer-Moore Algorithm

Haystack = "jim saw it at a barbershop"

Needle = "barber"

JIM SAW IT AT THE BARBERSHOP

BARBER

BARBER

BARBER

BARBER

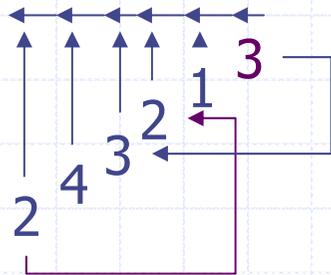
BARBER



# Space/Time Tradeoff

- ◆ Create a 'shift table' for the search

For "barber": [a: 4, b: 2, c: 6, d: 6, e: 1,...,r: 3,..., z: 6, ` `: 6]



- Set every element in the shift table to the length of the needle
- Go through each letter (from the second last to the first) of the needle
- If the distance from the letter to the end of the needle, is less than the letters current value in the shift table
  - ◆ update the shift table with the distance to the end of the needle

# Best when

- ◆ The “alphabet” used in the haystack is much larger than that of the needle.
- ◆ The haystack is long
- ◆ The same needle is to be used in many different searches
  
- ◆ In other cases, use `strstr()`!
  - Or a different available function.