Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

1. Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

What is an Algorithm?

An algorithm is a set of clear, step-by-step instructions used to solve a problem or complete a task. Algorithms are used in everyday life (like cooking a recipe) and in computer science (like searching or sorting data).

In programming, algorithms help computers solve problems efficiently. Examples include:

- Sorting algorithms (like Bubble Sort, Quick Sort)
- Searching algorithms (like Binary Search)
- Graph algorithms (like Dijkstra's for shortest path)

A good algorithm is:

- Correct (it gives the right answer)
- Efficient (it runs fast and uses little memory)
- Simple (easy to understand and maintain)

Knuth-Morris-Pratt (KMP) Algorithm

The KMP algorithm is a famous string searching algorithm. It finds the position of a pattern (substring) inside a text efficiently.

It was created by Donald Knuth, Vaughan Pratt, and James H. Morris in 1977.

Problem it Solves:

Algorithms and Knuth-Morris-Pratt (KMP) Algorithm

Imagine you want to find the word 'abc' inside a long text like 'abababcabc'. A simple approach would check every character one by one - that can be slow.

KMP improves this by not re-checking characters that have already been matched.

How it Works:

1. Prefix Table (also called LPS array): KMP pre-processes the pattern to build a table that tells where to jump when a mismatch happens.

2. Search Phase: It uses this table to skip unnecessary comparisons.

This makes KMP faster than naive search, especially for long texts or repeating patterns.

Time Complexity:

- Preprocessing (LPS table): O(m), where m is the length of the pattern.

- Search: O(n), where n is the length of the text.

So overall, KMP works in O(n + m) time - very efficient!

Example:

Find 'ababc' in 'ababcababc'. KMP uses the prefix table to avoid restarting from the beginning and finds all matches quickly.