

# Relazione per Progetto Laboratorio di ASD

## Prima Parte

Marco Giunta giunta.marco@spes.uniud.it  
Samuele Anzolin anzolin.samuele@spes.uniud.it  
Federico Casani casani.federico@spes.uniud.it  
Gianluca Giuseppe Maria De Nardi 142733@spes.uniud.it

Maggio 2021

# Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
<b>2</b>	<b>Ipotesi</b>	<b>2</b>
<b>3</b>	<b>Implementazione</b>	<b>2</b>
3.1	periodNaive . . . . .	2
3.2	periodSmart . . . . .	2
3.3	Algoritmo per il calcolo dei tempi medi . . . . .	2
<b>4</b>	<b>Analisi dei dati ottenuti</b>	<b>2</b>
4.1	Grafico dei tempi di periodNaive . . . . .	2
4.2	Grafico dei tempi di periodSmart . . . . .	3
4.3	Analisi logaritmica dei due algoritmi . . . . .	3
4.4	Analisi comparativa tra i due algoritmi . . . . .	3
<b>5</b>	<b>Conclusioni</b>	<b>3</b>

# 1 Introduzione

In questo progetto abbiamo implementato e analizzato due algoritmi per il calcolo del periodo frazionario minimo di una stringa. I due algoritmi implementati sono:

- PeriodNaive
- PeriodSmart

Il linguaggio di programmazione che abbiamo utilizzato per questo progetto è C, in quanto è un linguaggio veloce ed efficiente e qualitativamente migliore per un'analisi temporale.

# 2 Ipotesi

Essendo periodNaive un algoritmo con complessità asintotica nel caso peggiore pari a  $O(n^2)$  mentre il secondo raggiunge una complessità lineare, possiamo dedurre che periodSmart sarà notevolmente migliore nel caso peggiore.

# 3 Implementazione

## 3.1 periodNaive

L'implementazione di questo algoritmo come dice il suo nome, è abbastanza intuitiva.

Utilizziamo quindi un ciclo che scandisca l'intera stringa di input e internamente controlliamo la congruenza tra il prefisso e il suffisso precedentemente calcolato aumentando la lunghezza del bordo minimo man mano che il ciclo e il controllo avanzano.

## 3.2 periodSmart

Questo algoritmo invece, sfrutta il concetto di bordo, cioè una qualsiasi stringa che sia, prefisso e suffisso proprio della stringa principale. Come osservato quindi  $p$  è un periodo frazionario di  $s$  solo se  $p = |s| - r$  dove  $r$  è la lunghezza di un bordo di  $s$ , quindi il problema si riduce al solo calcolo del bordo massimo della stringa  $s$ .

Andiamo quindi ad analizzare l'implementazione vera e propria del algoritmo in questione, avremo quindi

## 3.3 Algoritmo per il calcolo dei tempi medi

Il calcolo dei tempi medi

# 4 Analisi dei dati ottenuti

## 4.1 Grafico dei tempi di periodNaive

Allegare naive.pdf trasformato in immagine (ingrandire le scritte)

Allegare naivedist.pdf trasformato in immagine (ingrandire le scritte)

## **4.2 Grafico dei tempi di periodSmart**

Allegare smart.pdf trasformato in immagine

Allegare smartdist.pdf trasformato in immagine

## **4.3 Analisi logaritmica dei due algoritmi**

Allegare smartnaive.pdf trasformato in immagine

Non potendo analizzare i due algoritmi essendo su due scale temporali completamente differenti, dobbiamo quindi analizzarli in una scala logaritmica:

Allegare smartlog.pdf trasformato in immagine

Allegare naivelog.pdf trasformato in immagine

## **4.4 Analisi comparativa tra i due algoritmi**

Allegare smart.pdf trasformato in immagine

Allegare naivedist.pdf trasformato in immagine

# **5 Conclusioni**

Come osservato dai grafici, abbiamo sicuramente constatato che tra i due algoritmi c'è una notevole differenza in termini di tempo. L'algoritmo smart impiega molto meno tempo dell'algoritmo rivale ed è quindi più consigliato il suo utilizzo.