

# Introduzione

Negli ultimi tempi, nel campo della bio-informatica, il *DNA computing* ha suscitato un grande interesse. A differenza delle attuali tecnologie basate sul silicio, alla base del DNA computing ci sono, come suggerisce il nome, molecole di DNA, quindi materiale organico. Uno dei vantaggi maggiori che l'utilizzo di sistemi basati sul DNA può offrire è proprio la capacità di poter eseguire un lavoro *in parallelo*. Per esempio, un'operazione di ricerca, invece di essere effettuata un elemento per volta, verrebbe effettuata analizzando tutti gli elementi simultaneamente. Per svolgere questi lavori, si possono effettuare alcune operazioni su una o più molecole di DNA, come, per esempio, tagliare e ricongiungere dei filamenti attraverso degli enzimi chiamati, rispettivamente, *endonucleasi* e *ligasi*.

Il lavoro svolto e documentato in questa tesi è nato avendo come motivazione e come traguardo la simulazione di un fenomeno biologico che utilizza le operazioni di “taglio” e “unione” tra filamenti di DNA: lo splicing. Nel 1987, Tom Head tentò per primo, mediante la teoria dei linguaggi formali, di definire alcuni metodi per analizzare questo fenomeno di ricombinazione e, dopo di lui, molti altri ricercatori sono stati in grado di fornire definizioni e proprietà sullo splicing.

Considerando quindi un linguaggio iniziale  $I$  associato alle molecole di

DNA è un set di regole  $R$ , cioè un linguaggio che formalizza il comportamento ricombinante consentito dagli enzimi poc'anzi nominati, è possibile generare un sistema splicing. Questo sistema, che itera le regole  $R$  sul linguaggio  $I$ , viene chiamato linguaggio splicing. Nello specifico, visto che il DNA può assumere anche forma circolare, si può parlare di sistemi splicing circolari, sistemi aventi come linguaggio iniziale il linguaggio definito dalle parole circolari, definite nel capitolo 1. Quindi, un sistema splicing circolare non è nient'altro che una tripla  $(A, I, R)$ , dove  $A$  è un alfabeto,  $I$  è un linguaggio circolare iniziale e  $R$  è un insieme di regole di splicing.

Il pacchetto software creato durante il periodo di tirocinio, è stato implementato tenendo conto della definizione di sistema splicing circolare appena descritta. È stato utilizzato come ambiente di sviluppo Eclipse Europa e come linguaggio di programmazione Java. Nello specifico, il pacchetto contiene le classi che rappresentano le parole (circolare, lineare e vuota) ed una classe atta a rappresentare una struttura dati capace di contenere parole (qualunque esse siano) eliminandone i duplicati. Anche se la motivazione principale di questo lavoro è lo splicing circolare, ciò non toglie la possibilità di utilizzare questo pacchetto software per altri scopi. Per esempio, è possibile dimostrare eventuali proposizioni o nuove proprietà di un linguaggio circolare, approfittando appunto della velocità di calcolo di un computer.

Partendo dall'idea che una parola è senza dubbio una stringa, è stata creata un'interfaccia che rappresentasse tutte le parole. Dopodiché sono state implementate la parola circolare e la parola lineare avendo come base una stringa di tipo `String`. È stata creata anche una classe apposita per la parola vuota, però, in questo caso, è stato opportuno marcare tale parola con un carattere speciale.

Per quanto riguarda la collezione di parole, come base è stata utilizzata una `LinkedList`.

Questo documento è organizzato nel modo seguente: nel primo capitolo c'è una breve descrizione di cos'è il DNA computing e qual è la sua importanza, spiegando cos'è lo splicing e quindi gli obiettivi di questo documento. A seguire, nel secondo capitolo, vengono fornite alcune definizioni sulle parole e sulle operazioni che possono essere eseguite su esse. Il terzo capitolo è dedicato al fenomeno biologico dello splicing ed alla sua formalizzazione in un linguaggio, mentre il quarto capitolo ed il quinto capitolo mostreranno l'implementazione delle classi del pacchetto `circularWord`.