

BAB 6

SIMPULAN DAN SARAN

6.1 Simpulan

Simpulan yang dapat diambil dari hasil pembuatan aplikasi Bioinformatika ini adalah:

1. Aplikasi Bioinformatika dapat memproses *alignment* dari dua buah sekuens DNA yang dipilih dengan algoritma Needleman-Wunsch dan algoritma Lempel-Ziv.
2. Algoritma Needleman-Wunsch memiliki keunggulan dalam *alignment* sekuens dengan panjang karakter kurang dari 1000, Needleman-Wunsch memiliki waktu proses dengan nilai minimal 1 *miliseconds* dan nilai maksimal 75 *miliseconds*. sementara algoritma Lempel-Ziv memiliki waktu kinerja yang lebih lama dengan waktu proses minimal 25 *miliseconds* dan maksimal 960 *miliseconds* dalam penelitian untuk karakter dengan panjang maksimal 1000, namun Lempel-Ziv memiliki keunggulan dalam *alignment* untuk sekuens dengan faktorisasi sempurna. Faktorisasi sempurna dapat dilihat dari frase yang terurut secara bertahap dengan pengulangan dari tiap frase yang sudah tercatat di atasnya dengan tambahan satu karakter selanjutnya dari tiap frase. Needleman-Wunsch dapat digunakan untuk pensejajaran sekuens pada umumnya namun Lempel-Ziv memiliki keunggulan dalam proses pengolahan *dataset* dengan faktorisasi sempurna.
3. Algoritma Needleman-Wunsch dan Lempel-Ziv dapat memberikan hasil *alignment* dengan visualisasi *gap* yang dapat dilihat jelas oleh pengguna, sehingga hasil *alignment* tersebut dapat digunakan untuk penelitian dalam bidang biologi molekuler untuk mencari tanda mutasi (*mutation mark*) dalam suatu kasus pensejajaran DNA.
4. Lempel-Ziv *factorisation phrase* memiliki keunggulan hanya dalam kasus sekuens dengan *perfect factorisation phrase*. Namun proses *factorisation phrase* tidak mengubah nilai *gap* terhadap hasil *alignment* sehingga data hasil pengujian tetap valid.

6.2 Saran

Saran pengembangan aplikasi Bioinformatika yang dapat diberikan adalah sebagai berikut:

1. Memberikan batasan karakter maksimal untuk pengolahan sekuens yang tidak melebihi kapasitas memori komputer/laptop yang digunakan, karena untuk saat ini, proses *alignment* dengan karakter yang melebihi 8000 akan mengalami masalah kelebihan pemakaian memori.
2. Memberikan *suggestion* terhadap algoritma Lempel-Ziv yang memiliki keunggulan dalam pemrosesan data untuk sekuens yang memiliki faktorisasi sempurna. *Suggestion* dapat berupa *disclaimer* yang terdapat di aplikasi, sehingga pengguna mengerti bagaimana tiap algoritma menangani sekuens yang diberikan.
3. Memisahkan data sekuens untuk Lempel-Ziv yang sudah mengalami *factorisation phrase* sehingga perhitungan untuk proses Lempel-Ziv diharapkan memiliki waktu yang tidak jauh berbeda dengan Needleman-Wunsch.