

BAB 6.

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berikut ini adalah kesimpulan yang didapatkan dari pengerjaan tugas akhir:

1. Perancangan aplikasi kategorisasi bahasa C dapat dilakukan dengan baik melalui penyusunan laporan tugas akhir ini, terbukti dengan adanya bab 3 sebagai analisis yang di kerjakan oleh penulis sebelum membuat aplikasi sehingga penulis dimudahkan dalam pembuatan aplikasi.
2. Aplikasi kategorisasi bahasa C ini dapat memudahkan dosen pengajar dalam mengelompokkan hasil kerja dari mahasiswa karena aplikasi akan menampilkan nrp-nrp mahasiswa mana yang tergolong dalam satu kelompok, sehingga dosen pengajar tidak perlu melakukan secara manual.
3. Aplikasi kategorisasi bahasa C ini dapat memudahkan dosen pengajar dalam menemukan berapa banyak kemungkinan yang dapat digunakan untuk menyelesaikan kasus yang diberikan karena aplikasi akan menampilkan jumlah kelompok yang terbentuk saat proses analisis dilakukan.
4. Aplikasi kategorisasi bahasa C ini dapat memudahkan dosen pengajar untuk melihat hasil kerja mahasiswa dengan kelas yang berbeda untuk menjadikan perbandingan kelas yang memiliki kreatifitas dalam mencari solusi kasus yang diberikan karena aplikasi dapat menampilkan hasil analisis untuk setiap kasus yang diberikan pada kelas yang berbeda.
5. Aplikasi kategorisasi bahasa C ini dapat memudahkan dosen pengajar untuk melihat proses kerja mahasiswa dalam mencari solusi kasus yang diberikan karena aplikasi dapat menampilkan hasil analisis dari setiap mahasiswa dalam kasus yang berbeda-beda.

6. Penerapan algoritma *Running Karp-Rabin Greedy String Tiling* dapat dilakukan dalam aplikasi sebagai basis pengukuran similaritas.
7. Kelebihan yang dimiliki oleh algoritma *Running Karp-Rabin Greedy String Tiling* dibandingkan dengan algoritma *string matching* yang lain adalah tidak bergantung pada urutan *substring* yang dicari dalam *string* teks yang dibandingkan. Algoritma *Running Karp-Rabin Greedy String Tiling* juga selalu mengutamakan *substring* identik yang lebih panjang, sehingga dapat lebih tepat menemukan kesamaan dalam sebuah *string* teks karena *substring* yang panjang lebih memungkinkan untuk mendeteksi bahwa kedua *string* yang dibandingkan memiliki kemiripan yang tinggi. Hal tersebut terbukti pada pembahasan subbab 3.3.1.5.
8. Kelemahan algoritma *Running Karp-Rabin Greedy String Tiling* yang ditemukan adalah penentuan panjang *minimal-search-length* yang merupakan sebuah parameter algoritma *Running Karp-Rabin Greedy String Tiling*. Semakin kecil nilai *minimum-search-length* yang ditentukan, maka kemungkinan sensitivitas deteksi kemiripan akan semakin tinggi, namun pada kenyataannya hal tersebut belum tentu merupakan sebuah pola yang memiliki arti yang sama. Hal tersebut terbukti pada pembahasan subbab 3.3.1.5.
9. Aplikasi kategorisasi bahasa C dapat memudahkan pengajar menganalisis hasil praktikum yang sudah dikerjakan oleh mahasiswa, karena aplikasi menyimpan data yang disimpan oleh pengguna setelah melakukan analisis. Analisis yang dilakukan pada aplikasi adalah melihat banyaknya kelompok yang terbentuk pada kelas tertentu, melihat banyaknya kelompok yang terbentuk pada mahasiswa tertentu, melihat banyaknya kelompok yang terbentuk pada kelas tertentu dan kasus tertentu, melihat banyaknya kelompok yang terbentuk pada topik tertentu, melihat banyaknya kelompok yang terbentuk pada topik tertentu dan kelas tertentu.
10. Pengelompokan hasil kerja mahasiswa satu dengan mahasiswa yang lain, dengan melihat data-data similaritas mahasiswa tersebut dengan mahasiswa yang lain.

11. Banyaknya kelompok yang akan terbentuk bergantung kepada banyaknya sampel *file* yang digunakan, kompleksitas soal yang diberikan dan kreatifitas mahasiswa. Semakin banyak sampel *file*, semakin komplek soal yang diberikan dan semakin tinggi tingkat kreatifitas mahasiswa dalam menjawab kasus yang diberikan, maka akan menghasilkan kelompok yang lebih beragam. Hal tersebut dibuktikan dengan pengujian yang dilakukan pada delapan kasus, tigapuluh sampel *file*, dan satu sampel kelas. Hasil pengujian dapat dilihat pada subbab 5.2.2.

6.2 Saran

Berikut ini saran yang didapat dengan pengerjaan tugas akhir :

1. Perlu adanya penanganan *syntax-syntax* yang lebih lengkap dari bahasa C seperti eksponensial, pointer dan lain sebagainya.
2. Perlu dilakukannya konversi potongan kode program yang lebih luas agar perbandingan similaritas lebih maksimal, seperti konversi potongan kode program untuk kasus rekursif.
3. Hasil analisis yang ditampilkan dengan menggunakan gambar diagram atau pohon agar pengguna lebih mudah memahami hasil analisis yang dihasilkan.