

BAB I

PENDAHULUAN

1.1 Latar belakang

Dalam program komputer objek dapat dibentuk menggunakan kelas dan memiliki atribut atau fitur untuk mendeskripsikan objek tersebut. Himpunan fitur suatu objek dapat dimiliki objek lain melalui relasi *proper superset* membentuk relasi objek komposisi. Relasi yang dibentuk antar objek ini dapat dipetakan menggunakan *graph*. Hasilnya dapat mengelompokkan objek yang membentuk *poset*. Kelompok objek yang satu juga dapat memiliki relasi dengan yang lainnya membentuk komponen dalam *graph*.

Penelitian yang telah dilakukan oleh *Soetrisno* (2016) berhasil membentuk relasi objek komposisi dengan *adjacency hyperedge matrix*. Metode tersebut menggunakan *hypergraph*, dimana setiap *edge*-nya dapat menghubungkan lebih dari dua *vertex*. Hasilnya membentuk *transitive closure* dan objek yang sama atau identik berdasarkan fitur juga dapat diidentifikasi. Setelah itu sifat *poset* yang dibentuk disederhanakan menjadi *hasse diagram* atau *minimum spanning subgraph* yang dilanjutkan dengan pembentukan komponen dalam *graph*.

Pembentukan *hasse diagram* dan identifikasi objek identik dapat dilakukan satu per satu untuk setiap objek menggunakan DFS. Dalam hal ini DFS digunakan untuk melakukan *searching* dalam proses konstruksi *graph* bukan *searching* pada *graph* yang sudah ada. Konstruksi *graph* yang dilakukan satu per satu, membuat posisi objek dan relasi yang dibentuk dapat berubah, karena itu metode

pembentukan *graph* ini disebut dengan *growing*. *Hasse diagram* dapat membentuk *transitive closure* dengan algoritma *Warshall* atau DFS. Pembentukan *hasse diagram* dapat mengkoneksikan semua objek (*connected graph*) atau membentuk beberapa komponen dengan DFS.

Identifikasi relasi objek komposisi dibentuk menggunakan bahasa pemrograman C++ dengan *library* STL. Algoritma dari STL dapat digunakan untuk membantu proses identifikasi relasi objek komposisi dengan *superset*. Menurut Winadi (2017), proses *sorting* dengan STL juga lebih cepat dan mudah untuk dilakukan dalam identifikasi relasi objek komposisi.

Dalam penelitian ini identifikasi yang dilakukan langsung membentuk *hasse diagram* yang dilanjutkan dengan pembentukan *transitive closure*. Setelah itu komponen dalam *graph* akan diidentifikasi. Pengujian terhadap setiap tahapan tersebut dilakukan dengan *white box testing* menggunakan *test case* yang telah ditetapkan.

1.2 Perumusan masalah

Bagian ini berisi perumusan masalah yang berdasarkan latar belakang yang telah dijabarkan pada bagian sebelumnya.

- 1) Bagaimana metode *growing* dapat diterapkan untuk membentuk relasi objek komposisi dalam bentuk *hasse diagram*?
- 2) Apakah DFS merupakan algoritma yang dapat diterapkan dalam metode *growing*?
- 3) Apakah pembentukan *transitive closure* dalam relasi objek komposisi dengan algoritma DFS lebih baik dibandingkan algoritma *Warshall* ?

- 4) Bagaimana proses identifikasi relasi objek komposisi dengan algoritma dari STL dapat dilakukan?

1.3 Batasan masalah

Dalam penelitian ini, diperlukan beberapa batasan yang digunakan sebagai acuan dalam pelaksanaan penelitian untuk memberikan arah yang jelas dalam pengembangannya. Batasan-batasan yang terdapat dalam penelitian adalah sebagai berikut:

- 1) Perancangan program menggunakan bahasa pemrograman C++ dengan STL.
- 2) Implementasi dan pengujian dilakukan terhadap metode dan algoritma yang sudah ada.
- 3) Setiap objek dalam *input data* harus memiliki himpunan ciri berupa pasangan yaitu *feature structure* dan *feature value*.
- 4) Program mengidentifikasi relasi objek komposisi berdasarkan *feature value* yang dilanjutkan dengan *feature structure*.

1.4 Tujuan

Tujuan dari penelitian ini adalah membuat program dengan bahasa C++ menggunakan STL untuk identifikasi relasi objek komposisi melalui konstruksi *graph poset* dalam bentuk *hasse diagram* dengan metode *growing* dan *transitive closure*. Pembentukan *hasse diagram* dilakukan dengan metode *growing* dan *transitive closure* dengan algoritma *Warshall* dan *DFS*. Setelah itu komponen dalam *graph* dapat diidentifikasi.

1.5 Metodologi

Untuk menyelesaikan masalah pada penelitian ini, metode-metode yang digunakan adalah:

- 1) Melakukan studi pustaka mengenai konstruksi *graph* dan beberapa algoritma yang diperlukan dari internet, buku, artikel, dan jurnal.
- 2) Membangun sebuah program untuk melakukan identifikasi relasi objek komposisi dengan STL pada C++.
- 3) Melakukan pengujian program dengan *white box testing* untuk menguji program

1.6 Sistematika penulisan

Laporan tugas akhir ini disusun menggunakan sistematika penulisan yang dapat dijelaskan sebagai berikut :

BAB I PENDAHULUAN

Bab satu berisi penjelasan mengenai latar belakang pemilihan topik yang dibahas pada tugas akhir. Dilanjutkan dengan rumusan dan batasan masalah topik, diikuti juga dengan tujuan dan metodologi yang digunakan. Pada akhir bab dijelaskan pula mengenai sistematika penulisan yang digunakan.

BAB II LANDASAN TEORI

Bab kedua menjelaskan tentang teori yang digunakan dalam penyusunan tugas akhir, yaitu mencakup teori tentang *set*, *graph*, *hypergraph*, relasi objek komposisi, *depth first search*, *Standard Template Library*, *poset*, *hasse diagram*, *Warshall Algorithm*, dan lain-lain.

BAB III PERANCANGAN SISTEM

Bab ketiga berisi tentang penjelasan mengenai rancangan sistem dari aplikasi yang dibuat sesuai dengan teori dan algoritma yang ada dalam membentuk *graph* menggunakan STL pada bahasa pemrograman C++.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Bab ini berisikan penjelasan mengenai implementasi aplikasi yang dirancang, serta penjelasan singkat mengenai cara menjalankan

aplikasi berdasarkan data yang ditentukan. Hasil dari pengujian sistem juga dijelaskan pada bab ini.

BAB V KESIMPULAN DAN SARAN

Bab lima ini merupakan bab terakhir yang berisikan kesimpulan yang diperoleh dari penelitian yang telah dilakukan serta saran untuk pengembangan lebih lanjut