

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Struktur data merupakan komponen yang mempunyai peranan penting dalam teknik pemrograman. Untuk meningkatkan efisiensi dan efektifitas dalam program yang dibuat diperlukan pemilihan struktur data yang tepat dan akurat. Salah satu topik yang dibahas dalam struktur data adalah masalah pencarian data (*searching*), pencarian data adalah suatu proses untuk mengumpulkan sejumlah informasi didalam memori komputer dan mencari kembali informasi yang diperlukan secepat mungkin.

Algoritma pencarian (*searching algorithm*) adalah algoritma yang menerima argumen K (sebagai kunci) dan dengan langkah-langkah tertentu akan mencari rekaman yang kuncinya bernilai K. Metode yang dapat digunakan tergantung pada model struktur data tersebut. Sehingga berawal dari model struktur data tersebut terdapat bermacam-macam metode pencarian, antara lain pencarian berurutan, pencarian biner, pencarian pada tabel yang sudah diurutkan, pohon telusur dan hashing.

Pohon telusur (*search tree*) mempunyai karakteristik dan tingkat kesulitan tersendiri dalam proses pencariannya. Hal ini lebih banyak disebabkan oleh bentuk atau model penyimpanan struktur datanya yang berupa pohon. Dalam proses pencarian bentuk pohon akan membawa efek terhadap efektifitas dan efisiensi dalam mendapatkan data, hal ini salah satunya disebabkan ketidak-tepatan peletakan komponen-komponennya dalam formasi pohon. Dengan kata

lain, jika ditinjau dari kompleksitas waktunya (*time complexity*), proses pencarian lebih cepat bilamana komponen-komponennya berada pada posisi yang tepat dibanding sebaliknya. Peletakan informasi pada pohon tergantung pada metode yang digunakan. Dalam tugas akhir ini, akan dibahas Algoritma pohon Huffman yaitu pohon telusur yang meletakkan komponen-komponennya berdasarkan frekuensi kemunculan data, sehingga mempunyai panjang jalur luar terbebani (*weighted external path length*) minimum yang disebut *Pohon Huffman*.

### 1.2. Perumusan Masalah

Dalam penulisan tugas akhir ini algoritma Pohon Huffman akan digunakan untuk menentukan kode biner setiap karakter dari suatu teks, sehingga permasalahan yang akan diangkat adalah bagaimana menentukan kode biner untuk mencari panjang lintasan minimum dengan algoritma pohon Huffman dan bentuk implementasinya dalam pemrograman bahasa Turbo Pascal 7.0.

### 1.3. Pembatasan Masalah

Pada penulisan tugas akhir dengan judul “ Implementasi Algoritma Pohon Huffman untuk Menentukan Kode Biner ” ini dibatasi pada cara kerja algoritma pohon Huffman dalam menentukan kode biner tiap karakter dari suatu teks.

#### 1.4. Tujuan Penulisan

Adapun tujuan dalam penulisan tugas akhir ini adalah sebagai berikut :

1. Memahami algoritma pohon *Huffman* untuk menentukan kode biner setiap karakter suatu teks.
2. Mengimplementasikan algoritma pohon Huffman untuk menentukan kode biner tiap karakter dari suatu teks dalam pemrograman bahasa Turbo Pascal 7.0.

#### 1.5. Sistematika Penulisan

Pada penulisan Tugas Akhir ini dikelompokkan menjadi empat bagian, mulai dari bab I sampai bab IV. Bab I merupakan pendahuluan yang meliputi latar belakang masalah, perumusan masalah, pembatasan masalah, tujuan penulisan dan sistematika penulisan.

Bab II adalah materi penunjang yang berisi beberapa materi yang digunakan sebagai pendukung materi pembahasan masalah pada bab berikutnya. Bab II ini membicarakan tentang konsep dasar himpunan, struktur pohon, algoritma, serta kompleksitas waktu yang terkait dengan algoritma pohon Huffman.

Pembahasan tentang proses pembentukan, cara kerja dan pengkodean dengan algoritma pohon Huffman diterangkan dalam bab III.

Terakhir bab IV berisi kesimpulan dari seluruh bab sebelumnya.