**JURUSAN INFORMATIKA**

**FAKULTAS TEKNOLOGI ELEKTRO DAN**

**INFORMATIKA CERDAS**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# IDENTITAS PENGUSUL

**NAMA : David Laksmana**

**NRP : 05111740000045**

**DOSEN WALI : Abdul Munif S.Kom, M.Sc.Eng**

**DOSEN PEMBIMBING : 1. Rully Sulaiman S.Kom,M.Kom  
 2. Misbakhul Munir Irfan Subakti, S.Kom,M.Sc**

# JUDUL TUGAS AKHIR

“Perbandingan *Assignment Algorithm* untuk Menyelesaikan Masalah Penugasan: Studi Kasus SPOJ 9971 *Adventurous Chess Master*”

# LATAR BELAKANG

Setiap pekerjaan dapat dilakukan pembagian menjadi tugas tugas yang lebih kecil. Setiap tugas dapat dilakukan oleh pekerja-pekerja yang ada dengan kecepatan yang berbeda. Karena kecepatan pengerjaan yang berbeda maka dapat dilakukan pencarian pasangan penugasan yang dapat mengurangi waktu dan biaya pengerjaan dari pekerjaan yang ada. Hubungan antara pekerja dan pekerjaan dapat dimodelkan dalam matriks relasi dengan nilai pada matriks berupa biaya jika pekerja mengerjakan pekerjaan yang bersesuaian.

Dalam penyelesaian masalah penugasan algoritma *Hungarian* (H.W.Kuhn, 1955)sering digunakan karena memiliki waktu polinomial. Selain *Hungarian* permasalahan ini dapat diselesaikan menggunakan pendekatan *linear programming* (Ramshaw & Tarjan, 2012) dan *Auction algorithm* (Bertsekas, 1979). Algoritma *Hungarian* mengerjakan pemasangan menggunakan sudut pandang keseluruhan sedangkan *Auction* menggunakan sudut pandang mulai dari sebagian permasalahan. Dalam penyelesaiannya masalah penugasan tidak memungkinkan untuk diselesaikan dengan cara mencoba setiap kombinasi pasangan karena akan terdapat n! pasangan yang harus dilakukan pengecekan.

Topik tugas akhir ini mengacu pada permasalahan *online judge* SPOJ dengan kode ACHESS. Permasalahan ini menggambarkan tentang sebuah papan catur dimana setiap bidak dapat berjalan dengan aturan khusus. Pada papan catur juga diberi beberapa lokasi tujuan yang semaksimal mungkin harus ditempati oleh satu bidak yang ada. Sehingga menghasilkan daftar tugas yang mungkin dilakukan oleh setiap bidak yang harus dioptimasikan untuk menghasilkan jumlah langkah yang minimum.

Dibandingkan dengan permasalahan lain ACHESS memiliki rentang biaya untuk menuju tujuan yang rendah(1 hingga 7) dan belum tentu semua bidak dapat menempati setiap tujuan juga jumlah tujuan dan bidak belum tentu sama sehingga graf bipartit yang dihasilkan tidak penuh dan tidak seimbang. Dibandingkan permasalahan dengan kode ASSIGN4 (Tu, 2006) pada SPOJ kasus ACHESS memiliki rentang nilai yang lebih kecil dan belum tentu seimbang.

Tugas akhir ini diharapkan dapat mempermudah pemilihan algoritma dari permasalahan penugasan secara optimal dan diharapkan dapat memberi kontribusi dalam perkembangan ilmu pengetahuan dan teknologi informasi.

# RUMUSAN MASALAH

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana cara merepresentasikan masukan permasalahan SPOJ *Adventurous Chess Master* dalam matriks relasi?
2. Bagaimana cara analisis dan desain algoritma penugasan untuk menyelesaikan permasalahan pada kasus SPOJ *Adventurous Chess Master?*
3. Bagaimana cara implementasi algoritma penugasan *hungarian* untuk menyelesaikan permasalahan pada kasus SPOJ *Adventurous Chess Master?*
4. Bagaimana cara implementasi algoritma penugasan menggunakan *linear programming* untuk menyelesaikan permasalahan pada kasus SPOJ *Adventurous Chess Master?*
5. Bagaimana cara implementasi algoritma *Auction* untuk menyelesaikan permasalahan pada kasus SPOJ *Adventurous Chess Master?*

# BATASAN MASALAH

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan antara lain:

1. Implementasi algoritma menggunakan bahasa pemrograman C++.
2. Batas maksimum lokasi tujuan 64.
3. Ukuran papan 8\*8.
4. Batas jumlah bidak 64.
5. Batas maksimum waktu perangkat lunak berjalan adalah 1 detik.
6. Batas memory perangkat lunak adalah 1536 MB.
7. Batas ukuran program adalah 50 kB.
8. Dataset yang digunakan adalah dataset pada permasalahan SPOJ *Adventurous Chess Master.*

# TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dari pembuatan tugas akhir ini antara lain:

1. Melakukan analisis dan desain algoritma penugasan untuk menyelesaikan permasalahan pada kasus SPOJ Adventurous Chess Master.
2. Melakukan implementasi algoritma penugasan untuk menyelesaikan permasalahan pada kasus SPOJ Adventurous Chess Master.
3. Melakukan perbandingan performa algoritma *Hungarian*, algoritma *Auction* dan algoritma menggunakan *linear* *programming* pada kasus SPOJ *Adventurous Chess Master.*

# MANFAAT TUGAS AKHIR

Tugas akhir ini diharapkan dapat membantu dalam pemilihan algoritma penugasan dan implementasinya dalam menyelesaikan masalah penugasan yang diberikan. Dimana memiliki rentang nilai biaya yang tidak besar.

# TINJAUAN PUSTAKA

**8.1. hungarian algorithm**

sebuah algoritma yang diguanakan dalam optimasi kombinatorik dalam waktu polynomial. Algoritma hungarian dapat bekerja dengan kompleksitas waktu sebesar O(n3).

Algoritma ini digunakan dengan input matriks n\*n dimana merepresentasikan hubungan antara bidak ke i dengan tujuan ke j.

**8.2. *linear programming***

sebuah cara untuk melakukan optimisasi pada fungsi linear pada pertidaksamaan linear atau batasan linear. Permasalahan penugasan dapat dilakukan perubahan dengan *flow network* dan akan menghasilkan *batasan-batasan* yang dapat digunakan sebagai perhitungan dalam *linear programming* (Ramshaw & Tarjan, 2012).

**8.3. *flow network***

sebuah graf terarah dimana setiap *edge* memiliki kapasitas, dan setiap *edge* dapat memiliki sebuah aliran. Permasalahan penugasan dapat dimodelkan dalam permasalahan *flow network* yaitu dengan menambahkan *vertex* sumber dengan *edge* menuju setiap bidak dengan bobot 0, *vertex* akhir yang menghubungkan setiap *edge* tujuan dan *edge* pada setiap hubungan bidak dengan bobot 0 dan tujuan dengan bobot sebesar biaya menuju tujuan dan setiap kapasitas *edge* dalam graf sebesar satu.

**8.4. *Hopcroft-Karp***

Sebuah algoritma dimana mencari pasangan edge dimana setiap vertex yang dipasangkan unik sebanyak mungkin. Permasalahan penugasan merupakan salah satu turunan dari permasalahan *maximum bipartite matching* dimana setiap vertex pada graph kiri maupun kanan harus dipasangkan antar graf semaksimal mungkin dan dicari biaya yang paling minimum. *Hopcroft-Karp* dapat menyelesaikan *maximum bipartite matching* dalam O().

**8.5. *Auction Algorithm***

Sebuah cara dimana setiap tujuan akan dilakukan penawaran oleh node sumber. Penawaran tujuan oleh node sumber akan mengakibatkan biaya untuk menuju tujuan tersebut menjadi naik sesuai nilai yang ditentukan sehingga mengakibatkan berkurangnya penawar pada tujuan tersebut dan membuat penyelesaian semakin konvergen (Bertsekas, 1979).

**8.6. Studi Kasus SPOJ 9971 *Adventurous Chess Master***

Permasalahan SPOJ *Adventurous Chess Master* membahas mengenai sebuah papan catur dengan beberapa bidak dan tempat tujuan dalam papan. Setiap tempat tujuan harus semaksimal mungkin ditempati oleh bidak yang tersedia dengan melakukan langkah sesuai dengan aturan:

* pion/*pawn* dapat berjalan maju atau mundur satu petak
  + pion jika berada pada baris pertama atau terakhir dapat memiliki biaya maksimum sebesar 7
  + selain pada kolom yang ditempati pion tidak dapat bergerak menuju posisi tersebut
* raja*/king* dapat berjalan ke delapan arah sebanyak satu petak
  + raja jika berada pada pinggir atau ujung sebuah papan dapat memiliki biaya maksimum sebesar 7
* ratu/*queen* dapat berjalan ke 8 arah hingga batas papan
  + ratu pada sembarang posisi dapat melakukan langkah menuju semua petak dengan biaya maksimum 2
* kuda/*knight* dapat berjalan 2 petak ke 4 petak yang bersebelahan kemudian dilanjut 1 petak ke arah yang tegak lurus dengan arah gerak pertama
  + kuda dapat bergerak menuju setiap petak. Jika berada disebuah ujung papan biaya maksimum dapat mencapai 6 langkah
* gajah/*bishop* dapat berjalan ke 4 arah diagonal hingga ujung papan
  + gajah dapat bergerak pada petak dengan warna yang sama dengan maksimum langkah 2 untuk petak terjauh
  + gajah tidak dapat bergerak menuju petak yang memiliki warna yang berbeda((xbidak+ ybidak)mod 2 ! = (xtujuan+ ytujuan)mod 2)
* benteng/*rook* dapat berjalan ke 4 arah yang bersebelahan hingga ujung papan
  + benteng dapat bergerak ke setiap petak dengan langkah maksimum sebesar 2
* setiap bidak dapat berada dalam petak yang sama dan dapat bergerak melewati bidak lain

Dari semua langkah yang ada permasalahan ini meminta jumlah langkah paling sedikit untuk melakukan sesuai dengan permintaan.

Masukan yang diberikan adalah P jumlah bidak yang ada dan L jumlah lokasi yang ada. Kemudian diikuti dengan koordinat bidak dan jenis bidak sebanyak P buah kemudian diikuti dengan koordinat lokasi yang harus ditempati.

Berikut adalah contoh dari masukan dan keluaran yang diharapkan dengan studi kasus:

* + - 1. terdapat buah pion pada posisi (1,1) dan 1 tujuan pada posisi (8,1) maka menghasilkan keluaran 7
      2. terdapat 3 buah bidak raja (2,8), ratu (2,8), dan gajah (7,5) dan 5 tujuan pada posisi (1,1), (2,2), (3,6), (6,3) dan (4,4) maka menghasilkan keluaran 5

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |  | Penjelasan pada tabel 8.5.1 |
| 1 | P |  |  |  |  |  |  |  |  | P : *Pion* |
| 2 |  |  |  |  |  |  |  |  |  | pion bergerak sebanyak 7 langkah menuju (1,8) |
| 3 |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |
| 8 | x |  |  |  |  |  |  |  |  |  |

Tabel 8.5.1 penggambaran kasus 1

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |  | Penjelasan pada tabel 8.5.2 |
| 1 | x |  |  |  |  |  |  |  |  | K: Raja |
| 2 |  | x |  |  |  |  |  |  |  | Q: Ratu |
| 3 |  |  |  |  |  | x |  |  |  | B: Gajah |
| 4 |  |  |  | x |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  | B |  |  | Ratu bergerak 1 langkah menuju (2,2) |
| 6 |  |  | x |  |  |  |  |  |  | Raja bergerak 2 langkah menuju (3,6) |
| 7 |  |  |  |  |  |  |  |  |  | Gajah bergerak 2 langkah menuju (4,4) |
| 8 |  | K,Q |  |  |  |  |  |  |  | Jumlah langkah minimal 5 |

Tabel 8.5.2 Penggambaran kasus 2

# RINGKASAN ISI TUGAS AKHIR

Permasalahan diatas memiliki beberapa solusi yang mungkin dari pasangan lokasi dan bidak yang menempati. Jika semua bidak dapat menempati setiap lokasi yang ada maka jumlah kemungkinan yang ada adalah P! jika P=L, P!/(P-L)! jika P>L dan P!\* jika P<L.

Pada bagian ini dilakukan pembahasa masukan dan keluaran pada kasus kedua.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | (1,1) | (2,2) | (3,6) | (6,3) | (4,4) |
| K | 7 | 6 | 2 | 5 | 4 |
| Q | 2 | 1 | 2 | 2 | 2 |
| B | 2 | 2 | x | x | 2 |

Tabel jarak antara bidak dan tujuan

Pertama untuk perhitungan jumlah langkah yang diperlukan diperlukan menghitung jumlah langkah setiap bidak menuju setiap lokasi seperti pada tabel diatas. Kemudian dilakukan perhitungan hingga mendapatkan sesuai pada penggambaran kasus 2.

# METODOLOGI

## Penyusunan proposal tugas akhir

Proposal tugas akhir ini berisi tentang deskripsi pendahuluan dari tugas akhir yang akan dibuat. Pendahuluan ini terdiri atas hal yang menjadi latar belakang diajukannya usulan tugas akhir, rumusan masalah yang diangkat, batasan masalah untuk tugas akhir, tujuan dari pembuatan tugas akhir, dan manfaat dari hasil pembuatan tugas akhir. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung pembuatan tugas akhir. Sub bab metodologi berisi penjelasan mengenai tahapan penyusunan tugas akhir mulai dari penyusunan proposal hingga penyusunan buku tugas akhir. Terdapat pula sub bab jadwal kegiatan yang menjelaskan jadwal pengerjaan tugas akhir.

## Studi literatur

Tahap kedua adalah mencari informasi dan studi literatur yang relevan untuk dijadikan referensi dalam mengerjakan Tugas Akhir. Informasi dan studi literatur didapat dari buku, scientific paper, artikel internet, dan materi kuliah yang berhubungan.

## Implementasi perangkat lunak

Tahap implementasi meliputi implementasi algoritma pada perangkat lunak yang telah didukung oleh hasil analisis dan desain pada tahap sebelumnya. Implementasi akan dilakukan dengan menggunakan bahasa pemrograman C++. i dari hungarian akan dibangun menggunakan bahasa C++

## Pengujian dan evaluasi

Tahap pengujian dan evaluasi dilakukan dengan menggunakan dataset SPOJ “Adventurous Chess Master” untuk mengetahui hasil dan performa metode yang telah dibangun. Evaluasi dan perbaikan akan dilakukan hingga perangkat lunak yang diuji menghasilkan hasil performa yang baik.

## Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
   1. Latar Belakang
   2. Rumusan Masalah
   3. Batasan Tugas Akhir
   4. Tujuan
   5. Metodologi
   6. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

# JADWAL KEGIATAN

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tahapan | 2020 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| April | | | | Mei | | | | Juni | | | | Juli | | | | Agustus | | | | September | | | | Oktober | | | | November | | | | Desember | | | |
| Penyusunan Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Pengujian dan Evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Penyusunan Buku |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# DAFTAR PUSTAKA

[1] Bertsekas, D. P. (1979, March). *Orig\_Auction.pdf.* Retrieved from mit.edu: https://www.mit.edu/~dimitrib/Orig\_Auction.pdf

[2] Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Linear Programming. In T. H. Cormen, C. E. Leiserson, R. L. Rivest, & C. Stein, *Introduction to Algorithm* (pp. 843-887). London: The MIT Press.

[3] H.W.Kuhn. (1955). The Hungarian method for the assignment problem. *Naval Research Logistic, vol. 2*, 83–97.

[4] Ibrahim, M. S. (2011, 11 13). *Problem ACHESS*. Retrieved from spoj.com: https://www.spoj.com/problems/ACHESS/

[5] Ramshaw, L., & Tarjan, R. E. (2012). *On Minimum-Cost Assignments in Unbalanced Bipartite Graphs.* HP Laboratories .

[6] Tu, N. D. (2006, 1 2). *SPOJ.com - Problem ASSIGN4*. Retrieved from spoj.com: https://www.spoj.com/problems/ASSIGN4/#