

IMPLEMENTASI ALGORITMA *DIRECTED ACYCLIC WORD GRAPH* DALAM PERANCANGAN GAME SCRABBLE

Arie Lasaprima¹, Agung Toto Wibowo², ZK Abdurahman Baizal³

^{1,3}Program Studi Teknik Informatika, Fakultas Informatika, Institut Teknologi Telkom, Bandung

²Program Studi Ilmu Komputasi, Fakultas Sains, Institut Teknologi Telkom, Bandung

Jl Telekomunikasi No 1, Terusan Buah Batu, Bandung

Telp. (022)7564108

Email : ¹arie.la54k@gmail.com, ²atw@ittelkom.ac.id, ³bayzal@gmail.com

ABSTRAK

Menerapkan Artificial Intelligence (AI) pada komputer yang dapat melakukan proses perhitungan lebih cepat dibanding manusia dapat membuat sebuah permainan menjadi lebih menantang dan tidak membosankan. Salah satu platform yang bagus untuk menguji teknik AI pada game yaitu game scrabble.

AI scrabble yang selalu memilih kata dengan nilai tertinggi dari kemungkinan kata yang dapat disusun di setiap giliran akan dengan sangat mudah dikalahkan oleh pemain manusia yang ahli dalam scrabble. Game ini dirancang agar manusia dapat melawan AI yang yang tidak gampang dikalahkan sehingga permainan menjadi lebih menarik dan menantang. Oleh karena itu dirancang strategi permainan sehingga manusia tidak dapat dengan mudah mengalahkan AI system. AI system akan mengevaluasi pemilihan kata yang akan dimainkan pada setiap giliran. Setiap pilihan kata akan mempengaruhi baik papan permainan maupun kemungkinan pilihan kata pada giliran berikutnya. Pada penelitian ini, dalam membentuk kamus data untuk menyimpan semua kata yang valid digunakan algoritma DAWG. Algoritma DAWG dapat memperkecil ruang penyimpanan dan membantu proses mendapatkan kata yang akan dimainkan. Selain itu juga dilakukan analisis dari ketiga strategi permainan yang dibentuk, untuk melihat performansi masing-masing strategi.

Dari hasil implementasi dan pengujian yang dilakukan, strategi permainan scrabble yang memilih kata pada setiap giliran hanya berdasarkan score tertinggi mempunyai persentase kemenangan hanya 40%

Kata kunci : Game Scrabble, Algoritma directed acyclic word graph.

1. PENDAHULUAN

Seiring perkembangan jaman dan teknologi, perilaku manusia dalam berfikir sangat mungkin untuk diterapkan pada komputer khususnya pada game, penerapan perilaku manusia ini disebut artificial intelligence atau kecerdasan buatan, dengan menerapkan AI pada komputer yang dapat melakukan proses perhitungan lebih cepat dibanding manusia dapat membuat sebuah permainan menjadi lebih menantang dan tidak membosankan.

Salah satu platform yang bagus untuk menguji teknik AI pada game yaitu game *scrabble* (Sheppard, Brian, 2002). *Scrabble* adalah permainan kata dimana dua sampai empat pemain membentuk kata dari tujuh huruf yang ada pada mereka pada sebuah papan yang mempunyai luas 15x15 kotak. Permainan ini telah dijual di 121 negara sepanjang dunia dan lebih dari 100 juta set permainan telah terjual (Scrabble history, 2010), di Amerika tiap tiga rumah mempunyai satu set dari permainan ini. Pada akhir permainan yaitu pada saat huruf yang disediakan pada kantong telah habis, pemain yang mempunyai total jumlah nilai tertinggi akan menjadi pemenangnya.

AI *scrabble* yang selalu memilih kata dengan nilai tertinggi dari kemungkinan kata yang dapat disusun di setiap giliran akan dengan sangat mudah dikalahkan oleh pemain manusia yang ahli dalam *scrabble* (Sheppard, Brian, 2002), karena hal tersebut tidak menjamin pemain dapat mengalahkan

lawan atau memenangkan permainan. Saat ini algoritma yang akan diterapkan untuk membantu membangun Artificial Intelligence pada permainan kata ini adalah *directed acyclic word graph*, *directed acyclic word graph* (DAWG) adalah data struktur yang mempunyai efisiensi ruang dalam menangani dan menganalisa pengulangan pada sebuah text^[3,4] sehingga dalam penggunaannya DAWG membutuhkan memory yang sangat kecil. Penelusuran kata pada DAWG seperti pada sebuah trie atau pohon data dimana setiap awalan dan akhiran digunakan bersama-sama untuk memperkecil jumlah ruang yang diperlukan untuk menyimpan kosa kata (A. Appel and G. Jacobson, 1988) dan juga memperkecil waktu yang diperlukan untuk meretrieve kata.

2. ATURAN PERMAINAN DALAM PERMAINAN SCRABBLE

Permainan dimainkan dua sampai 4 orang, pada penelitian ini hanya akan ada 2 player, human player vs komputer player dan komputer player vs komputer player, player mengumpulkan nilai dengan meletakkan keping huruf pada papan permainan. Setiap huruf mempunyai point yang berbeda. Sehingga pemenang adalah player yang mendapatkan nilai tertinggi di akhir permainan.

Papan permainan berukuran 15x15 kotak, dimana kotak-kotak ini yang digunakan sebagai tempat meletakkan keping huruf. Beberapa kotak

dalam papan *Scrabble* mempunyai nilai tambah tersendiri sesuai dengan nama dan warnanya yaitu :

- *Double Letter Scores* - ketika keping huruf yang kita letakkan diletakkan pada kotak ini maka nilai keeping huruf dikalikan 2.
- *Triple Letter Score* - ketika keeping huruf yang kita letakkan diletakkan pada kotak ini maka nilai keeping huruf dikalikan 3.
- *Double Word Score* - ketika keeping huruf yang kita letakkan diletakkan pada kotak ini maka nilai kata atau jumlah dari nilai keeping huruf yang membentuk kata dikalikan 2.
- *Triple Word Score* - ketika keeping huruf yang kita letakkan diletakkan pada kotak ini maka nilai kata atau jumlah dari nilai keeping huruf yang membentuk kata dikalikan 3.
- *One Single Use* – setiap kotak bonus hanya bisa dipakai satu kali, artinya nilai akan dikalikan hanya satu kali, jika suatu kata terbentuk dari salah satu huruf dari kata yg sudah terbentuk pada salah satu kotak bonus maka nilai tidak dikalikan.

Permainan dimainkan dengan 100 keping huruf, 98 huruf dari a sampai z dengan jumlah yang ditentukan dan 2 keping kosong adalah bonus atau blank, bisa digunakan untuk mengganti huruf apa saja, tetapi tidak mempunyai score.

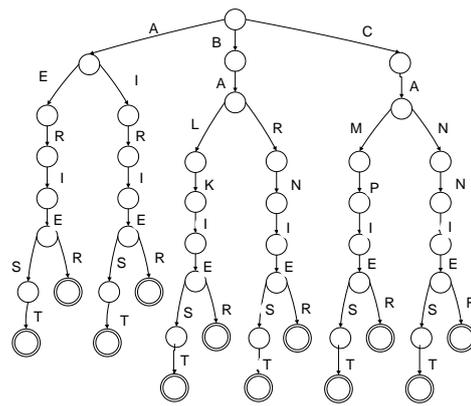
Point-point pada tiap huruf pada permainan *Scrabble* :

- 0 Points - *Blank tile*.
- 1 Point - A, E, I, L, N, O, R, S, T dan U.
- 2 Points - D dan G.
- 3 Points - B, C, M dan P.
- 4 Points - F, H, V, W dan Y.
- 5 Points - K.
- 8 Points - J dan X.
- 10 Points - Q dan Z

3. ALGORITMA DAWG

Kata yang dapat dimainkan dalam permainan *scrabble* adalah semua kata yang ada didalam kamus. Sehingga untuk mendukung AI dalam program ini perlu dibangun sebuah prosedur untuk mencari kata yang akan dimainkan pada kamus, dengan asumsi bahwa komputer mempunyai semua perbendaharaan kata yang ada didalam kamus. Untuk melakukan pencarian kata dalam kamus perlu dibuat struktur data yang merepresentasikan semua kata yang ada di dalam kamus, secara sederhana dapat direpresentasikan dengan sebuah tree. Tree yang dibangun memiliki node akhir yang khusus ditandai sebagai tanda bahwa node tersebut merupakan akhir dari kata.

Seperti contoh misalkan dalam sebuah kamus terdapat beberapa kata yaitu ; *AERIER*, *AERIEST*, *AIRIER*, *AIRIEST*, *BALKIER*, *BALKIEST*, *BARNIER*, *BARNIEST*, *CAMPIER*, *CAMPIEST*, *CANNIER*, dan *CANNIEST*. Kata-kata ini dapat direpresentasikan dalam sebuah tree seperti gambar 1.

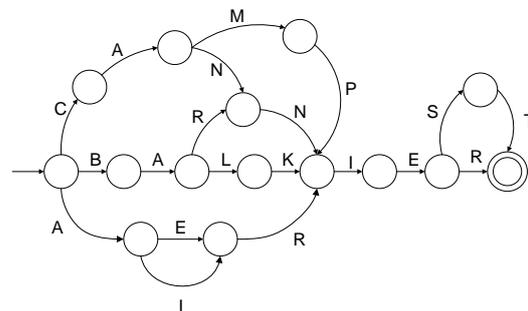


Gambar 1. Pohon huruf (trie)

Seperti yang terlihat pada gambar 1, terdapat banyak sekali huruf-huruf yang seharusnya dapat digunakan bersama. Jika struktur data pada pohon pada gambar 1 dapat diperkecil atau node yang sama dipakai bersama maka proses pencarian akan lebih cepat dan memori yang terpakai akan lebih kecil. Idennya adalah dengan memakai bersama awalan dan akhiran yang sama.

Struktur data yang merepresentasikan kata-kata dengan efisiensi ruang dengan menggunakan awalan dan akhiran yang sama bersama-sama disebut *directed acyclic word graph* (A. Appel and G. Jacobson, 1988) (*Directed acyclic word graph*, 2010)

Directed Acyclic Word Graph atau DAWG adalah sebuah struktur data yang menyimpan kosakata atau kata dimana dalam penggunaannya DAWG membutuhkan memory yang sangat kecil, dimana setiap awalan dan akhiran digunakan bersama-sama untuk memperkecil jumlah ruang yang diperlukan untuk menyimpan kosakata (A. Appel and G. Jacobson, 1988), sehingga memperkecil waktu yang diperlukan untuk meretrieve kata. DAWG dapat direpresentasikan seperti gambar 2.

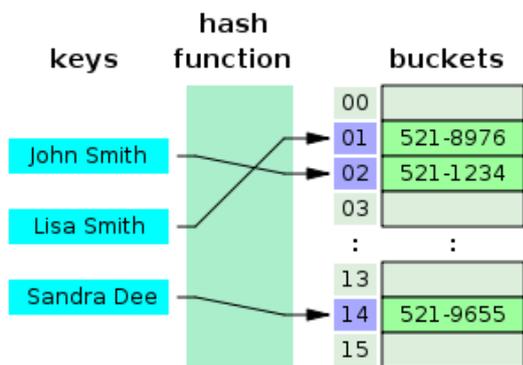


Gambar 2. Sebuah Directed Acyclic Word Graph

4. STRUKTUR PENYIMPANAN KAMUS DATA PADA JAVA

Hasil implementasi algoritma *DAWG* adalah berupa graph finite automata yang dapat menerima semua kata bahasa Inggris yang terdapat dalam data pengujian dalam penelitian ini, dimana kata-kata yang dipakai hanya sebuah sampel data yang terdiri dari 178691 kata.

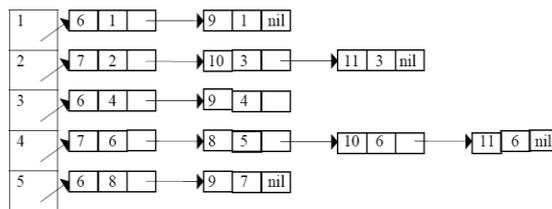
Dalam implementasinya sendiri semua node-node yang dibentuk oleh algoritma *DAWG* disimpan menggunakan metode yang telah disediakan oleh Java, dimana semua data yang berkaitan akan dimaping kedalam sebuah metode yaitu hashtable atau hash map. Hash table adalah data struktur yang memanfaatkan fungsi hash dalam meng-asosiasikan sebuah *key* dengan sebuah nilai, sehingga *key* menjadi index dari data array yang disimpan, dalam hal ini index berbeda dengan kunci tetapi untuk mencari index dihitung dari kunci dengan menggunakan fungsi *hash*. Karena mirip dengan sebuah array sehingga tabel hash juga dikatakan *associative array*. Misal kan terdapat data nama dan nomor telpon, maka Tabel hash akan menjadi seperti gambar 3.



Gambar 3. Tabel Hash

Sedangkan dalam penyimpanan kamus data pada penelitian ini yang menjadi index adalah semua node mulai dari root sampai leaf simpul terakhir yang akan diurut mulai dari simpul paling kiri ke simpul paling kanan. Dan huruf-huruf yang terdapat pada edge masing-masing node akan disimpan sebagai nilai yang ditunjuk oleh index yang berupa node itu sendiri. Setiap huruf disimpan dalam bentuk integer dimana a direpresentasikan sebagai 1 dan z sebagai 26 dengan nilai biner.

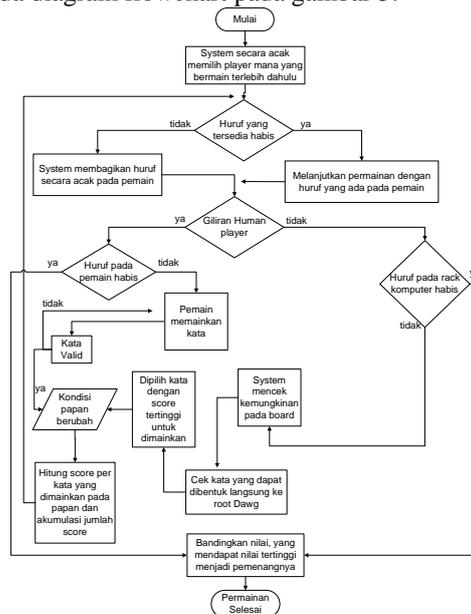
Nilai array tiap index akan ditempati oleh huruf yang terdapat pada edge yang meninggalkan node tersebut, node pertama yang meninggalkan node tersebut akan disimpan terlebih dahulu diikuti edge selanjutnya berurutan dari simpul paling kiri. Hal ini mungkin terjadi karena setiap lokasi array adalah suatu list berantai yang berisi pasangan kunci dan nilai, jika suatu nilai mempunyai kode hash yang sama maka nilai tersebut akan menepati list yang sama. Hal ini dapat dilihat pada gambar 4.



Gambar 4. Array List Berantai

5. PERANCANGAN SISTEM

Berikut adalah alur permainan scrabble sesuai dengan aturan permainannya yang digambarkan pada diagram flowchart pada gambar 5.



Gambar 5. Diagram Flowchart Game Scrabble

Permainan scrabble didukung oleh kamus data yang memuat kata-kata yang akan digunakan dalam menentukan apakah kata yang dimainkan valid atau tidak ataukah kata yang dimainkan pemain terdapat didalam kamus atau tidak, karena kata yang dapat dimainkan dalam permainan hanya kata yang ada dalam kamus. Untuk itu dibentuk kamus data menggunakan algoritma *Directed Acyclic Word Graph* sebelum dibentuk nya aplikasi permainan.

6. IMPLEMENTASI ALGORITMA DAWG

Algoritma *DAWG* berperan dalam membantu system menggenerate langkah dengan cara membangun struktur data dari semua kata yang terdapat dalam kamus berbentuk finite automata dengan jumlah state paling minimal (Inenaga Shunusuke, dkk, 2001)

Berikut proses system dalam memilih langkah yang dibantu oleh DAWG :

- 1) Jika pada papan permainan belum terdapat kata maka system langsung melakukan searching kata mulai dari root pada kamus data dawg dari 7 kata acak yang didapat sehingga semua kata yang didapat merupakan kata yang valid.
- 2) Kemudian dipilih kata dengan nilai tertinggi untuk dimainkan dan diletakkan pada papan permainan.
- 3) Jika pada papan telah terdapat kata, maka system melakukan pencarian pada papan huruf-huruf yang mungkin untuk digabung dengan 7 huruf acak pada pemain dilihat dari posisinya.
- 4) Pencarian pada papan langsung dilakukan searching pada kamus data dawg.
- 5) Pencarian dipisahkan antara kata yang terbentuk secara horizontal dan 4vertical.
- 6) Setiap kata yang terbentuk dibandingkan dengan kata yang terbentuk berikutnya dan kata dengan score tertinggi dipilih untuk dimainkan, begitu juga kata yang terbentuk secara vertical dan horizontal dibandingkan score yang dan dipilih score yang tertinggi untuk dimainkan pada papan.
- 7) Kata yang dipilih untuk dimainkan kemudian ditampilkan pada papan.

7. ANALISA HASIL PENGUJIAN

7.1 Analisa Pengujian Algoritma Pembentuk kamus data

Pembentukan kamus data menggunakan algoritma DAWG bertujuan untuk memperkecil ruang penyimpanan node-node yang berisi huruf-huruf pembentuk kata dengan melakukan sharing huruf-huruf yang sama atau awalan dan akhiran yang sama pada kata-kata yang berbeda. Sehingga penyimpanan kamus data pada memori akan semakin kecil, untuk itu perlu dibuktikan apakah hasil convert algoritma DAWG yang dibentuk mempunyai ukuran file lebih kecil dibandingkan file asli kamus data tersebut. Untuk itu dapat dilihat pada table 1 pada lampiran, ukuran file asli dan ukuran file kamus data yang telah dikonversi.

Dari hasil pengujian dapat disimpulkan bahwa semakin banyak node yang dapat di hemat oleh DAWG maka space memory yang terpakai pun akan semakin berkurang dari ukuran file awal. Dan dari pengujian no 1 dan no 2 terlihat perbedaan node yang disimpan yaitu 1 node dan 1 edge dan perbedaan ukuran setelah di convert yaitu 4 byte sehingga terlihat bahwa 1 node dan 1 edge pada java disimpan dalam ukuran 4 byte.

7.2 Hasil Analisa Pengujian Strategi Permainan Scrabble

Setelah diberikan sepuluh sampel acak kombinasi huruf yang masing-masing digabung dengan huruf s dan q dapat terlihat perbedaan pengaruh huruf s dan q dimana pada tabel *basic rack evaluation* s memiliki point evaluasi 7,5

sedangkan q mempunyai point -11,5 dengan teori bahwa semakin besar point yang dimiliki semakin efektif satu huruf dikombinasikan dengan huruf-huruf lain. Hasil pengujian dapat dilihat pada tabel 2 pada lampiran.

Dari informasi yang didapat terbukti bahwa huruf s mempunyai efektifitas lebih tinggi untuk dikombinasikan dengan huruf-huruf lain dari pada huruf q, dan berpengaruh terhadap nilai tertinggi yang dapat dihasilkan dan terbukti bahwa nilai evaluasi rack yang lebih tinggi kemungkinan score yang dapat dibentuk pun lebih tinggi. Dalam hal ini dirancang 3 strategi, yaitu :

1. Strategi pertama yaitu dimana player akan memilih pilihan kata di setiap giliran berdasarkan score tertinggi yang didapat tanpa mengevaluasi rack.
2. Strategi kedua yaitu sedikit perbaikan dari strategi pertama dimana pemilihan kata yang akan dimainkan tetap berdasarkan nilai tertinggi tetapi pada saat kasus tertentu dimana terdapat beberapa nilai tertinggi maka diantara score tertinggi tersebut pilihan kata yang akan dimainkan berdasarkan evaluasi rack.
3. Strategi ketiga yaitu pemilihan kata berdasarkan gabungan dari score dan evaluasi rack, dimana nilai score akan ditambah dengan point rack.

Dari 10 kali percobaan yang dilakukan dapat terlihat bahwa strategi 3 dapat membentuk rata-rata nilai tertinggi lebih besar dari strategi 2.

Untuk lebih memperjelas persentase kemenangan tiap-tiap strategi maka dilakukan 60 kali percobaan dan didapat data hasil pengujian perbandingan masing-masing strategi pada tabel 3 pada lampiran. Dari hasil pengujian didapatkan informasi sebagai berikut:

- Strategi 2 mempunyai persentase kemenangan lebih tinggi dibandingkan strategi 1
- Strategi 3 mempunyai persentase kemenangan lebih tinggi dibandingkan strategi 1
- Strategi 3 mempunyai persentase kemenangan lebih tinggi dibandingkan strategi 2

8. KESIMPULAN

Dari hasil pengujian dan analisis yang telah dilakukan, dapat diambil beberapa kesimpulan sebagai berikut :

1. Untuk menyimpan kata-kata atau membuat sebuah kamus data, algoritma directed acyclic word graph sangat mangkus dalam hal menghemat ruang penyimpanan tergantung kondisi kamus data.
2. Semakin banyak node yang redundant atau node yang dapat dihemat oleh *dawg* maka ukuran file pun akan semakin jauh berkurang dari ukuran file awal.
3. Strategi permainan scrabble yang memilih kata ditiap giliran hanya berdasarkan score tertinggi

mempunyai persentase kemenangan hanya 40% dari 60 kali percobaan dibanding strategi 2 yang telah memiliki evaluasi rack.

PUSTAKA

- A. Appel and G. Jacobson, 1988, The world's fastest Scrabble program, *Commun. ACM*, 31,(5), 572–578,585
- C. Maxime and V. Renaud, 1997, On Compact Directed Acyclic Word Graphs, Institut Gaspard Monge Université de Marne-La-Vallée, 2, rue de la Butte Verte, F-93160 Noisy-Le-Grand.
- C. Lucchesi and T. Kowaltowski, 1992, Application of Finite Automata Representing Large Vocabularies, Department Science and Computer, Universitas Estadual de Campinas Brasil.
- Directed acyclic word graph, 2010, <http://www.wutka.com/dawg.html> diakses pada tanggal 20 Februari 2010.
- Inenaga Shunusuke, Shinohara Ayumi, Takeda Masayuki, Arikawa Setsuo, 2001, Compact Directed Acyclic Word Graphs for a Sliding Window, Proc. 8th Int. Symp. String Processing and Information Retrieval, 2001. SPIRE 2001, pp. 96–110
- Sheppard, Brian, 2002. Towards Perfect Play of Scrabble. PhD thesis, IKAT/Computer Science Department
Universiteit Maastricht,
ISBN 90-5278-351-9
- Scrabble history, 2010, <http://www.mattelscrabble.com/en/adults/history/> diakses pada tanggal 03 MEI 2010.
- S. Gordon, 1993, A Comparison Between Probabilistic Search and Weighted Heuristics in A Game with Incomplete Information, AAAI Fall Symposium Series, Raleigh, NC

LAMPIRAN

Tabel 1 Form pengujian algoritma DAWG

No.	Jumlah kata dalam kamus	Ukuran file asli	Ukuran file setelah di convert dengan DAWG	Banyaknya node yang terpakai setelah di convert	Jumlah node yang dihemat
1.	1 huruf 1 kata	1 byte	12 byte	2 node	0 node
2.	2 huruf 1 kata	2 byte	16 byte	3 node	0 node
3.	3	13 byte	32 byte	5 node	2 node
4.	12	108 byte	88 Byte	15 Node	22 Node
5.	178691	1,897 kb	495 KB	54259 Node	224848 Node

Tabel 2 Tabel hasil perbandingan dua huruf

No	Sampel acak kombinasi huruf	Huruf yang akan digabung	Jumlah pilihan langkah yang dapat dibentuk	Huruf tertinggi yang dapat dibentuk	Score huruf tertinggi yang dapat dibentuk
1	alnrru	s	322	Lunars	14
2	alnrru	q	120	Qua	24
3	begjio	s	364	Bogjes	24
4	begjio	q	114	Qi	22
5	aeikst	s	876	Steaks	30
6	aeikst	q	560	Kites	28
7	eefopy	s	332	Sepoy	28
8	eefopy	q	96	Foy	18
9	ciprmv	s	312	Scrimp	30
10	ciprmv	q	72	Crimp	28
11	imnpr	s	270	Prism	24
12	imnpr	q	80	Qi	22
13	aadiin	s	298	Naiads	18
14	aadiin	q	124	Qadi	28
15	cdeiop	s	938	Copied	28
16	cdeiop	q	266	Copied	28
17	dijopr	s	344	Prods	22
18	dijopr	q	116	Qi	22
19	jolmtw	s	366	Jowls	46
20	jolmtw	q	124	Jowl	28

Tabel 3. Tabel hasil perbandingan beberapa strategi 60 kali percobaan

No	Pasangan	Player	Jumlah Kemenangan	Presentase Kemenangan
1	Strategi 1 (P1) VS Strategi 2(P2)	P1	24	40
		P2	36	60
2	Strategi 1 (P1) VS Strategi 3(P2)	P1	26	43,33333333
		P2	34	56,66666667
3	Strategi 2 (P1) VS Strategi 3(P2)	P1	27	45
		P2	33	55