

Chess

Proyek matakuliah Code Reengineering akan dibagi menjadi 2 deadline.

- Deadline pertama: pertemuan terakhir sebelum UTS (4 April 2020)
- Deadline kedua: pertemuan terakhir sebelum UAS (13 Juni 2020)

Masing-masing deadline memiliki bobot 50% nilai TM.

1st Deadline

- *draw board*
- *move notation (coordinate or algebraic)*
- *all piece regular moves*
- *check*
- *castling*
- *validate illegal move: king must always in safe position*

2nd Deadline

- *en passant*
- *pawn promotion*
- *checkmate*
- *draw on stalemate position*
- *unit tests: test coverage min. 90%*
- *reduce code smells*

Penjelasan Fitur

Semua gerakan buah catur sesuai dengan [Chess movement wikipedia](#).

Kondisi menang kalah sesuai dengan [Checkmate wikipedia](#).

Program akan menampilkan 1-0 bila putih menang. 0-1 bila hitam menang.

Kondisi seri yang perlu dibuat adalah kondisi *stalemate*. Bila seri, program akan menampilkan 1/2-1/2 .

Draw Board

R	N	B	Q	K	B	N	R	8
P	P	P	P	P	P	P	P	7
-	+	-	+	-	+	-	+	6
+	-	+	-	+	-	+	-	5
-	+	-	+	-	+	-	+	4
+	-	+	-	+	-	+	-	3
p	p	p	p	p	p	p	p	2
r	n	b	q	k	b	n	r	1
A	B	C	D	E	F	G	H	

Agar lebih simpel, tampilan proyek kali ini tidak perlu menggunakan warna. Bidak hitam diwakili dengan huruf besar. Bidak putih diwakili dengan huruf kecil. Kotak putih diwakili dengan tanda `-`, kotak hitam diwakili dengan tanda `+`.

Masing-masing bidak diwakili dengan huruf:

- bidak hitam: P = pawn, R = rook, N = knight, B = bishop, Q = queen, K = king
- bidak putih: p = pawn, r = rook, n = knight, b = bishop, q = queen, k = king

Berikan keterangan *rank* pada kanan papan dan keterangan *file* pada bawah papan.

Catatan: Huruf besar/kecil pada tampilan papan tidak ada hubungan sama sekali dengan move notation. Huruf besar/kecil murni hanya untuk keperluan tampilan.

Move Notation

Anda boleh memilih *coordinate* atau *algebraic notation*. Disarankan mengerjakan *coordinate* terlebih dahulu karena jauh lebih mudah.

Lihat contoh gerakan dalam tabel disini [Chess notation wikipedia](#) agar lebih paham.

Coordinate notation

Gerakan tidak perlu ada keterangan simbol buah catur. Cukup pedulikan koordinat asal dan koordinat tujuan. *Rank* adalah angka 1 sampai 8. *File* adalah huruf kapital A sampai H.

Contoh *query* gerakan **Pawn** dari E2 ke E4: **E2-E4**

Castling

Untuk *castling* gunakan koordinat **King** lalu ke kiri 2 langkah untuk *queenside castling* atau ke kanan 2 langkah untuk *kingside castling*.

Contoh *white kingside castling*: **E1-G1**

Pawn promotion

Untuk *pawn promotion*, tambahkan keterangan bidak promosi di akhir *query*. Bidak promosi bersimbolkan (Q = queen, N = knight, R = rook, B = bishop).

Contoh **Pawn** putih promosi menjadi **Knight**: **F7-F8N**

En passant

Untuk *en passant*, gerakkan **Pawn** ke koordinat tujuan, contoh:

E2-E4

G8-F6

E4-E5

D7-D5

E5-D6

Algebraic notation

Dalam notasi ini, tidak ada keterangan koordinat awal, program akan menebak sendiri buah catur mana yang akan digerakkan.

Bila di awal *query* tidak ada keterangan simbol buah catur di karakter pertama *query*, maka gerakan diasumsikan untuk **Pawn**. *Query* cukup menuliskan koordinat tujuan setelah keterangan simbol. *Rank* adalah angka 1 sampai 8. *File* adalah huruf kecil a sampai h.

Contoh *query* gerakan bidak dari E2 ke E4: **e4**

Contoh *query* gerakan kuda dari G8 ke F5: **Nf5**

Castling

Untuk *castling* gunakan **0-0-0** untuk *queenside castling* atau **0-0** untuk *kingside castling*.

Pawn promotion

Untuk *pawn promotion*, tambahkan keterangan bidak promosi di akhir *query*. Bidak promosi bersimbolkan (Q = queen, N = knight, R = rook, B = bishop).

Contoh **Pawn** putih promosi menjadi **Knight**: **f8N**

En passant

Untuk *en passant*, gerakkan **Pawn** ke koordinat tujuan, contoh:

e4
Nf6
e5
d5
d6

Untuk mempersingkat *query, algebraic notation* dalam proyek ini tidak perlu mengikutsertakan simbol makan, skak, skakmat atau en passant.

Disambiguating Moves

Dalam *algebraic notation*, terdapat kemungkinan koordinat tujuan dapat diakses lebih dari satu buah catur. Bila hal ini terjadi, *query* dinyatakan ambigu. Teknik menyelesaikannya sesuai dengan yang ada di [link ini](#).

Contoh:

-	+	-	R	-	+	-	R	8
B	-	+	-	+	K	+	-	7
-	+	-	B	-	+	-	+	6
r	-	+	-	+	-	+	-	5
-	+	-	+	q	+	-	q	4
+	-	+	-	+	-	+	-	3
-	+	-	+	-	+	-	+	2
r	k	+	-	+	-	+	q	1
A	B	C	D	E	F	G	H	

Setelah gerakan-gerakan berikut dijalankan:

R1a3
Bdb8
Qh4e1
Rdf8
Q4e2

Hasilnya:

-	B	-	+	-	R	-	R	8
B	-	+	-	+	K	+	-	7
-	+	-	+	-	+	-	+	6
r	-	+	-	+	-	+	-	5
-	+	-	+	-	+	-	+	4
r	-	+	-	+	-	+	-	3
-	+	-	+	q	+	-	+	2

+	k	+	-	q	-	+	q	1
A	B	C	D	E	F	G	H	

Stalemate

Terdapat beberapa peraturan catur untuk mengakhiri permainan dengan hasil seri. Untuk proyek kali ini, Anda hanya perlu membuat *code* untuk satu peraturan seri, yaitu *stalemate*. Kondisi *stalemate* adalah ketika tidak ada lagi buah catur yang dapat digerakkan, namun tidak dalam kondisi skak.

Contoh:

-	+	-	K	-	+	-	+	8
+	-	+	B	+	-	+	-	7
-	Q	-	+	-	+	-	+	6
+	r	+	-	+	-	+	-	5
k	p	-	+	-	+	-	+	4
p	p	+	-	+	-	+	-	3
-	+	-	+	-	+	-	+	2
+	-	+	-	+	-	+	-	1
A	B	C	D	E	F	G	H	

Pada contoh diatas, tidak ada buah catur putih yang bisa digerakkan tanpa mengancam keselamatan raja. Namun raja putih tidak dalam kondisi skak. Kondisi ini adalah kondisi seri.