Assignment 5.

11. **Due 03/02/15.**
Generate Zone with the main trajectory shown in the figure below. Use the grammar of Zones G_Z, show values of the key functions and sets. Avoid unnecessary details.

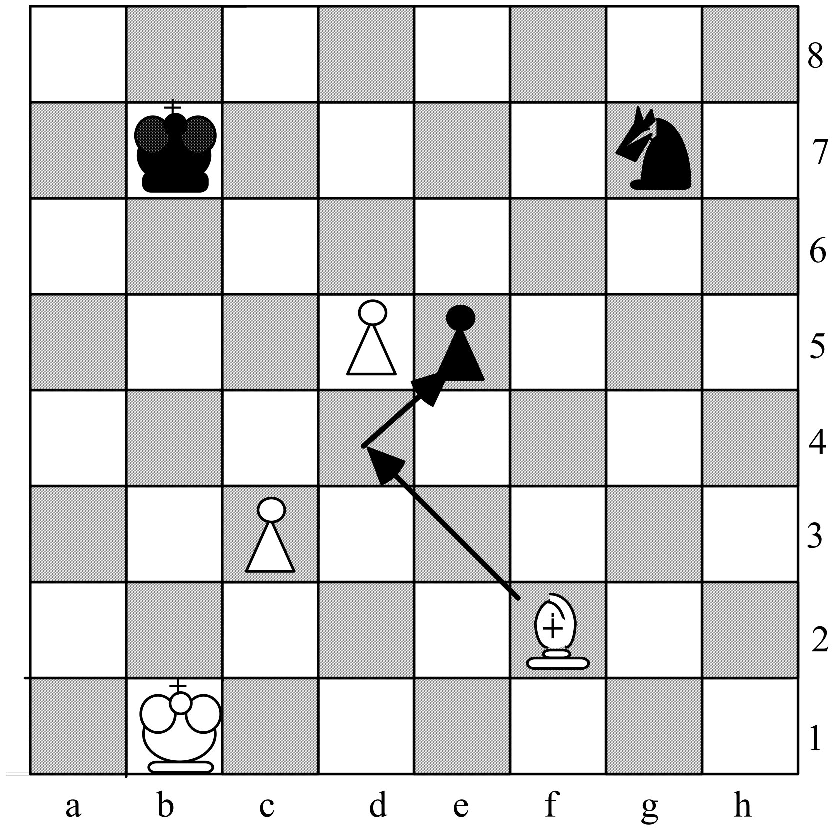

Figure 1.

12. **Due 03/02/15.**
Modify the grammar of Zones G_Z in order to generate all the attack Zones within horizon H in <u>any given position</u> by ONE application of the grammar. The original grammar G_Z generates only one Zone (with a given start and end of the main trajectory).
Apply it for generation of all the Zones in the initial position of the R. Reti endgame (Figure 2) and for other position of your choice. Do not forget about pawn promotion. When modifying the grammar you can limit the number and types of Zones to be generated by this grammar. What are the input parameters of your grammar?
Show generation. Avoid unnecessary details.

RETI ENDGAME

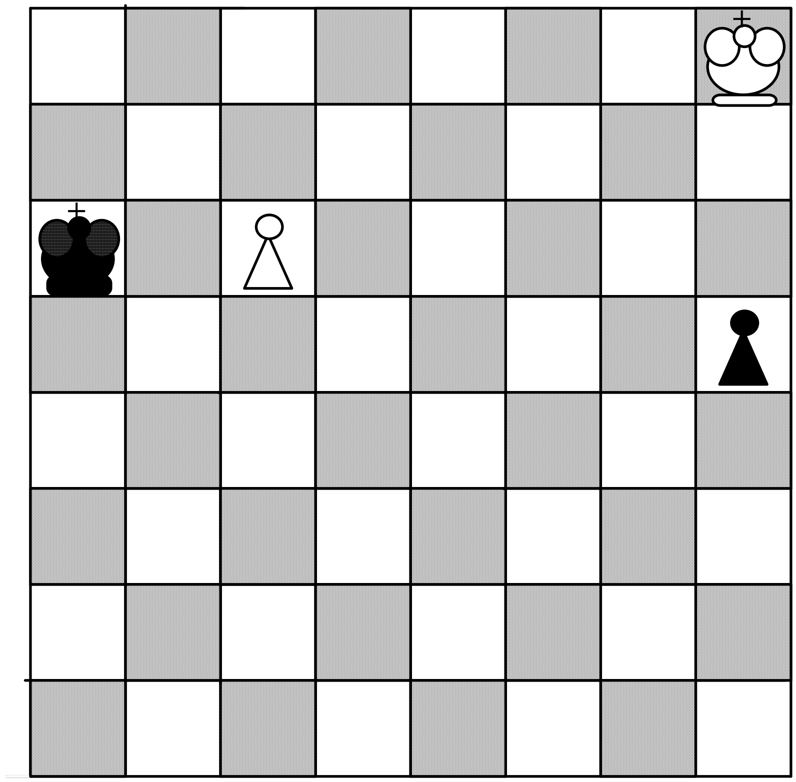

Figure 2. White begins and makes a draw

13. Project 3: Due 03/16/15 (3 weeks)

Write a program that can generate all the attack Zones in an arbitrary position. In particular it should be able to generate all the Zones for the positions shown in Fig. 1 and 2. of this assignment. Use your program from the second project for generating shortest and admissible trajectories. Use the grammar of Zones. Every Zone generated should be printed on a separate diagram as a graph (including all the trajectories and pieces).

Output:

Print all the required attack Zones from tasks 11. and 12. of this assignment. Print also all the Zones in an arbitrary position of your choice.

Remark. Read Chapter 13 of the book about the problems with data structures for trajectories and Zones.

1	$\boldsymbol{Q_1}$	$\boldsymbol{S}(u, v, w) \to \boldsymbol{A}(u, v, w)$		*two*	ø
2_i	$\boldsymbol{Q_2}$	$\boldsymbol{A}(u, v, w) \to \boldsymbol{t}(\boldsymbol{h_i^o}(u), l_o+1)$ $\boldsymbol{A}((0, 0, 0), \boldsymbol{g}(\boldsymbol{h_i^o}(u), w), zero)$	TIME(z)=$\boldsymbol{DIST}$(z,$\boldsymbol{h_i^o}(u)$)	3	ø
3	$\boldsymbol{Q_3}$	$\boldsymbol{A}(u, v, w) \to \boldsymbol{A}(\boldsymbol{f}(u, v), v, w)$	NEXTTIME(z)= $\boldsymbol{init}(u$, NEXTTIME(z))	*four*	5
4_j	$\boldsymbol{Q_4}$	$\boldsymbol{A}(u, v, w) \to \boldsymbol{t}(\boldsymbol{h_j}(u)$, TIME(y))) $\boldsymbol{A}(u, v, \boldsymbol{g}(\boldsymbol{h_j}(u), w))$	NEXTTIME(z)= $\boldsymbol{ALPHA}$(z, $\boldsymbol{h_j}$(u), TIME(y) – l_j+1)	3	3
5	$\boldsymbol{Q_5}$	$\boldsymbol{A}(u, v, w) \to \boldsymbol{A}((0, 0, 0), w, zero)$	TIME(z) = NEXTTIME(z)	3	6
6	$\boldsymbol{Q_6}$	$\boldsymbol{A}(u, v, w) \to \boldsymbol{e}$		ø	ø

$\boldsymbol{Q_1}(u) = (\mathrm{ON}(p_o) = x) \wedge (\mathrm{MAP}_{x,p_o}(y) \le l \le l_o) \wedge (\exists q\, ((\mathrm{ON}(q) = y) \wedge (\mathrm{OPPOSE}(p_o, q))))$

$\boldsymbol{Q_2}(u) = T$; $\boldsymbol{Q_3}(u) = (x \neq n) \vee (y \neq n)$

$\boldsymbol{Q_4}(u) = (\exists p\, ((\mathrm{ON}(p) = x) \wedge (l > 0) \wedge (x \neq x_o) \wedge (x \neq y_o)) \wedge ((\neg \mathrm{OPPOSE}(p_o, p) \wedge (\mathrm{MAP}_{x,p}(y) = 1)) \vee (\mathrm{OPPOSE}(p_o, p) \wedge (\mathrm{MAP}_{x,p}(y) \le l)))$ $\boldsymbol{Q_5}(w) = (w \neq zero)$;$\boldsymbol{Q_6} = T$

$$\boldsymbol{init}(u, \mathrm{r}) = \begin{cases} 2\mathrm{n}, & \text{if } u = (0,0,0), \\ \mathrm{r}, & \text{if } u \neq (0,0,0). \end{cases}$$

$$f(u,\ v) = \begin{cases} (\mathrm{x}+1, \mathrm{y}, l), & \text{if } ((\mathrm{x} \neq \mathrm{n}) \wedge (l > 0)) \vee ((\mathrm{y} = \mathrm{n}) \wedge (l \le 0)) \\ (1,\ \mathrm{y}+1,\ \mathit{TIME}(\mathrm{y}+1) \times v_{\mathrm{y}+1}), & \text{if } (\mathrm{x} = \mathrm{n}) \vee ((l \le 0) \wedge (\mathrm{y} \neq \mathrm{n})). \end{cases}$$

Let $t_o \in \mathbf{L_t}^{l_o}(S)$, $t_o = \boldsymbol{a}(z_o)\boldsymbol{a}(z_1)...\boldsymbol{a}(z_m)$, $t_o \in t_{p_o}(z_o, z_m, m)$;

If $((z_m = y_o) \wedge (p = p_o) \wedge (\exists\, k\, (1 \le k \le m) \wedge (x = z_k))) \vee$
$(((z_m \neq y_o) \vee (p \neq p_o)) \wedge (\exists\, k\, (1 \le k \le m - 1) \wedge (x = z_k)))$
then $\boldsymbol{DIST}(x, p_o, t_o) = k+1$ **else** $\boldsymbol{DIST}(x, p_o, t_o) = 2n$

$$\mathit{ALPHA}(\mathrm{x}, \mathrm{p_o}, \mathrm{t_o}, \mathrm{k}) = \begin{cases} \mathit{max}\,(\mathit{NEXTTIME}(\mathrm{x}), \mathrm{k}), & \text{if } (\mathit{DIST}(\mathrm{x}, \mathrm{p_o}, \mathrm{t_o}) \neq 2\mathrm{n}) \wedge (\mathit{NEXTTIME}(\mathrm{x}) \neq 2\mathrm{n}); \\ \mathrm{k}, & \text{if } \mathit{DIST}(\mathrm{x}, \mathrm{p_o}, \mathrm{t_o}) \neq 2\mathrm{n}) \wedge (\mathit{NEXTTIME}(\mathrm{x}) = 2\mathrm{n}); \\ \mathit{NEXTTIME}(\mathrm{x}), & \text{if } \mathit{DIST}(\mathrm{x}, \mathrm{p_o}, \mathrm{t_o}) = 2\mathrm{n}). \end{cases}$$

$$\boldsymbol{g}_{\mathbf{r}}(\mathrm{p_o}, \mathrm{t_o}, w) = \begin{cases} 1, & \text{if } \boldsymbol{DIST}(\mathrm{r}, \mathrm{p_o}, \mathrm{t_o}) < 2n, \\ w_{\mathrm{r}}, & \text{if } \boldsymbol{DIST}(\mathrm{r}, \mathrm{p_o}, \mathrm{t_o}) = 2\mathrm{n}. \end{cases}$$

$$\mathrm{TRACKS}_{p_o} = \{p_o\} \times \Big(\bigcup_{1 \le k \le l} \mathrm{L}[\,\mathbf{G_t^{(2)}}(x, y, k, p_o)]$$

If $\mathrm{TRACKS}_{p_o} = e$
then $\boldsymbol{h_i^o}(u) = e$

else $\mathrm{TRACKS}_{p_o} = \{(p_o, t_1), (p_o, t_2), \ldots, (p_o, t_b)\}, (b \le M)$ **and** $\boldsymbol{h_i^o}(u) = \begin{cases} (\mathrm{p_o}, \mathrm{t_i}), & \text{if } \mathrm{i} \le \mathrm{b}, \\ (\mathrm{p_o}, \mathrm{t_b}), & \text{if } \mathrm{i} > \mathrm{b}. \end{cases}$

$\mathrm{TRACKS} = \bigcup_{\mathrm{ON}(p)=x} \mathrm{TRACKS}_p$, where TRACKS_p is the same as for $\boldsymbol{h_i}^o$

If TRACKS = e
then $\boldsymbol{h_i}(u) = e$

else $\mathrm{TRACKS} = \{(p_1, t_1), (p_1, t_2), \ldots, (p_m, t_m)\}$, $(m \le M)$ **and** $\boldsymbol{h_i}(u) = \begin{cases} (\mathrm{p_i}, \mathrm{t_i}), & \text{if } \mathrm{i} \le \mathrm{m}, \\ (\mathrm{p_m}, \mathrm{t_m}), & \text{if } \mathrm{i} > \mathrm{m}. \end{cases}$

At the beginning : $u = (x_o, y_o, l_o)$, $w = zero$, $v = zero$, $x_o \in X$, $y_o \in X$, $l_o \in \mathbf{Z}_+$,
$p_o \in P$, and TIME(z)=2n, NEXTTIME(z)=2n for all z from X.