

Laporan Tugas Besar 2
IF3170 Intelegensi Buatan

MEMBUAT AGEN MINESWEEPER BERDASARKAN
KNOWLEDGE BASED SYSTEM



oleh:

Qurrata A'yuni	13518004
Hasna Roihan Nafiisah	13518008
Ferdina Wiranti Afifah	13518046
Wildan Zaim Syaddad	13518068

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG

2020 / 2021

A. Tahapan Membangun Aplikasi

1. Membuat Masukan

- a. Program menerima masukan *board* permainan (n). Ukuran board $4 \leq n \leq 10$.
- b. Program menerima masukan jumlah bom pada permainan ($nboard$).
- c. Program menerima masukan sebanyak $nboard$ titik koordinat untuk bom.

2. Membangun *Board*

- a. Program membangun *board* untuk permainan dengan men-*generate* titik koordinat berdasarkan masukan ukuran *board* (n).
- b. Setiap titik koordinat diinisiasi dengan nilai nol untuk kemudian akan diganti berdasarkan keberadaan bom di board.
- c. Pada awal permainan semua koordinat diberi status *closed* yang menyatakan bahwa koordinat tersebut tertutup.
- d. Pada *board* dibangun *rules* yang akan meng-*update* nilai koordinat berdasarkan keberadaan bom. Jika suatu bom berada pada koordinat (x,y) maka nilai koordinat di sekitarnya yang mengikuti arah mata angin akan ditambah satu. Terdapat delapan koordinat yang di-increment nilainya yaitu: ($x-1, y-1$), ($x-1, y+0$), ($x-1, y+1$), ($x+0, y-1$), ($x+0, y+1$), ($x+1, y-1$), ($x+1, y+0$), dan ($x+1, y+1$). Semua koordinat di sekitar bom nilainya akan di-*update* berdasarkan *rules* tersebut.
- e. Pada board dibuat *rules* yang akan memulai game yaitu jika *board* sudah selesai dibuat, lalu membuka koordinat(0,0).

3. Membuat *Rules* serta Aksi Agen untuk Memberikan Prediksi

- a. Membuka setiap tetangga dari koordinat(x,y), jika jumlah bom di sisi koordinat(x,y) adalah nol.
- b. Memasang *flag* pada tetangga koordinat(x,y), jika jumlah koordinat kosong di sisi koordinat(x,y) sama dengan jumlah bom di sisi koordinat(x,y).
- c. Membuka semua tetangga koordinat(x,y), jika jumlah *flag* di sekitar koordinat(x,y) sama dengan jumlah bom di sisi koordinat(x,y).

- d. Membuat *rules* status menang, yaitu ketika jumlah *flag* sama dengan jumlah bom.
- e. Membuat *rules* status kalah, yaitu ketika ada koordinat(x,y) yang terbuka yang koordinat(x,y) itu juga merupakan bom.

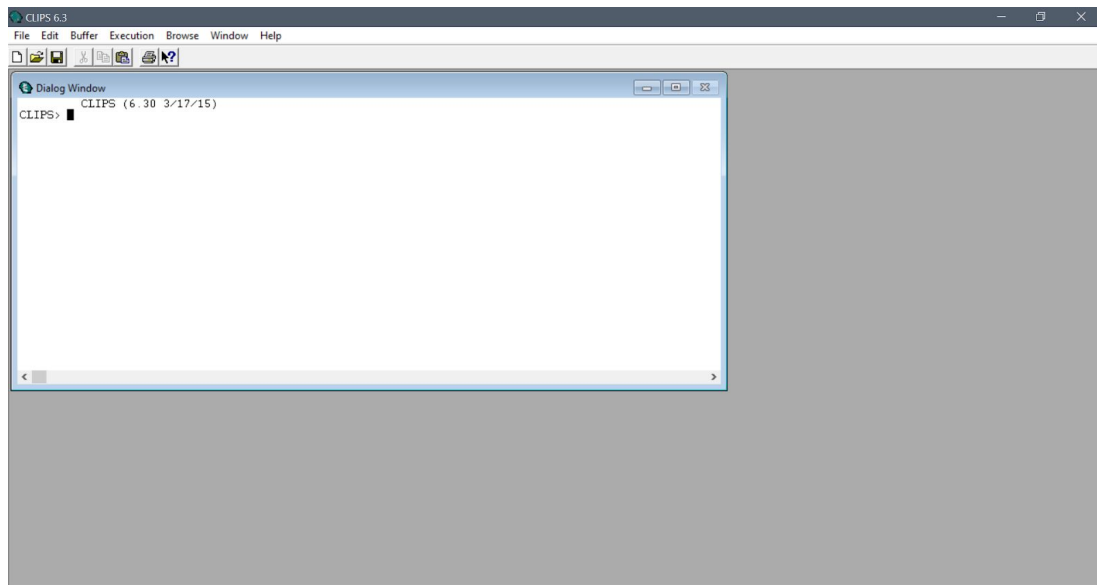
B. Repository

Berikut adalah link repository github pengerjaan tugas besar 2 intelegensi buatan milik kelompok kami.

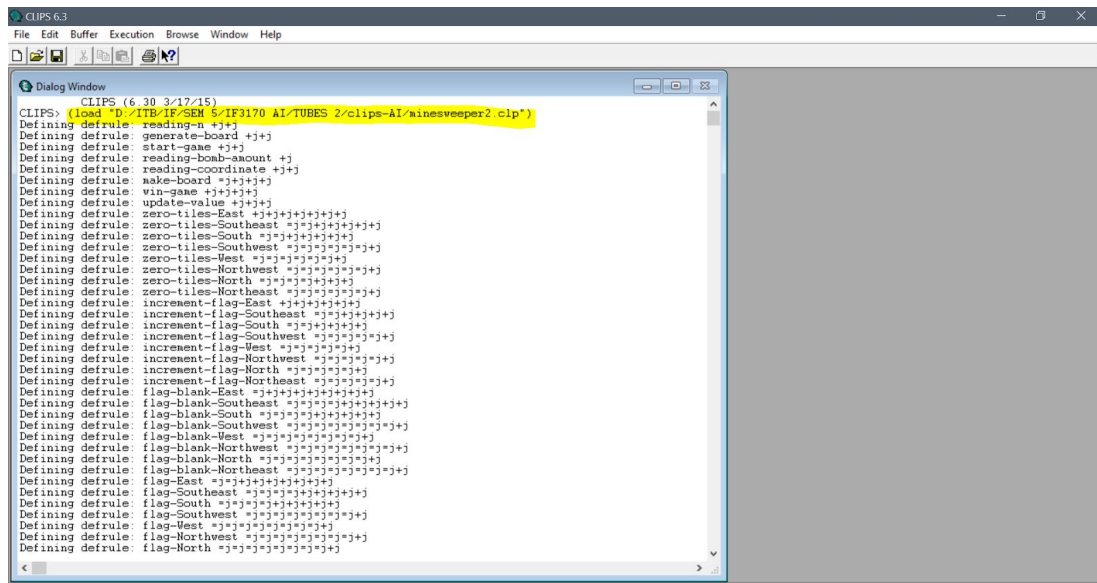
<https://github.com/hasnaroihan/clips-AI.git>

C. User Manual

1. Buka *dialog window* pada aplikasi CLIPS.

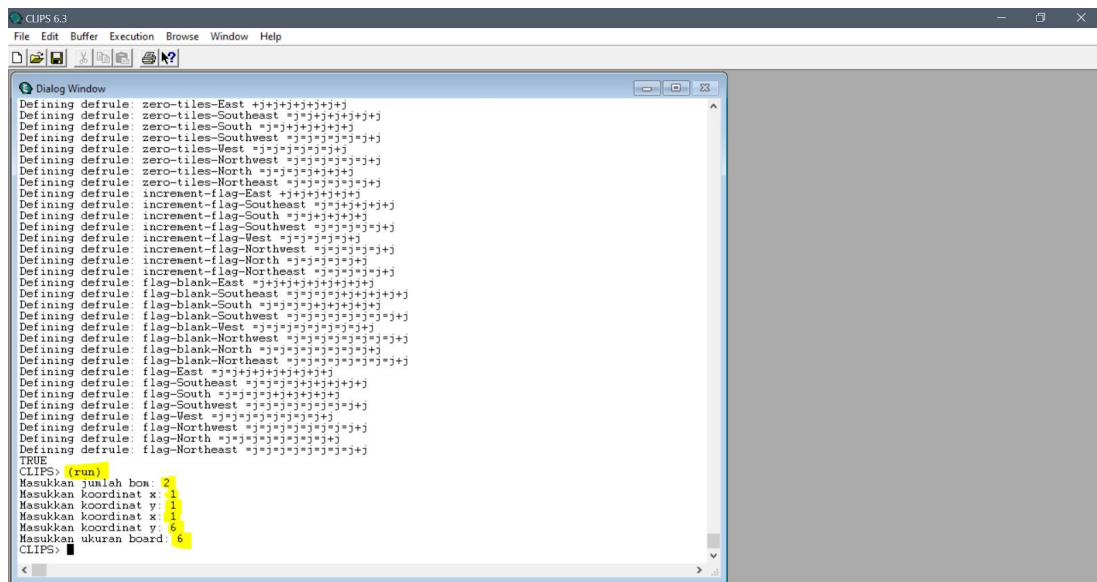


2. Lalu, gunakan *command* (load “direktori minesweeper2.clp”) untuk memuat file minesweeper2.clp yang akan dijalankan.



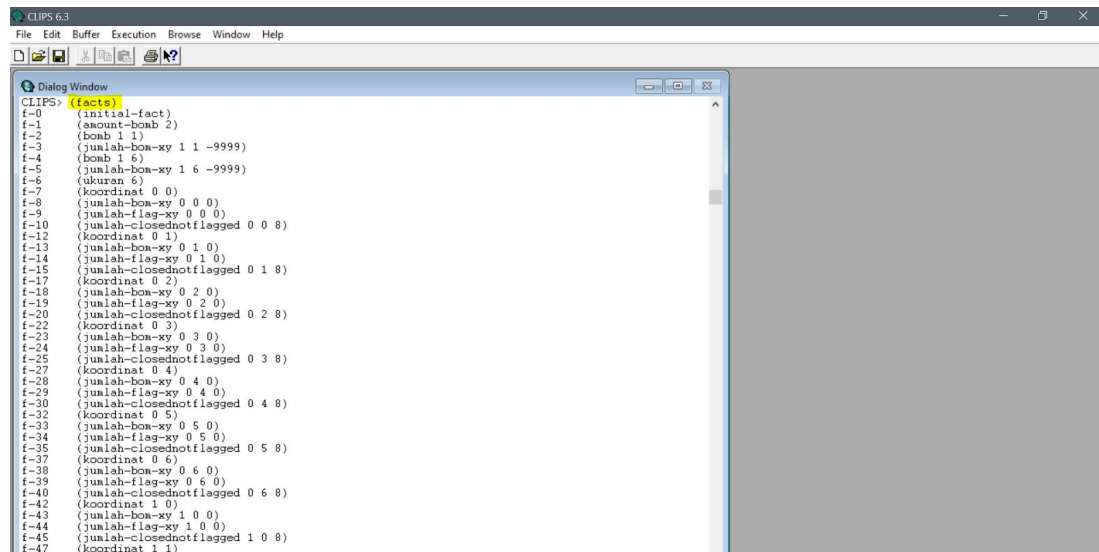
The screenshot shows the CLIPS 6.3 interface with a Dialog Window open. The window title is "CLIPS (6.30 3/17/15)". The command prompt shows the command: `CLIPS> (load "D:/ITB/IF/SEM 5/IF3170 AI/TUBES 2/clips-AI/minesweeper2.clp")`. Below the command, a list of rules is displayed, including: `Defining defrule: reading-n +j+j`, `Defining defrule: generate-board +j+j`, `Defining defrule: start-game +j+j`, `Defining defrule: reading-bomb-amount +j`, `Defining defrule: reading-coordinate +j+j`, `Defining defrule: make-board +j+j+j`, `Defining defrule: win-game +j+j+j+j`, `Defining defrule: update-value +j+j+j`, `Defining defrule: zero-tiles-East +j+j+j+j+j+j`, `Defining defrule: zero-tiles-Southeast +j+j+j+j+j+j`, `Defining defrule: zero-tiles-South +j+j+j+j+j+j`, `Defining defrule: zero-tiles-Southwest +j+j+j+j+j+j`, `Defining defrule: zero-tiles-West +j+j+j+j+j+j`, `Defining defrule: zero-tiles-Northwest +j+j+j+j+j+j`, `Defining defrule: zero-tiles-North +j+j+j+j+j+j`, `Defining defrule: zero-tiles-Northeast +j+j+j+j+j+j`, `Defining defrule: increment-flag-East +j+j+j+j+j+j`, `Defining defrule: increment-flag-Southeast +j+j+j+j+j+j`, `Defining defrule: increment-flag-South +j+j+j+j+j+j`, `Defining defrule: increment-flag-Southwest +j+j+j+j+j+j`, `Defining defrule: increment-flag-West +j+j+j+j+j+j`, `Defining defrule: increment-flag-Northwest +j+j+j+j+j+j`, `Defining defrule: increment-flag-North +j+j+j+j+j+j`, `Defining defrule: increment-flag-Northeast +j+j+j+j+j+j`, `Defining defrule: flag-blank-East +j+j+j+j+j+j`, `Defining defrule: flag-blank-Southeast +j+j+j+j+j+j`, `Defining defrule: flag-blank-South +j+j+j+j+j+j`, `Defining defrule: flag-blank-Southwest +j+j+j+j+j+j`, `Defining defrule: flag-blank-West +j+j+j+j+j+j`, `Defining defrule: flag-blank-Northwest +j+j+j+j+j+j`, `Defining defrule: flag-blank-North +j+j+j+j+j+j`, `Defining defrule: flag-blank-Northeast +j+j+j+j+j+j`, `Defining defrule: flag-East +j+j+j+j+j+j`, `Defining defrule: flag-Southeast +j+j+j+j+j+j`, `Defining defrule: flag-South +j+j+j+j+j+j`, `Defining defrule: flag-Southwest +j+j+j+j+j+j`, `Defining defrule: flag-West +j+j+j+j+j+j`, `Defining defrule: flag-Northwest +j+j+j+j+j+j`, and `Defining defrule: flag-North +j+j+j+j+j+j`.

3. Kemudian, gunakan *command* (run) untuk menjalankan program dan program akan langsung menjalankan file. Pada program ini, setelah file dijalankan pemain diminta untuk memasukkan jumlah bom yang akan diletakkan pada permainan, koordinat untuk meletakkan bom, dan masukan untuk ukuran *board* yang akan digunakan.



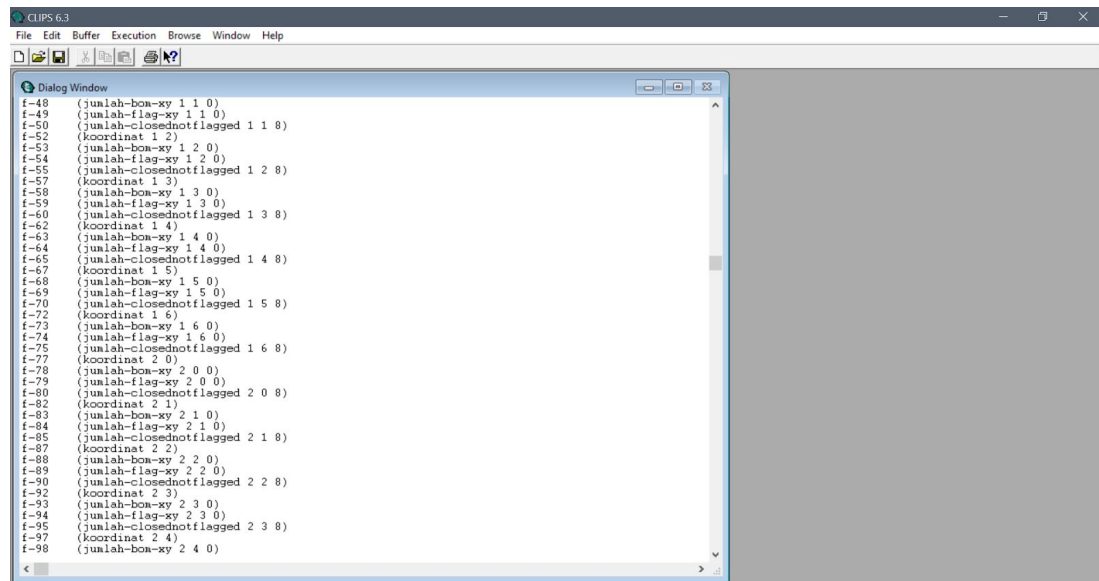
The screenshot shows the CLIPS 6.3 interface with a Dialog Window open. The window title is "CLIPS (6.30 3/17/15)". The command prompt shows the command: `CLIPS> (run)`. Below the command, the program prompts the user for input: `Masukkan jumlah bom: 2`, `Masukkan koordinat x: 1`, `Masukkan koordinat y: 1`, `Masukkan koordinat x: 6`, `Masukkan koordinat y: 6`, and `Masukkan ukuran board: 8`. The command prompt then shows: `CLIPS>`.

4. Selanjutnya, gunakan *command* (facts) untuk melihat hasil *debugging* dari program yang telah dijalankan.



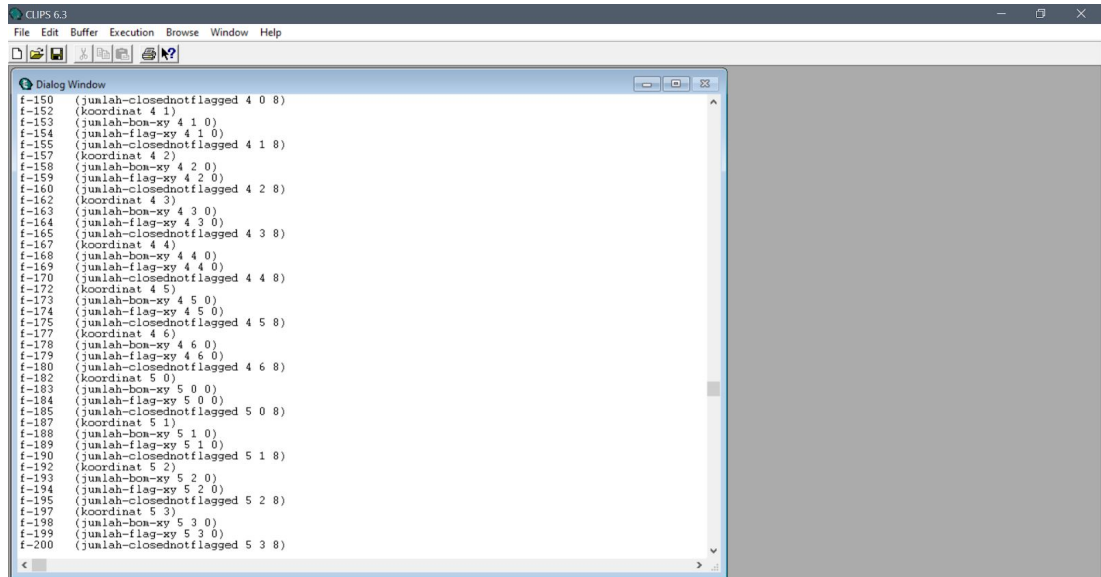
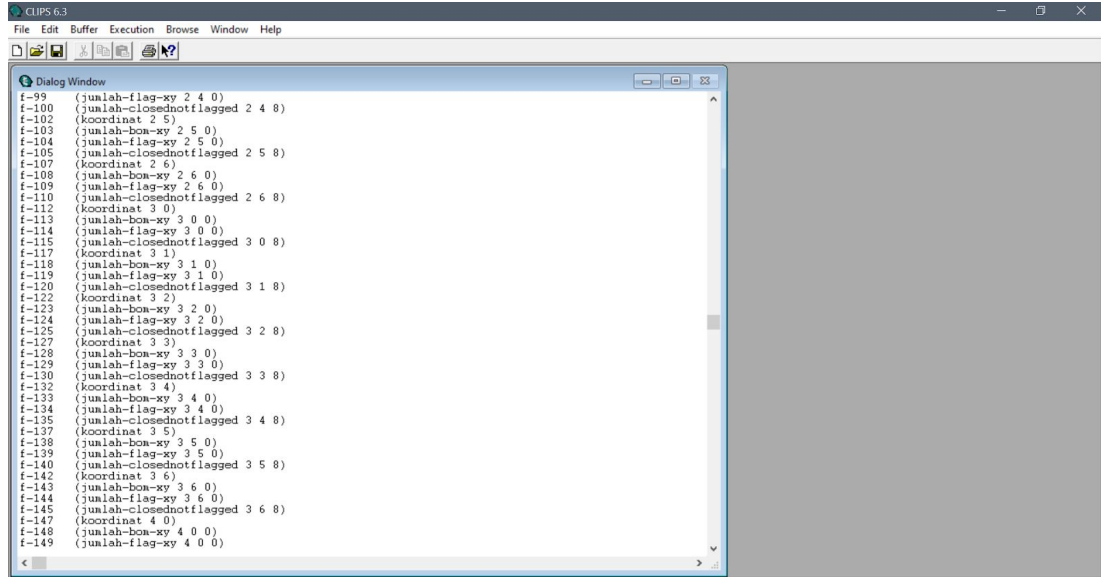
The screenshot shows the CLIPS 6.3 interface with the Dialog Window open. The window title is "CLIPS 6.3" and the menu bar includes File, Edit, Buffer, Execution, Browse, Window, and Help. The Dialog Window displays a list of facts from f-0 to f-47. The facts are as follows:

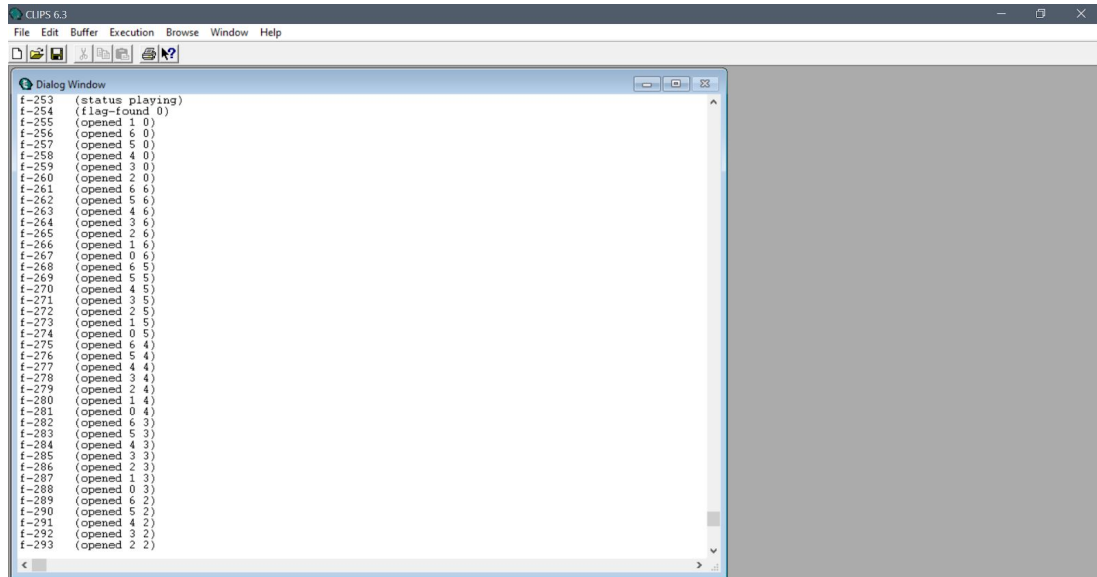
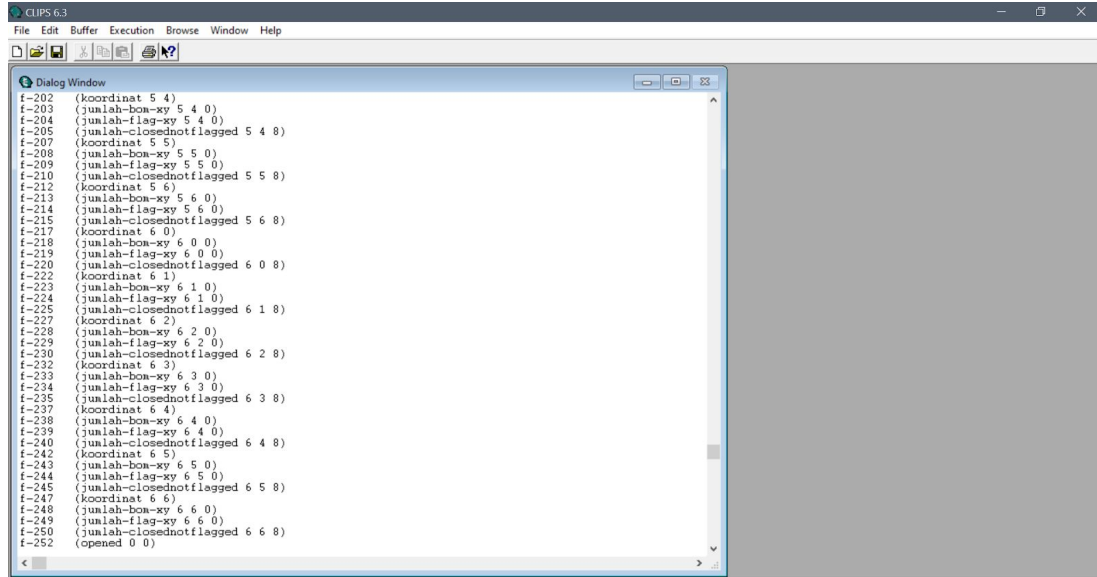
```
f-0 (initial-fact)
f-1 (encount-bomb 2)
f-2 (bomb 1 1)
f-3 (jumlah-bom-xy 1 1 -9999)
f-4 (bomb 1 5)
f-5 (jumlah-bom-xy 1 6 -9999)
f-6 (ukuran 6)
f-7 (koordinat 0 0)
f-8 (jumlah-bom-xy 0 0 0)
f-9 (jumlah-flag-xy 0 0 0)
f-10 (jumlah-closednotflagged 0 0 8)
f-12 (koordinat 0 1)
f-13 (jumlah-bom-xy 0 1 0)
f-14 (jumlah-flag-xy 0 1 0)
f-15 (jumlah-closednotflagged 0 1 8)
f-17 (koordinat 0 2)
f-18 (jumlah-bom-xy 0 2 0)
f-19 (jumlah-flag-xy 0 2 0)
f-20 (jumlah-closednotflagged 0 2 8)
f-22 (koordinat 0 3)
f-23 (jumlah-bom-xy 0 3 0)
f-24 (jumlah-flag-xy 0 3 0)
f-25 (jumlah-closednotflagged 0 3 8)
f-27 (koordinat 0 4)
f-28 (jumlah-bom-xy 0 4 0)
f-29 (jumlah-flag-xy 0 4 0)
f-30 (jumlah-closednotflagged 0 4 8)
f-32 (koordinat 0 5)
f-33 (jumlah-bom-xy 0 5 0)
f-34 (jumlah-flag-xy 0 5 0)
f-35 (jumlah-closednotflagged 0 5 8)
f-37 (koordinat 0 6)
f-38 (jumlah-bom-xy 0 6 0)
f-39 (jumlah-flag-xy 0 6 0)
f-40 (jumlah-closednotflagged 0 6 8)
f-42 (koordinat 1 0)
f-43 (jumlah-bom-xy 1 0 0)
f-44 (jumlah-flag-xy 1 0 0)
f-45 (jumlah-closednotflagged 1 0 8)
f-47 (koordinat 1 1)
```

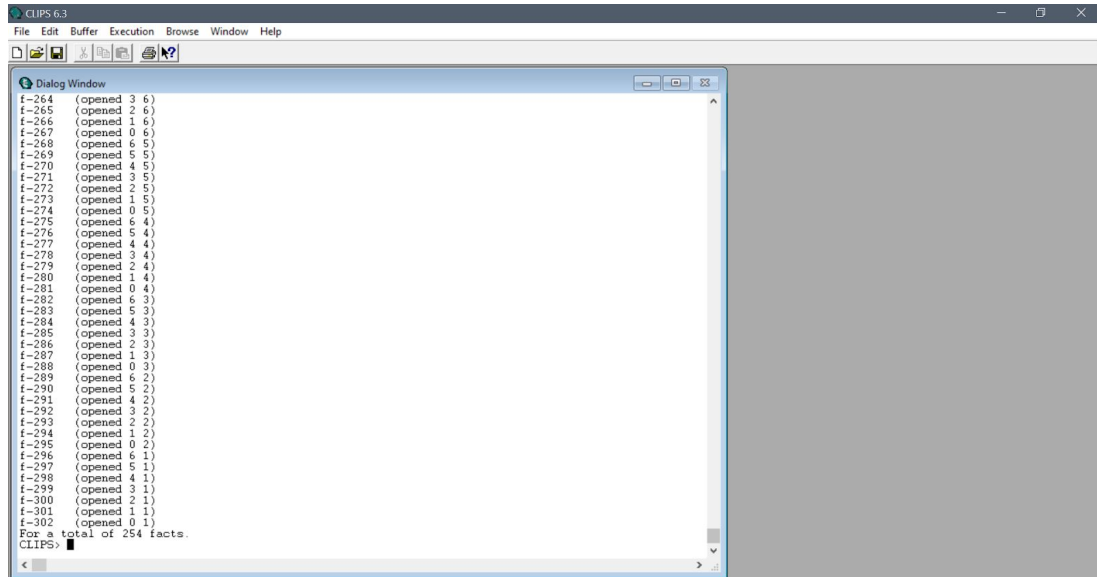


The screenshot shows the CLIPS 6.3 interface with the Dialog Window open. The window title is "CLIPS 6.3" and the menu bar includes File, Edit, Buffer, Execution, Browse, Window, and Help. The Dialog Window displays a list of facts from f-48 to f-98. The facts are as follows:

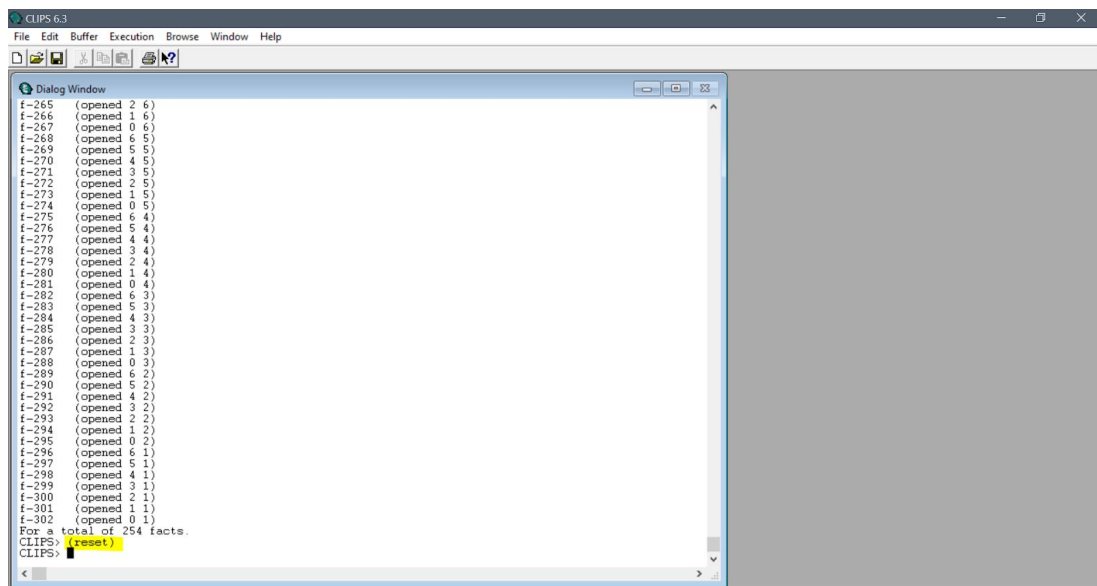
```
f-48 (jumlah-bom-xy 1 1 0)
f-49 (jumlah-flag-xy 1 1 0)
f-50 (jumlah-closednotflagged 1 1 8)
f-52 (koordinat 1 2)
f-53 (jumlah-bom-xy 1 2 0)
f-54 (jumlah-flag-xy 1 2 0)
f-55 (jumlah-closednotflagged 1 2 8)
f-57 (koordinat 1 3)
f-58 (jumlah-bom-xy 1 3 0)
f-59 (jumlah-flag-xy 1 3 0)
f-60 (jumlah-closednotflagged 1 3 8)
f-62 (koordinat 1 4)
f-63 (jumlah-bom-xy 1 4 0)
f-64 (jumlah-flag-xy 1 4 0)
f-65 (jumlah-closednotflagged 1 4 8)
f-67 (koordinat 1 5)
f-68 (jumlah-bom-xy 1 5 0)
f-69 (jumlah-flag-xy 1 5 0)
f-70 (jumlah-closednotflagged 1 5 8)
f-72 (koordinat 1 6)
f-73 (jumlah-bom-xy 1 6 0)
f-74 (jumlah-flag-xy 1 6 0)
f-75 (jumlah-closednotflagged 1 6 8)
f-77 (koordinat 2 0)
f-78 (jumlah-bom-xy 2 0 0)
f-79 (jumlah-flag-xy 2 0 0)
f-80 (jumlah-closednotflagged 2 0 8)
f-82 (koordinat 2 1)
f-83 (jumlah-bom-xy 2 1 0)
f-84 (jumlah-flag-xy 2 1 0)
f-85 (jumlah-closednotflagged 2 1 8)
f-87 (koordinat 2 2)
f-88 (jumlah-bom-xy 2 2 0)
f-89 (jumlah-flag-xy 2 2 0)
f-90 (jumlah-closednotflagged 2 2 8)
f-92 (koordinat 2 3)
f-93 (jumlah-bom-xy 2 3 0)
f-94 (jumlah-flag-xy 2 3 0)
f-95 (jumlah-closednotflagged 2 3 8)
f-97 (koordinat 2 4)
f-98 (jumlah-bom-xy 2 4 0)
```







5. Terakhir, gunakan *command* (reset) untuk menjalankan ulang program.



D. Proses Updating dan Inferencing

Updating data dilakukan dengan cara melakukan iterasi pada setiap elemen pada map pertama kali setelah koordinat bomb dimasukkan. Setelah itu akan diberikan nilai untuk setiap elemen tersebut dengan defrule make-board dengan dilakukan *increment* pada setiap elemen yang berada di sekitar bomb. Setelah semua elemen tersebut diupdate, maka akan dilakukan updating status untuk setiap elemen

yang dibuka dan diberi flag. Hal ini akan dilakukan berkali kali setiap setelah ada elemen yang dibuka dan diberi flag. Setelah itu, *updating* dilakukan untuk setiap jumlah bom.

Proses inferensi fakta pada program dimulai dengan menginisiasi fakta dari masukan program. Fakta yang dihasilkan setelah memasukkan ukuran board adalah ukuran board permainan. Selanjutnya fakta yang dihasilkan setelah memasukkan banyaknya bom yang akan diinisiasi adalah banyaknya bom yang akan diletakkan di dalam board permainan. Tahap berikutnya akan dihasilkan fakta terkait koordinat bom. Hasil generate board di awal program akan menghasilkan fakta koordinat board beserta nilai koordinat tersebut yang diinisiasi dengan nol.