Gabriele Cazzaniga Mat. 859738

Progetto Intelligenza Artificiale

SPECIFICA DEI TIPI

```
type [erba(number,number), bosco(number,number),
      roccia(number,number),erbetta_soffice(number,number)]: terreno.
type vita(number).
type [destra(number, number), sinistra(number, number),
      sopra(number, number), sotto(number, number)]: action.
type [pos_pecora(number, number), pos_recinto(number,number),
      visitata_erbetta(number,number)]: fluent.
```

SPECIFICA DEI PREDICATI

```
add_erbettaSoffice(list(terreno),number).
% add_erbettaSoffice(--T, ++E) det
% Spec: dato un numero E di caselle erbetta soffice da inserire, chiede all'utente
%       di inserirne una alla volta e le posiziona nella lista terreno T

add_and_del(number,number,list(stato),list(stato).
% add_and_del(++X,++Y,+S1,-S2) det
% Spec: dati due numeri X e Y e uno stato S1, sostituisce l'elemento
%       che si trova nella posizione corrispondente con una casella di erbetta
%       soffice, e restituisce lo stato così modificato

crea_mondo(number, number, list(terreno)).
% crea_mondo(++X, ++Y, --Mondo) nondet
% Spec: dati due numeri X e Y, Mondo è una griglia XxY ed ogni punto della
%       griglia è un tipo di terreno

calcolacosto(number,number,number).
% calcolacosto(++X, ++Y, --Costo) det
% Spec: dati due numeri X e Y, calcola il costo del tipo terreno che si trova in
%       quel punto

starting_state(list(fluent)).
% starting_state(--S) det
% Spec: S è lo stato iniziale
```

trovato(list(fluent)).
% trovato(--S) semidet
% Spec: fallisce a meno che S sia uno stato finale

add_del(action, list(stato),list(fluent),list(fluent),number).
% add_del(?Action, +S, ?Add, ?Del, ?Costo) semidet
% Spec: dato lo stato corrente calcola l'azione da intraprendere e il costo
% complessivo dell'azione (Costo terreno + Vita), Add e Del indicano
% cosa aggiungere e cosa togliere allo stato corrente

h(list(fluent),number).
% h(++S, --H) det
% Spec: dato uno stato S restituisce il valore euristico H per quello stato
% (5*(distanza in linea d'aria) +10/vita)

SPECIFICA DEI PREDICATI DI STAMPA

print_story(list(list(terreno)),fluent).
%print_story(++T,++X) det
%Spec: dato un insieme di liste di terreno T, stampa la griglia associata per ogni
% lista di terreno tenendo conto della posizione di X(recinto).

myprint(list(terreno)).
%myprint(++T) det
%Spec: data una lista di terreno T, stampa la griglia righe-colonne associata
% a quella lista

print_raw(list(terreno),number,number).
%print_raw(++T,++X,++Y) det
%Spec: data una lista di terreno T, una larghezza X e un'altezza Y, stampa gli
% elementi colonna associati ad ogni riga

print_column(list(terreno),number,number).
%print_column(++T,++X,++Y) det
%Spec: data una lista di terreno T e la posizione (X,Y) di ogni terreno della
% lista, stampa gli elementi di ogni colonna in base al tipo di terreno della
% posizione indagata