

## COMPITO DI APPLICAZIONI DI INTELLIGENZA ARTIFICIALE

6 Dicembre 2004 (Punteggio su 30/30; Tempo 2h)

### Esercizio 1 (punti 8)

Dato il seguente training set S:

A1	A2	Classe
False	?	Yes
True	A	No
False	C	Yes
False	A	Yes
True	C	No
False	C	No
True	A	Yes
False	B	No
True	B	No
False	C	Yes

- Si calcoli l'entropia del training set rispetto all'attributo Classe
- Si calcoli il gain ratio dei due attributi rispetto a questi esempi di training.
- si costruisca un albero decisionale ad un solo livello per il training set dato, indicando le etichette delle foglie (numero di esempi finiti nella foglia/numero di esempi finiti nella foglia non appartenenti alla classe della foglia).
- si classifichi l'istanza

False	C
-------	---

### Esercizio 2 (punti 8)

Un programma CLP deve compilare il piano degli studi di uno studente che si iscrive al primo anno di laurea specialistica. Il piano degli studi è costituito da 6 corsi (chiamati A, B, C, D, E, F), che devono essere distribuiti in due anni. Ciascuno degli anni è costituito da due semestri. Ogni esame viene tenuto in un semestre (o il primo o il secondo). Valgono inoltre le seguenti regole:

- Gli esami A, C, E sono tenuti al primo semestre, mentre B, D, F sono tenuti al secondo
- Il piano degli studi deve contenere al massimo 3 esami per ogni anno
- Non ci possono essere più di 2 esami per semestre
- L'esame A è propedeutico all'esame B, quindi deve essere seguito prima
- L'esame D è propedeutico all'esame E, quindi deve essere seguito prima
- Il corso C non verrà tenuto nel prossimo anno accademico, mentre il corso F non viene tenuto nell'A.A. corrente (verrà tenuto il prossimo A.A.).

Si scriva un programma CLP(FD) che risolve il problema, sapendo che il linguaggio fornito contiene il seguente vincolo:

atmost(+N, ?List, +V): At most N elements of the list List have the value V.

+N    An integer

?List    A list of domain variables or integers

+V    An integer

### Esercizio 3 (punti 8)

È dato lo stato iniziale descritto dalle seguenti formule atomiche:

**[ontable(b), on(a,b), ontable(c), clear(a), clear(c), handempty]**

(a,b, c rappresentano dei blocchi e si suppone ci siano infinite posizioni occupabili del tavolo

- le azioni sono modellate opportunamente come segue:

#### **pickup(X)**

PRECOND: ontable(X), clear(X), handempty

DELETE: ontable(X), clear(X), handempty

ADD: holding(X)

#### **putdown(X)**

PRECOND: holding(X)

DELETE: holding(X)

ADD: ontable(X), clear(X), handempty

#### **stack(X,Y)**

PRECOND: holding(X), clear(Y)

DELETE: holding(X), clear(Y)

ADD: handempty, on(X,Y), clear(X)

#### **unstack(X,Y)**

PRECOND: handempty, on(X,Y), clear(X)

DELETE: handempty, on(X,Y), clear(X)

ADD: holding(X), clear(Y)

e il goal **ontable(a) and on(b,c)**

si descriva come l'algoritmo lineare backward STRIPS trova un piano per questo goal. Si descriva lo stato e lo stack dei goal passo passo.

### Esercizio 4 (punti 6)

Discutere in che cosa si differenzia la pianificazione come ricerca nello spazio dei piani dalla pianificazione effettuata dall'algoritmo STRIPS-like.

## SOLUZIONE

### Esercizio 1:

a)  $\text{info}(S) = -5/10 \cdot \log_2 5/10 - 5/10 \cdot \log_2 5/10 = 1$

b)

$$\text{info}_{A1}(S) = 4/10 \cdot (-1/4 \cdot \log_2 1/4 - 3/4 \cdot \log_2 3/4) + 6/10 \cdot (-4/6 \cdot \log_2 4/6 - 2/6 \cdot \log_2 2/6)$$

$$= 0,4 \cdot 0,811 + 0,6 \cdot 0,918 = 0,875$$

$$\text{gain}(A1) = 1 - 0,875 = 0,125$$

$$\text{splitinfo}(A1) = -4/10 \cdot \log_2(4/10) - 6/10 \cdot \log_2(6/10) = 0,971$$

$$\text{gainratio}(A1) = 0,125 / 0,971 = 0,129$$

Per calcolare il guadagno dell'attributo A2 non si usa l'entropia calcolata su tutto il training set ma solo sugli esempi che hanno A2 noto (insieme F):

$$\text{info}(F) = -4/9 \cdot \log_2 4/9 - 5/9 \cdot \log_2 5/9 = 0,991$$

$$\text{info}_{A2}(F) = 3/9 \cdot (-2/3 \cdot \log_2 2/3 - 1/3 \cdot \log_2 1/3) + 2/9 \cdot (-2/2 \cdot \log_2 2/2 - 0/2 \cdot \log_2 0/2) + 4/9 \cdot (-2/4 \cdot \log_2 2/4 - 2/4 \cdot \log_2 2/4) =$$

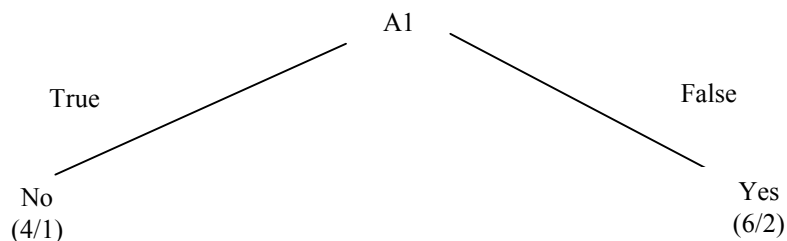
$$= 0,333 \cdot 0,918 + 0,222 \cdot 0 + 0,444 \cdot 1 = 0,750$$

$$\text{gain}(A2) = 9/10 \cdot (0,991 - 0,750) = 0,217$$

$$\text{splitinfo}(A2) = -3/10 \cdot \log_2(3/10) - 2/10 \cdot \log_2(2/10) - 4/10 \cdot \log_2(4/10) - 1/10 \cdot \log_2(1/10) = 1,846$$

$$\text{gainratio}(A2) = 0,217 / 1,846 = 0,118$$

c)



d) l'istanza viene classificata nella foglia di destra, quindi appartiene alla classe Yes con probabilità  $4/6 = 0,666$  e alla classe No con probabilità  $2/6 = 0,333$

### Esercizio 2

Per ciascun corso usiamo una variabile per indicare l'anno in cui viene messo: 0= primo anno, 1=secondo anno. Usiamo anche una variabile che rappresenta il semestre in cui lo studente segue l'esame (0=primo semestre del 1 anno, 1=secondo semestre del primo anno, 2=primo semestre del secondo anno, 3=secondo semestre del secondo anno).

pianostudio(LEsami) :-

```
% Lista degli anni in cui vengono seguiti gli esami
LEsami = [A, B, C, D, E, F],
LEsami :: 0..1,
% Lista dei semestri in cui vengono seguiti gli esami
LSemestri = [Ta, Tb, Tc, Td, Te, Tf],
% Collegamento fra gli esami ed i semestri
Ta #= A*2, Tb #= B*2+1,
Tc #= C*2, Td #= D*2+1,
Te #= E*2, Tf #= F*2+1,
```

```

Ta #< Tb,    % A è propedeutico a B
Td #< Te,    % D è propedeutico a E
C #= 0,      % C viene tenuto solo quest'anno
F #=1,       % F viene tenuto solo l'anno prossimo
  % Al massimo 2 esami per ogni semestre
atmost(2,LSemestri,0),
atmost(2,LSemestri,1),
atmost(2,LSemestri,2),
atmost(2,LSemestri,3),
  % Al massimo 3 esami all'anno.
atmost(3,LEsami,0),
atmost(3,LEsami,1),
labeling(LEsami).

```

### Esercizio 3

**Stato:**

```

ontable(b)
ontable(c)
on(a,b)
clear(a)
clear(c)
handempty

```

**Stack:**

```

ontable(a) and on(b,c)

```

Split del goal:

**Stato:**

```

ontable(b)
ontable(c)
on(a,b)
clear(a)
clear(c)
handempty

```

**Stack:**

```

ontable(a)
on(b,c)
ontable(a) and on(b,c)

```

Goal regression con l'azione **putdown(X)**:

**Stato:**

```

ontable(b)
on(a,b)
on(c,d)
clear(a)
clear(c)
handempty

```

**Stack:**

holding(a)

**putdown(a)**

on(b,c)

ontable(a) and on(b,c)

Goal regression con l'azione **unstack(X,Y):****Stato:**

ontable(b)

ontable(c)

on(a,b)

clear(a)

clear(c)

handempty

**Stack:**

handempty and on(a,b) and clear(a)

**unstack(a,b)****putdown(a)**

on(b,c)

ontable(a) and on(b,c)

Split del goal:

**Stato:**

ontable(b)

ontable(c)

on(a,b)

clear(a)

clear(c)

handempty

**Stack:**

handempty

on(a,b)

clear(a)

handempty and on(a,b) and clear(a)

**unstack(a,b)****putdown(a)**

on(b,c)

ontable(a) and on(b,c)

Tutti i primi 4 goal sono verificati nello stato corrente:

**Stato:**

ontable(b)

ontable(c)

on(a,b)

clear(a)

clear(c)

handempty

**Stack:**  
**unstack(a,b)**  
**putdown(a)**  
on(b,c)  
ontable(a) and on(b,c)

Esecuzione dell'azione unstack(a,b):

**Stato:**  
ontable(b)  
ontable(c)  
clear(c)  
holding(a)  
clear(b)

**Stack:**  
**putdown(a)**  
on(b,c)  
ontable(a) and on(b,c)

Esecuzione dell'azione putdown(a):

**Stato:**  
ontable(b)  
ontable(c)  
clear(c)  
clear(b)  
ontable(a)  
clear(a)  
handempty

**Stack:**  
on(b,c)  
ontable(a) and on(b,c)

Goal regression con l'azione **stack(X,Y)**:

**Stato:**  
ontable(b)  
ontable(c)  
clear(c)  
clear(b)  
ontable(a)  
clear(a)  
handempty

**Stack:**  
holding(b) and clear(c)  
**stack(b,c)**  
on(b,c)  
ontable(a) and on(b,c)

Split del goal:

**Stato:**  
ontable(b)  
ontable(c)  
clear(c)  
clear(b)  
ontable(a)  
clear(a)  
handempty

**Stack:**  
holding(b)  
clear(c)  
holding(b) and clear(c)  
**stack(b,c)**  
on(b,c)  
ontable(a) and on(b,c)

Goal regression con l'azione **pickup(X)**:

**Stato:**  
ontable(b)  
ontable(c)  
clear(c)  
clear(b)  
ontable(a)  
clear(a)  
handempty

**Stack:**  
ontable(b) and clear(b) and handempty  
**pickup(b)**  
clear(c)  
holding(b) and clear(c)  
**stack(b,c)**  
on(b,c)  
ontable(a) and on(b,c)

Split del goal, poi i primi 4 goal sono verificati nello stato corrente.

**Stato:**  
ontable(b)  
ontable(c)  
clear(c)  
clear(b)  
ontable(a)  
clear(a)  
handempty

**Stack:**  
**pickup(b)**

clear(c)  
holding(b) and clear(c)  
**stack(b,c)**  
on(b,c)  
ontable(a) and on(b,c)

Esecuzione di pickup(b):

**Stato:**

ontable(c)  
clear(c)  
clear(b)  
ontable(a)  
clear(a)  
holding(b)

**Stack:**

clear(c)  
holding(b) and clear(c)  
**stack(b,c)**  
on(b,c)  
ontable(a) and on(b,c)

Primi due goal soddisfatti nello stato corrente:

**Stato:**

ontable(c)  
clear(c)  
clear(b)  
ontable(a)  
clear(a)  
holding(b)

**Stack:**

**stack(b,c)**  
on(b,c)  
ontable(a) and on(b,c) and on(c,d)

Esecuzione di stack(b,c)

**Stato:**

ontable(c)  
on(b,c)  
clear(b)  
ontable(a)  
clear(a)  
handempty

**Stack:**

on(b,c)  
ontable(a) and on(b,c)

Tutti i goal nello stack sono soddisfatti nello stato corrente.