CSEP 573 Applications of Artificial Intelligence Winter 2011
Assignment 3

Due: Wednesday February 16, 6:30PM

Q 1: [9 points] The purpose of this question is to show that STRIPS is more expressive than it appears. You only have a planner that can handle STRIPS. This means no negative preconditions and no negative literals in the goal condition, and only conjunctive preconditions and conjunctive goals. How will you plan in domains that have the following characteristics?

q1.a: [3 points] [Negative Preconditions] Say an action a(?x) that has a precondition not p(?x).

q1.b: [3 points] [Disjunctive Preconditions] Say an action a(?x) that has a disjunctive precondition p1(?x) OR p2(?x).

q1.c: [3 points] [Disjunctive Goals] Say the goal condition is p1(?x) OR p2(?x).

If you suggest compilations to STRIPS, then discuss all rules to change actions, preconditions and effects.

Q 2: [16 points]

Encode the dynamics of a Fedex logistics domain in STRIPS. The domain is as follows: there are packages that need to be transported from one city to another city, and there are trucks that can pickup a package and transport to another city.

To encode the domain, first declare all the types. Then decide on a set of predicates and include the types of the arguments for each predicate. Finally, decide on a set of actions and code its preconditions and effects. Use the syntax in classical planning slides.

[Extra credit]: Download FF planner from http://www.loria.fr/~hoffmanj/ff/FF-v2.3.tgz and write a small logistic problem (initial state, goal condition), with a small problem size: say 3 cities, 2 trucks and 3 packages. Now see if FF can solve it. How about a slightly bigger problem, say 10 cities, 5 trucks and 20 packages?

Q 3: [10 points]

A contestant plays the game of 'find the gold coin' with the host of a show. The game happens as follows.

There are four boxes -- say marked A, B, C, and D. The gold coin is in exactly one of the boxes. The host knows where the gold coin is.

First the contestant selects a box, say A. The host then selects a box from the remaining three (say B) and opens it to show that the coin is not in the open box. At this point the contestant has the option to either stay with her originally chosen box or switch to any of the unopen boxes. This will be the contestant's final choice.

At this point, host opens the final selected box and the contestant wins if the coin is in the box. What should an ideal rational contestant do when asked for the final choice?

Should she stick to her original choice (A) or switch to one of the unopen boxes (C or D)? Does it matter? What is the expected reward in each alternative (assume coin is worth $1)?

Q 4: [15 points]

Exercise 14.6 parts (c, d, e, f) from R&N 3rd edition.

Q 5: [50 points] Programming Project. Your task is to learn the optimal policy for a BlackJack Player. Blackjack is the most popular casino card game. The basic objective is to have a hand with a value higher than the dealer's, but not over 21. The player can choose how she plays her hand, but the dealer's strategy for playing is fixed.

Description of the Programming Project

Terminology

cards:

a standard deck of playing cards is used, i.e., four suits (clubs, diamonds, spades, and hearts) and 13 different cards within each suit (the numbers 2 through 10, jack, queen, king, and ace). In this assignment, **we will replace 10, jack, queen and king** with a generic 'face card'. We will assume an infinite number of decks available in the pack.

card values:

the numbered cards (2 through 9) count as their numerical value. The generic face card (replacing 10, jack, queen, and king) counts as 10, and the ace may count as either 1 or 11 (whichever works in the player's favor).

hand value:

the value of a hand is the sum of the values of all cards in the hand. The values of the aces in a hand are such that they produce the highest value that is 21 or under (if possible). A hand where any ace is counted as 11 is called a soft hand. The suits of the cards do not matter in blackjack.

pair:

the two card hand where both cards have the same value. (example, two aces, a pair of sixes, and for our assignment, a pair of face cards).

blackjack:

is a two-card hand where one card is an ace and the other card is any face card.

busted:

the value of the hand has exceeded 21.

Rules of Play

There are some slight variations on the rules and procedure of blackjack. Below is **the simplified procedure that we will use for this assignment**. We will not be using insurance, surrender or dealer peeking, which are options in the standard casino game.

1. Each player places a bet on the hand.

2. The dealer deals two cards to each player, including himself. The player's cards will be face-up. One of the dealers cards is face-up, but the other is face-down.

3. If a player has blackjack, then that player wins immediately. The player is paid 1.5 times his or her bet.

4. If not, the player must do one of the following:

H - Hit: the player receives one additional card (face up). A player can receive as many cards as he or she wants, but if the value of the player's hand exceeds 21, the player busts and loses the bet on this hand irrespective of dealer's hand.

S - Stand: the player does not want any additional cards

D - Double-down: before the player has received any additional cards, she may double-down. This means that the player doubles her bet on the hand and will receive exactly one additional card. The disadvantage of doubling-down is that the player cannot receive any more cards (beyond the one additional card); the advantage is that the player can use this option to increase the bet when conditions are favorable.

P - sPlit: before the player has received any additional cards, if the original hand is a pair, then the player may split the hand. This means that instead of playing one hand the player will be playing two independent hands. She puts in the bet amount for the second hand. Two additional cards are drawn, one for each hand. The play goes on as if the player was playing two hands instead of one. If the drawn cards result in more pairs she is allowed to resplit. The player is allowed endless resplits in our version of the game. [There is an exception associated with a pair of Aces, see below]

5. Once the player stands, the dealer turns over his face-down card. The dealer then hits or stands according to the following deterministic policy:
If the value of the hand is less than 17 (or soft 17), the dealer must hit. Otherwise, the dealer must stand. This means, that the dealer stands if his Cards are (A,2,4) because that makes a soft 17. If the dealer busts, then he loses the bets with the non-busted players.

6. PayOffs (in this order)

(a) If the player had a blackjack she already received 1.5 times her bet.

(b) If the player busted, she lost her bet.

(c) If the dealer busted, he lost and the dealer pays the player an amount equal to her bet.

(d) If the value of dealer's hand is greater than player's the player loses her bet.

(e) If the value of player's hand is greater than dealer's the player won and dealer pays an amount equal to the bet.

(f) If the value of the two hands is equal, it is a push and the player gets back her bet money.

7. Other rules and exceptions.

(a) Doubling is allowed after split. That means, after splitting the pair, the player is allowed to double down either or both her hands if she wishes to.

(b) Player can resplit as many times as she desires (whenever allowed).

(c) Splitting Aces. This is an exception to the rule. If the player gets a pair of aces that is a very strong hand. She can split this but she will only get one additional card per split hand, and she will not be allowed to resplit. Moreover, if the card is a face card, it will not be counted as blackjack, and will be treated as a regular 21.

To familiarize yourself with the game, play it online for a few minutes. There are many online applets available, for example, http://javaboutique.internet.com/blackjack/. The rules they follow may be slight variations of those used in the assignment, but you will get the general idea.

In this assignment your task is to learn the policy for the optimal player. As usual the first step is to carefully think of the state space that you will need. To give you a headstart we propose that the state space contains the following fields:

1. MinSum: the minimum sum of the player's hand so far.
2. NumAces: the number of Aces in the player's hand so far.
3. dMinSum: the minimum sum of the dealer's hand so far.
4. dNumAces: the number of Aces in the dealer's hand so far.
5. isTwoCards: a Boolean that represents that the player's hand has just two cards so far.
6. isBlackJack: a Boolean that represents that the player got a BlackJack.
7. pair: has value between 0 and 10. Zero indicates that the player doesn't have a pair. Any other value i indicates that the player has two cards of value i.
8. turn: a Boolean to indicate whether it is player's turn or dealer's.

Take a few minutes and convince yourself that the aforementioned representation is enough to model the different relevant states of the game. If you feel it is not, please let us know (or feel free to change it to suit your convenience).

Design a state transition function, and reward model to encode the dynamics of the play. Solve the game to compute the best play in each state. The best play is defined as the action (hit/stand/double/split) that maximizes the expected return. Make sure you double or split only in the states it is allowed. Assume that the player bets $1.

Program for a `BlackJack(p)` game. Assume that the probability of getting a face card is p (an input to the program) and the probability of getting all other cards, 2-9 and Ace, is uniformly (1-p)/9. Note that p = 4/13 captures the standard Blackjack game.

After you solve the problem, the solution to BlackJack(4/13) should look very close to this. In the first column, representing your hand, a single integer represents the sum of the two cards, and indicates that they are not a pair and that neither is a face card. For the actions, you do not need to distinguish between "D" and "DS"

Grading for the Programming Project

[30 points]

Your description of how you modeled the BlackJack game.

Describe how you modeled the dynamics and rewards of problem. Include a discussion on how you handled double down and split. Splitting can be a bit tricky -- if confused send us email.

Include the best policy (in the same format as on the blackjackinfo website) that your program computed. Do this for blackJack(4/13) at least. Also discuss how you computed the expected long term reward of a player playing this game repeatedly. Also compute the average long term reward a player will get if they play this game repeatedly. and discuss how you computed it. Assume that each bet is $1.

[20 points]

Your code as tested by varying the values of p. For example:

./run.sh 0.307
...
./runsh 0.4

Output should be in rows with a a format like:

A,2 H H D D D H H H H H

Use a tab as field separator between the "hand" field and the action fields. Use a space as field separator between the actions. Do not give a header or footer row, or any other extraneous output. Make sure you have the correct number of rows and colums.

What to Turn In

Use the Class Dropbox to turn in a single file: "a3submit.zip". This should contain the a3submit directory with a single .pdf file called assignment3Report.pdf and all of your code.

Your code should include at minimum a `run.sh` file, and a `compile.sh` file if necessary. Do not include any pre-compiled files. Your code should compile and run in the attu.cs.washington.edu environment. (Note that attu uses Python version 2.6.4 and java version 1.6 - newer features might not work)