CISC 204 Group 3 Presents:

Aidan Sibley, Ethan Fighel, Will Wu, Oliver Dantzer

Rules of Yaniv:

The game is played with a standard deck of 52 cards.

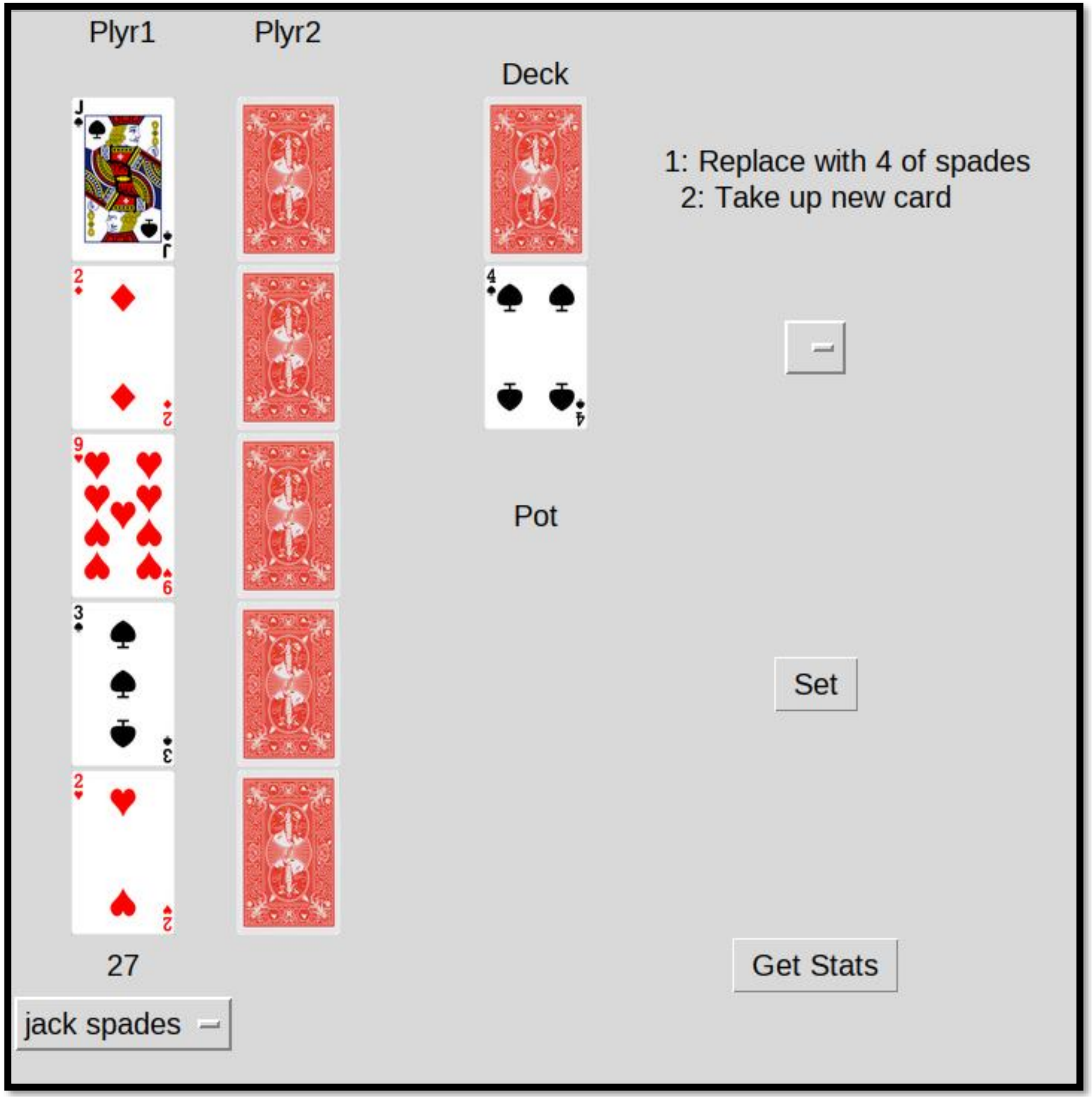
Each card represents a digit from 1-13, where an ace would represent 1 and a king would represent 13.

Each player of the game is dealt 5 cards to begin. The remaining cards are placed in a pile face down.

To begin the game a single card is faced upwards creating a new pile. This will be known as the "pile care" or "face-up card".

The goal of the game is to be the first person with cards that sum to 7 or less.

To do this players on their turn must replace their cards.



Sample game layout

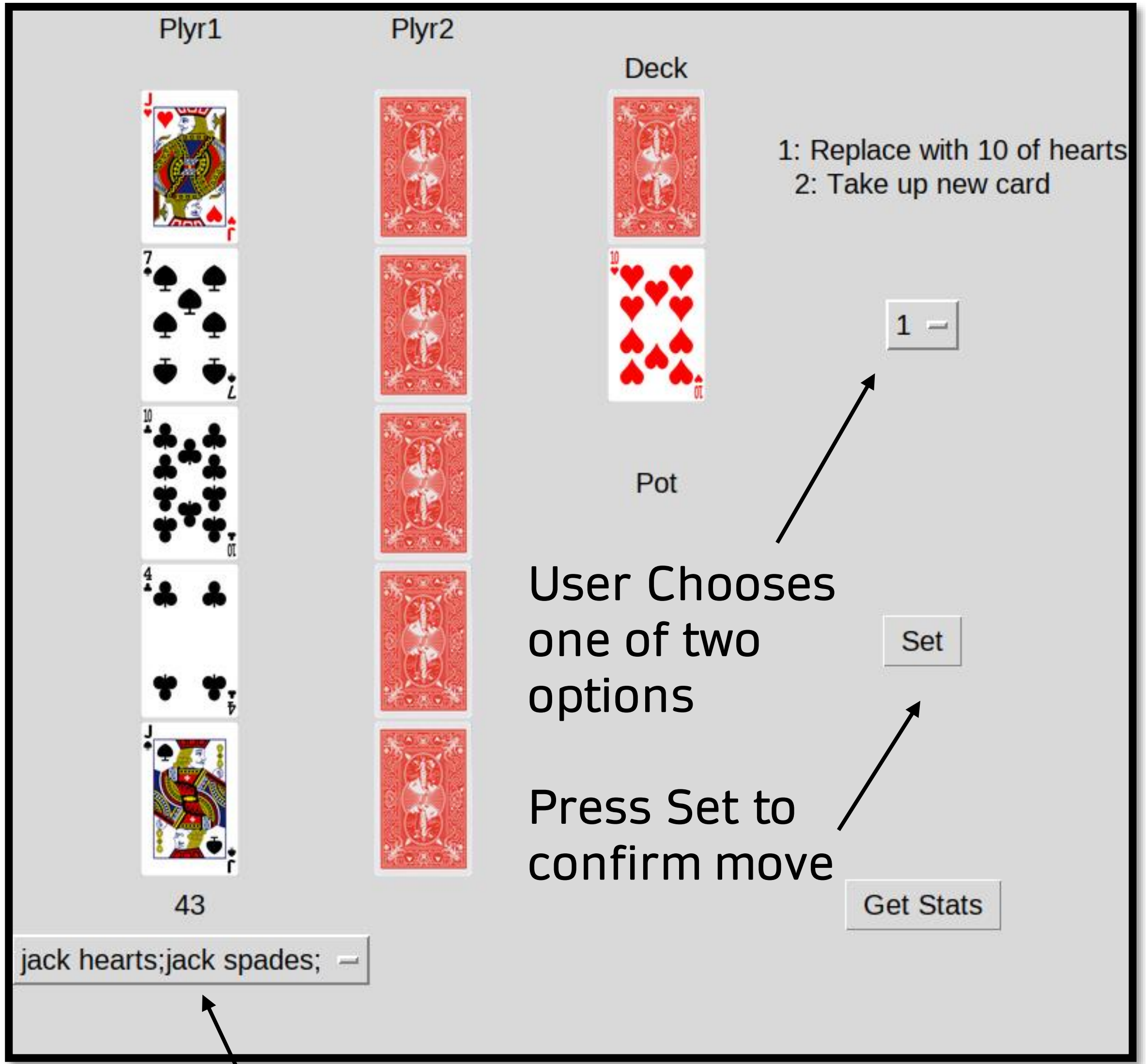
# of Yaniv: Continued

Each player's hand can be split into one of 2 groups:

1. Cards in a sequence containing the same suit
2. Same card in different suits.

A single card or a group of cards satisfying these requirements can be replaced with either the existing known flipped card already present, or a new card can be taken from the deck.

To reiterate, the user has two choices to replace a single card or group of cards (as previously outlined). Note that the player will only substitute a group for one card back, so it is possible to have a hand less than



User Choosing Group from Drop Down menu

Our Problem Statement:

The most efficient move in Yaniv

To be more specific, group3's goal is to find using logic and by extension the rules of natural deduction to determine the most productive more given the current state of the game.

Propositions and Constraints

The following slides will show the initial propositions/constraints alongside the current propositions/constraint to highlight the progress made.

Propositions:
h_n: This is **true** if card with value *n* is highest in hand
m2_n: This is **true** is if card with value *n* has one card in hand with the same value
e.g m2_5 is **true** if there are two 5's in a hand
m3_n: same as prev but for triples
m4_n: same as prev but for quadruples

sm2_n: This is true if card with value n has a card with the same n at the top of the pile
e.g if there's a 9 in the pile of cards put down and there's a 9 in your hand then s_n is true
sm3_n: same as prev but with triples
sm4_n: same as prev but with triples
s3fl_n: Same as prev ones but with straight flush of length 3
s4fl_n: Same as prev with length 4
s5fl_n: Same as prev length 5

w_n: this is true if sum of the hand is 7 or less

3fl_n_s: this is true if a card with value n and suit s is surrounded by two consecutive cards with the same s
e.g 3,4,5 of clubs means that 3fl_n_s where 4 is the n
4fl_n_s: same as prev but with three others
5fl_n_s: same as prev but with four others

e_{ns} = true if a card in hand has value n and suit s
k_i = true if hand has an i number of cards.
p_{ns} = true if piled card is value n and suit s
w = true if the sum of n's for all e_{ns} ¡= 7.
$m_{nq} = e_{ns}$ has q cards with same value n in hand (group type 1 a)
$sm_{nq} = p_{ns}$ has q cards with same value n that are in hand (group type 1 b)
$f_{sc} = e_{ns}$ has c consecutive cards with same s in hand (group type 2 a)
$sf_{sc} = p_{ns}$ has c consecutive cards with same s in hand (group type 2 b)
g_d = true if a group in hand has sum of n equal to d. e.g 2 hearts, 3 hearts, 4 hearts is a group with sum of n equal to 9.
d_g = true if g_d (or just g denoting this specific group) is the highest group in terms of d in the hand.
sg_d = true if a group in hand+pile has sum of n equal to d.
sd_g = true if sg_d is the highest group in terms of sum d in hand+pile.
c = true when an d_g and a sd_g have at least one e_{ns} in common.
(e.g 2 hearts, 3 hearts, 4 hearts is sd_g and 4 hearts is in hand, and then 4 hearts, 4 diamonds, and 4 spades is d_g)
vd = true when sd_g has higher sum d than d_g.
$rx = d_g$ is put down on top of pile.
ry = best under d_g in hand is put down on top of pile.
us = true if pile card is picked up (option 1)
ud = true if random card from deck is picked up (option 2)
$l = p_{ns}$ less than 7

Constraints:
5fl_n_s s5fl_ns,
4fl_n_s m4_n s4fl_ns sm4_n,
3fl_n_s m3_n s3fl_n_s sm3_n,
m2_n sm2_n,
h_n in descending order is the order of what takes precedence in deciding a move.

if w_n is true, you must say yaniv.

Card constraints for a 52 card deck:

$1 \leq i \leq 5$, so the number of cards in hand is in between 1-5.
$1 \leq n \leq 13$, so n can only be ace, 1, 2,...,10, jack, queen, king.
$1 \leq s \leq 4$, so n can only be diamonds = 1, clubs = 2, hearts = 3, spades = 4.

Every card has a unique suit and value combination assigned to it: e.g there can only be one e_{71}.

Yaniv Game constraints using a 52 card deck:
$1 \leq q \leq 3$, so the number of cards with the same n as a given e_{ns} or p_{ns} is limited to 1, 2 or 3 others.
$2 \leq c \leq 4$, so the number of cards consecutive to a given e_{ns} or p_{ns} with the same s is limited to 2, 3, or 4 others.
$g_d <-> (m_{nq} \wedge f_{sc} \wedge e_{ns} \wedge m_{nq} \wedge f_{sc} \wedge e_{ns} \wedge m_{nq} \wedge f_{sc} \wedge f_{sc} \wedge e_{ns} \wedge m_{nq} \wedge e_{ns})$ (these three things are just groups, categorized because they have different constraints placed on them, similar to one below)
$sg_d <-> (sm_{nq} \vee sf_{sc} \wedge sm_{nq} \wedge sf_{sc})$
$d_g \rightarrow g_d$ (in order for g_d to be the highest, g_d needs to exist, similar to one below)
$sd_g \rightarrow sg_d$
$w \rightarrow e_{71} \vee e_{72} \vee ... \vee e_{74} \vee (e_{51} \wedge e_{11}) \vee$ ... or any combination of cards in hand that adds up to a sum of n ≤ 7.
$c \rightarrow sd_g \wedge d_g$
$ry \longleftrightarrow c \wedge vd$
$rx \longleftrightarrow !c$
$l \vee ry \wedge g_d \wedge d_g \wedge rx \wedge sg_d \wedge sd_g) -> us$
$!l \longleftrightarrow ud$

Propositions:

Original Propositions (Oct 16th, 2022)

Propositions:
h_n: This is **true** if card with value *n* is highest in hand
m2_n: This is **true** is if card with value *n* has one card in hand with the same value
e.g m2_5 is **true** if there are two 5's in a hand
m3_n: same as prev but for triples
m4_n: same as prev but for quadruples

sm2_n: This is true if card with value n has a card with the same n at the top of the pile
e.g if there's a 9 in the pile of cards put down and there's a 9 in your hand then s_n is true
sm3_n: same as prev but with triples
sm4_n: same as prev but with triples
s3fl_n: Same as prev ones but with straight flush of length 3
s4fl_n: Same as prev with length 4
s5fl_n: Same as prev length 5

w_n: this is true if sum of the hand is 7 or less

3fl_n_s: this is true if a card with value n and suit s is surrounded by two consecutive cards with the same s
e.g 3,4,5 of clubs means that 3fl_n_s where 4 is the n
4fl_n_s: same as prev but with three others
5fl_n_s: same as prev but with four others

- Initially, to define the propositions for the groups of cards, I distinguished between doubles, triples, quadruples, etc. By individually assigning variables to their truth values.
- The highest card truth value was only assigned to single cards (that value for groups was not yet defined).
- The concept for hand + pile groups and just hand groups was in mind from the beginning, but the interaction between these two types was not explored.
- The truth values for which card is picked up is not explored either.

Revised/Final Propositions:

- In this revision, the truth values for the groups of cards became more compact using subscripts.
- The truth value for whether something was a group in general was brought in and the h_n was removed entirely. Then those groups could be compared by sum.
- The idea of overlap between a hand + pile group and just hand group was introduced.
- This idea of overlap ended up impacting which group in hand would be put down, which is explored in rx and ry.
- Then the truth value for whether the pile card was less than 7 was codified, as it determines whether you pick up a pile card or not in most cases.
- This led to us and ud being made to show which card was picked up
- There was no initialization of the pile card in the previous set of propositions, or whether a hand has a given number of cards.

e_{ns} = true if a card in hand has value n and suit s
k_i = true if hand has an i number of cards.
p_{ns} = true if piled card is value n and suit s
w = true if the sum of n's for all e_{ns} ¡= 7.
$m_{nq} = e_{ns}$ has q cards with same value n in hand (group type 1 a)
$sm_{nq} = p_{ns}$ has q cards with same value n that are in hand (group type 1 b)
$f_{sc} = e_{ns}$ has c consecutive cards with same s in hand (group type 2 a)
$sf_{sc} = p_{ns}$ has c consecutive cards with same s in hand (group type 2 b)
g_d = true if a group in hand has sum of n equal to d. e.g 2 hearts, 3 hearts, 4 hearts is a group with sum of n equal to 9.
d_g = true if g_d (or just g denoting this specific group) is the highest group in terms of d in the hand.
sg_d = true if a group in hand+pile has sum of n equal to d.
sd_g = true if sg_d is the highest group in terms of sum d in hand+pile.
c = true when an d_g and a sd_g have at least one e_{ns} in common.
(e.g 2 hearts, 3 hearts, 4 hearts is sd_g and 4 hearts is in hand, and then 4 hearts, 4 diamonds, and 4 spades is d_g)
vd = true when sd_g has higher sum d than d_g.
$rx = d_g$ is put down on top of pile.
ry = best under d_g in hand is put down on top of pile.
us = true if pile card is picked up (option 1)
ud = true if random card from deck is picked up (option 2)
$l = p_{ns}$ less than 7

Constraints:
5fl_n_s s5fl_ns,
4fl_n_s m4_n s4fl_ns sm4_n,
3fl_n_s m3_n s3fl_n_s sm3_n,
m2_n sm2_n,
h_n in descending order is the order of what takes precedence in deciding a move.

if w_n is true, you must say yaniv.

Revised/Final Constraints

- Here, the constraints are labelled into two separate sections: one for modelling individual cards in a standard 52 card deck, and one for modelling an arbitrary game of Yaniv.

Card constraints:

- Here, properties of individual cards (I.e., number, suit, position in hand) have proper guidelines.
- As well, it states cards must be separate/individual.

Game constraints:

- Here, different combinations are clearly labelled out, as well as the precedence hierarchy.
- Other relationships between propositions are listed. E.g., The constraints tell us what combo to put down, given the hands in a deck and the combo hierarchy (we don't always put down the immediate best combo)

Card constraints for a 52 card deck:

$1 \leq i \leq 5$, so the number of cards in hand is in between 1-5.
$1 \leq n \leq 13$, so n can only be ace, 1, 2,...,10, jack, queen, king.
$1 \leq s \leq 4$, so n can only be diamonds = 1, clubs = 2, hearts = 3, spades = 4.

Every card has a unique suit and value combination assigned to it: e.g there can only be one e_{71}.

Yaniv Game constraints using a 52 card deck:
$1 \leq q \leq 3$, so the number of cards with the same n as a given e_{ns} or p_{ns} is limited to 1, 2 or 3 others.
$2 \leq c \leq 4$, so the number of cards consecutive to a given e_{ns} or p_{ns} with the same s is limited to 2, 3, or 4 others.
$g_d <-> (m_{nq} \wedge f_{sc} \wedge e_{ns} \wedge m_{nq} \wedge f_{sc} \wedge e_{ns} \wedge m_{nq} \wedge f_{sc} \wedge f_{sc} \wedge e_{ns} \wedge m_{nq} \wedge e_{ns})$
(these three things are just groups, categorized because they have different constraints placed on them, similar to one below)
$sg_d <-> (sm_{nq} \vee sf_{sc} \wedge sm_{nq} \wedge sf_{sc})$
$d_g \rightarrow g_d$ (in order for g_d to be the highest, g_d needs to exist, similar to one below)
$sd_g \rightarrow sg_d$
$w \rightarrow e_{71} \vee e_{72} \vee ... \vee e_{74} \vee (e_{51} \wedge e_{11}) \vee$ ... or any combination of cards in hand that adds up to a sum of $n \leq 7$.
$c \rightarrow sd_g \wedge d_g$
$ry \longleftrightarrow c \wedge vd$
$rx \longleftrightarrow !c$
$l \vee ry \wedge g_d \wedge d_g \wedge rx \wedge sg_d \wedge sd_g) -> us$
$!l \longleftrightarrow ud$

Legend for Jape Proofs

A1-A5: Individual cards in a hand

B111-B110: Double combo (multiples) within hand cards
B121-B125: Double combo (multiples) with pile cand and hand card

C111-C110: Triple combo (multiples) within hand cards
C121-C120: Triple combo (multiples) with pile card and hand cards
C211-C210: Triple combo (straight) within hand cards
C221-C220: Triple combo (straight) with pile card and hand card

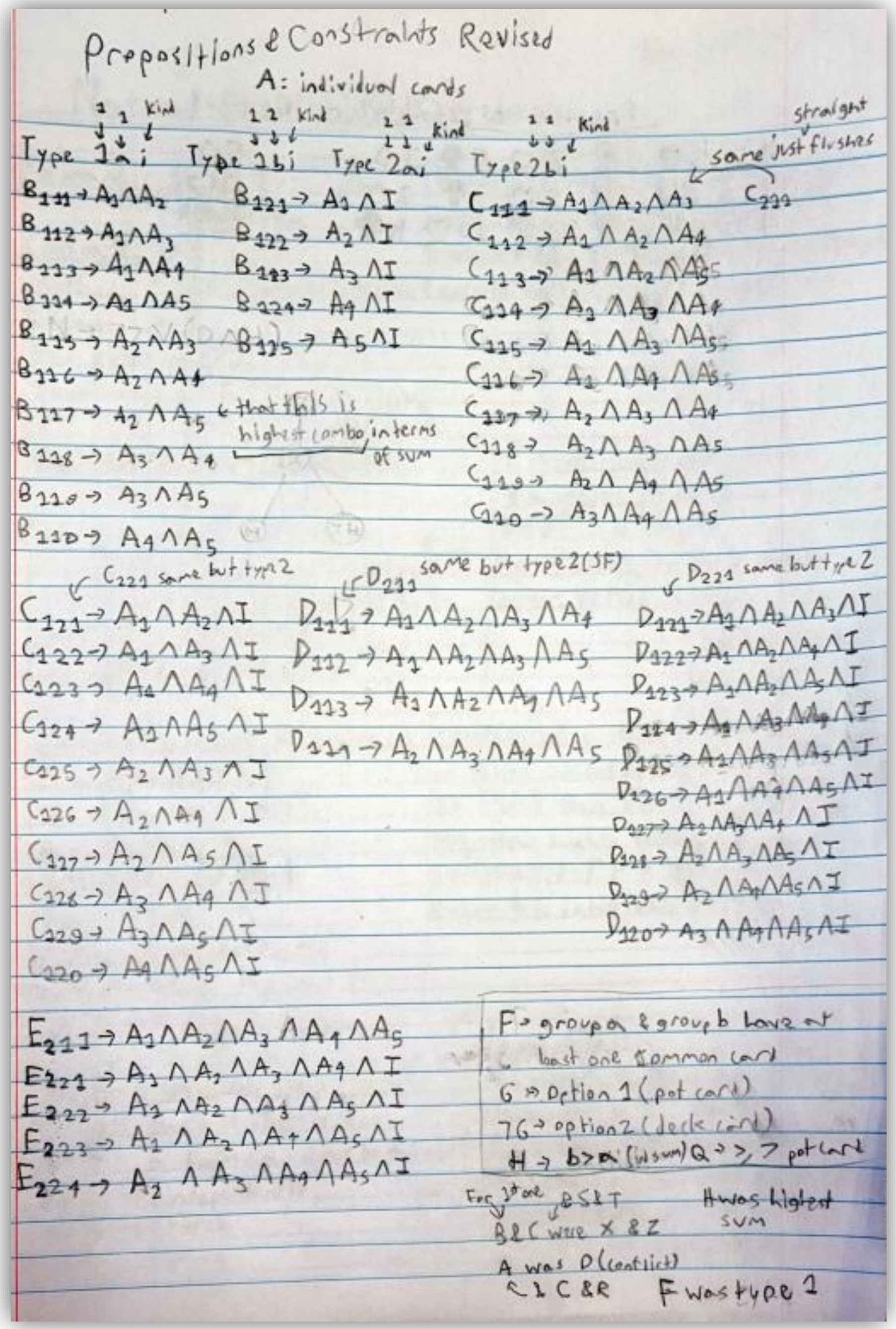
D111-D114: Quadruple combo (multiples) within hand cards
D121-D120: Quadruple combo (multiples) with pile card and hand cards
D211-D214: Quadruple combo (straight) within hand cards
D221-D220: Quadruple combo (straight) with pile card and hand cards

E211: Quintuple combo (straight) within hand cards
E221-E224: Quintuple combo (straight) with pile card and hand cards

F: There is overlap/conflict between combination within hand cards, and combination involving pile card and hand cards

G: Pick up pile card

H: Combo with pile card is greater than combo within hand cards



Prepositions & Constraints Revised

A: individual cards

Type 1ai | Type 1bi | Type 2ai | Type 2bi | straight; same just flushes; C_{211}

$B_{111} \to A_1 \wedge A_2$ | $B_{121} \to A_1 \wedge I$ | $C_{111} \to A_1 \wedge A_2 \wedge A_3$
$B_{112} \to A_1 \wedge A_3$ | $B_{122} \to A_2 \wedge I$ | $C_{112} \to A_1 \wedge A_2 \wedge A_4$
$B_{113} \to A_1 \wedge A_4$ | $B_{123} \to A_3 \wedge I$ | $C_{113} \to A_1 \wedge A_2 \wedge A_5$
$B_{114} \to A_1 \wedge A_5$ | $B_{124} \to A_4 \wedge I$ | $C_{114} \to A_1 \wedge A_3 \wedge A_4$
$B_{115} \to A_2 \wedge A_3$ | $B_{125} \to A_5 \wedge I$ | $C_{115} \to A_1 \wedge A_3 \wedge A_5$
$B_{116} \to A_2 \wedge A_4$ | $C_{116} \to A_1 \wedge A_4 \wedge A_5$
$B_{117} \to A_2 \wedge A_5$ ← that this is highest combo in terms of sum | $C_{117} \to A_2 \wedge A_3 \wedge A_4$
$B_{118} \to A_3 \wedge A_4$ | $C_{118} \to A_2 \wedge A_3 \wedge A_5$
$B_{119} \to A_3 \wedge A_5$ | $C_{119} \to A_2 \wedge A_4 \wedge A_5$
$B_{110} \to A_4 \wedge A_5$ | $C_{110} \to A_3 \wedge A_4 \wedge A_5$

C_{221} same but type 2 | D_{211} same but type 2 (SF) | D_{221} same but type 2

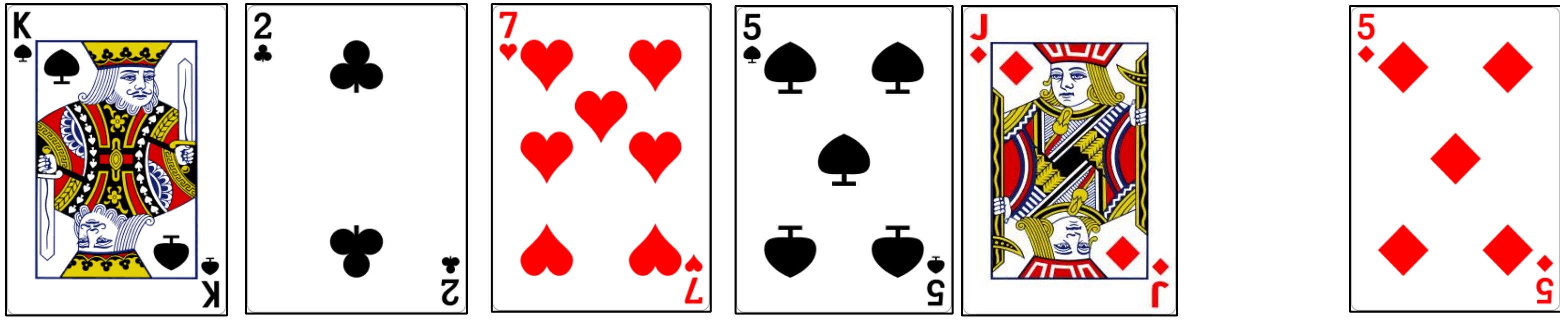
$C_{121} \to A_1 \wedge A_2 \wedge I$ | $D_{111} \to A_1 \wedge A_2 \wedge A_3 \wedge A_4$ | $D_{121} \to A_1 \wedge A_2 \wedge A_3 \wedge I$
$C_{122} \to A_1 \wedge A_3 \wedge I$ | $D_{112} \to A_1 \wedge A_2 \wedge A_3 \wedge A_5$ | $D_{122} \to A_1 \wedge A_2 \wedge A_4 \wedge I$
$C_{123} \to A_1 \wedge A_4 \wedge I$ | $D_{113} \to A_1 \wedge A_2 \wedge A_4 \wedge A_5$ | $D_{123} \to A_1 \wedge A_2 \wedge A_5 \wedge I$
$C_{124} \to A_1 \wedge A_5 \wedge I$ | $D_{114} \to A_2 \wedge A_3 \wedge A_4 \wedge A_5$ | $D_{124} \to A_1 \wedge A_3 \wedge A_4 \wedge I$
$C_{125} \to A_2 \wedge A_3 \wedge I$ | $D_{125} \to A_1 \wedge A_3 \wedge A_5 \wedge I$
$C_{126} \to A_2 \wedge A_4 \wedge I$ | $D_{126} \to A_1 \wedge A_4 \wedge A_5 \wedge I$
$C_{127} \to A_2 \wedge A_5 \wedge I$ | $D_{127} \to A_2 \wedge A_3 \wedge A_4 \wedge I$
$C_{128} \to A_3 \wedge A_4 \wedge I$ | $D_{128} \to A_2 \wedge A_3 \wedge A_5 \wedge I$
$C_{129} \to A_3 \wedge A_5 \wedge I$ | $D_{129} \to A_2 \wedge A_4 \wedge A_5 \wedge I$
$C_{120} \to A_4 \wedge A_5 \wedge I$ | $D_{120} \to A_3 \wedge A_4 \wedge A_5 \wedge I$

$E_{211} \to A_1 \wedge A_2 \wedge A_3 \wedge A_4 \wedge A_5$
$E_{221} \to A_1 \wedge A_2 \wedge A_3 \wedge A_4 \wedge I$
$E_{222} \to A_1 \wedge A_2 \wedge A_3 \wedge A_5 \wedge I$
$E_{223} \to A_1 \wedge A_2 \wedge A_4 \wedge A_5 \wedge I$
$E_{224} \to A_2 \wedge A_3 \wedge A_4 \wedge A_5 \wedge I$

F → group a & group b have at least one common card
G → option 1 (pot card)
¬G → option 2 (deck card)
H → b > a (w sum) Q → >, 7 pot card

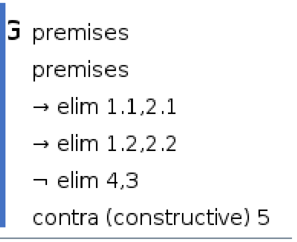
For 1st one, B S & T; B & C were X & Z; A was D (conflict); ¬1 C & R
H was highest sum
F was type 1

Jape Proof #1



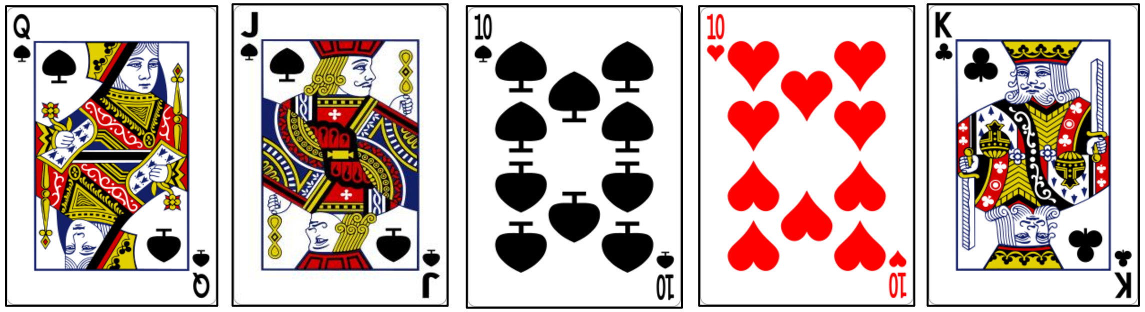
Given hand: king spades (A1), 2 clubs (A2), 7 hearts (A3), 5 clubs (A4), jack diamonds (A5) - 5 diamonds as pile card (I)

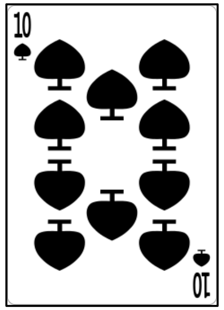
There is a possible double combination with A4 and pile card: B124 is true



Here, we prove that the right move to take is to put down A1 and pick up the pile card (rather than the unknown deck card).

Jape Proof #2





Given there is a Queen of spades (A1), Jack of spades (A2), 10 of spades (A3), 10 of hearts (A4), K of clubs (A5) in hand and a 10 of clubs as pile card.

Given one in set of combinations is highest implies we have an overlap between some of them.

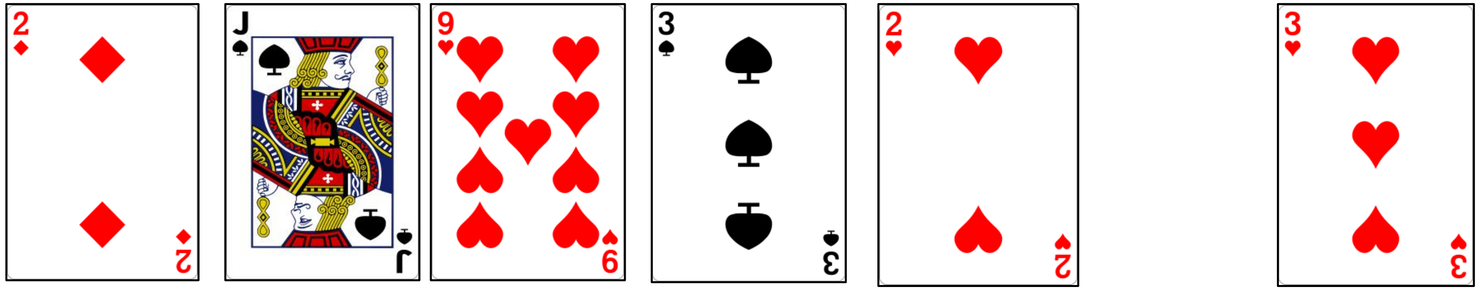
Given there is an overlap between some combos, pile card greater than or equal to 7, and group a >= group b implies deck card.

Finally, given there is overlap between some combos, pile card greater than or equal to 7, and group a >= group b is true.

We conclude that combo C211 being the highest implies that deck card is picked up.

1:	$(C211 \lor B118 \lor C128 \lor A1 \lor A2 \lor A3 \lor A4 \lor A5) \to F$, $(\neg H \land Q \land F) \to \neg G$, $(\neg H \land Q \land F)$	premises
2:	$\neg G$	→ elim 1.2,1.3
3:	$C211$	assumption
4:	$C211 \lor B118$	∨ intro 3
5:	$C211 \lor B118 \lor C128$	∨ intro 4
6:	$C211 \lor B118 \lor C128 \lor A1$	∨ intro 5
7:	$C211 \lor B118 \lor C128 \lor A1 \lor A2$	∨ intro 6
8:	$C211 \lor B118 \lor C128 \lor A1 \lor A2 \lor A3$	∨ intro 7
9:	$C211 \lor B118 \lor C128 \lor A1 \lor A2 \lor A3 \lor A4$	∨ intro 8
10:	$C211 \lor B118 \lor C128 \lor A1 \lor A2 \lor A3 \lor A4 \lor A5$	∨ intro 9
11:	F	→ elim 1.1,10
12:	$\neg G$	hyp 2
13:	$C211 \to \neg G$	→ intro 3-12

Jape Proof #3



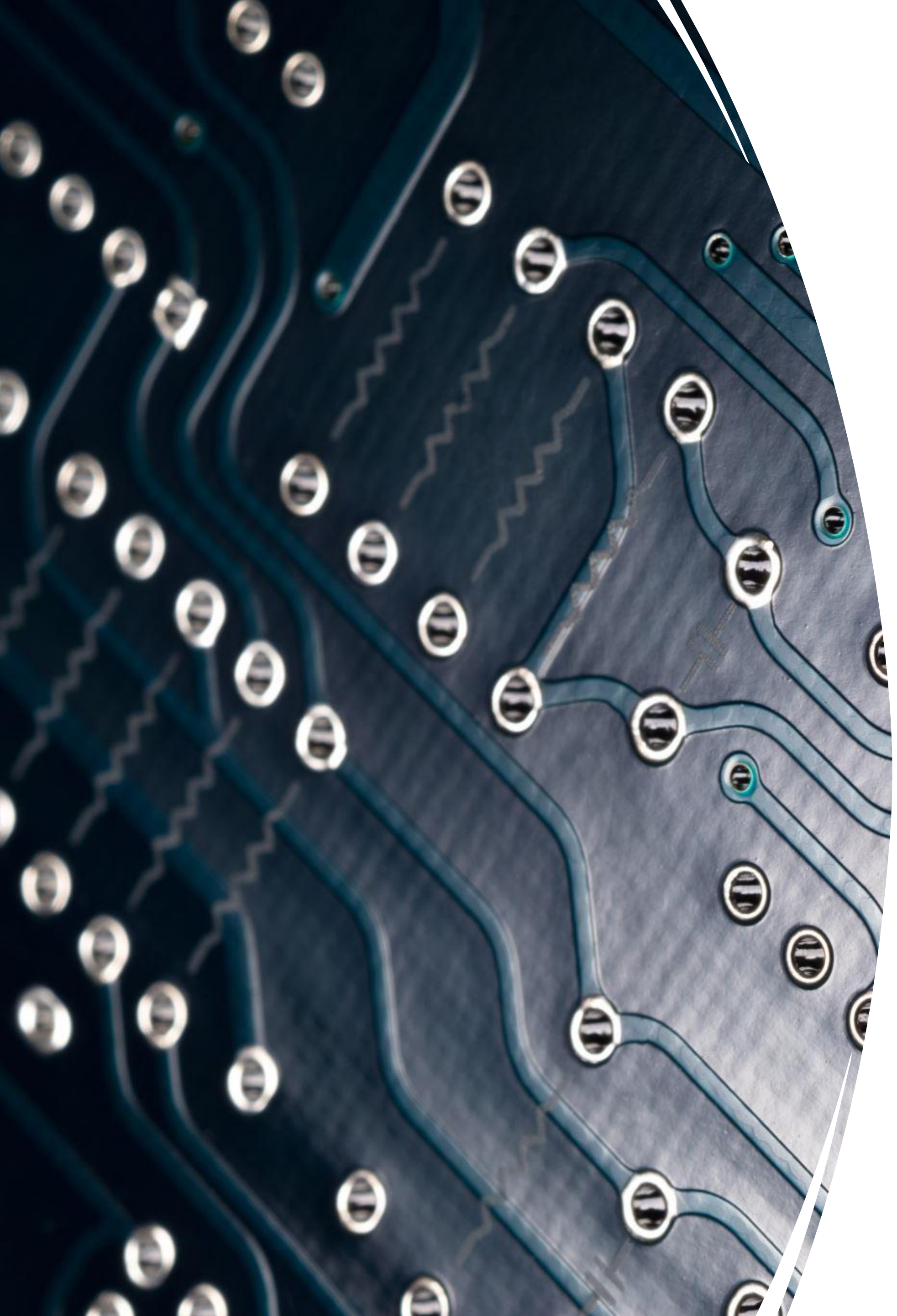
1:	$(A1 \vee A2 \vee A3 \vee A4 \vee A5)$	premise
2:	$(A1 \vee A2 \vee A3 \vee A4 \vee A5) \rightarrow (A2 \wedge \neg G)$	premise
3:	$B124,\ B124 \rightarrow (A2 \wedge G)$	premises
4:	$A2 \wedge \neg G$	$\rightarrow$ elim 2,1
5:	$\neg G$	$\wedge$ elim 4
6:	$A2 \wedge G$	$\rightarrow$ elim 3.2,3.1
7:	G	$\wedge$ elim 6
8:	$\perp$	$\neg$ elim 7,5
9:	$A2$	contra (constructive) 8

Given hand: 2 diamonds (A1), jack spades (A2), 9 hearts (A3), 3 spades (A4), 2 hearts (A5) - 3 hearts is pile card (I)

Here, we can prove that A2 is the best possible choice for what to put down.

There is a possible double combination with pile card and A4: B124 exists

We will put down single card with highest self-value: A2.



First-Order Extension

For first-order extension, the universe of discourse could be a standard 52 card deck. Each variable would be arbitrary card from said deck.

Originally, the propositions and constraints relied on subscripts to distinguish individual elements in the universe of discourse. However, the implementation of predicate logic allows us to use the ∀ and ∃ quantifiers to effectively do the job of the original subscripts.

First-Order Extension

Example 1:

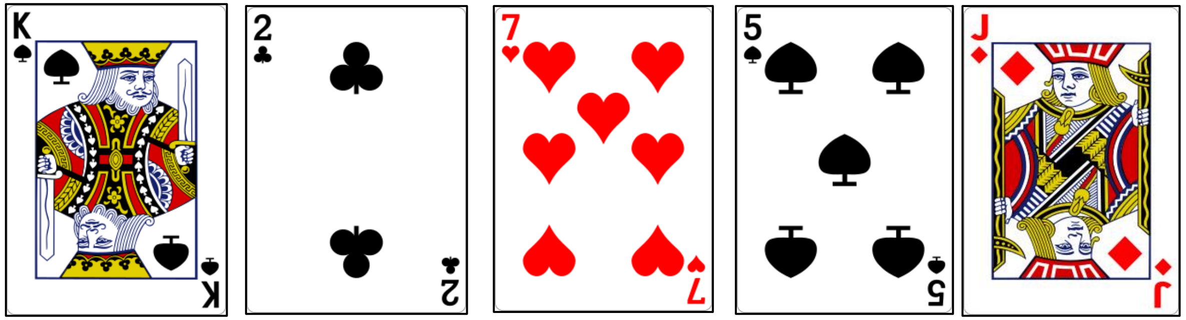
Domain = {52 card deck}

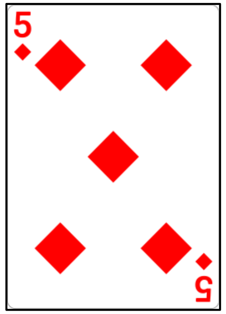
H(a,b,c,d,e) = cards a,b,c,d,e are in a hand together

D(a,b) = a and b are doubles

$\exists a,b,c,d,e.(H(a,b,c,d,e) \wedge \exists a,b.D(a,b))$

There exists 5 cards in a hand, and there exists a Double combo (multiple) within the hand.





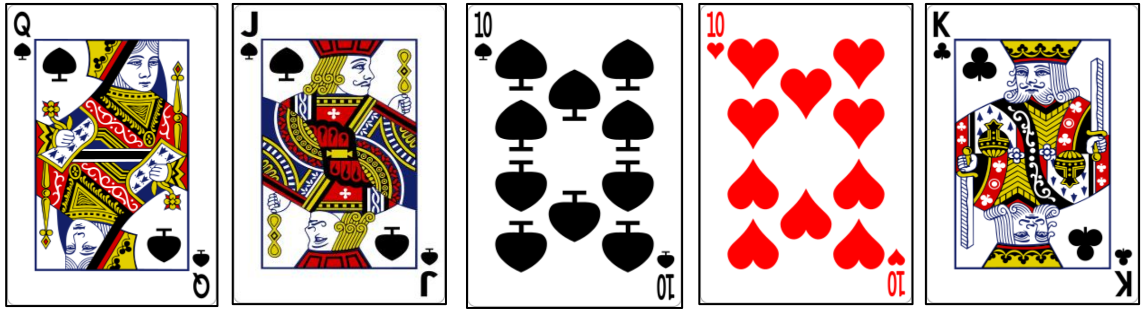
First-Order Extension

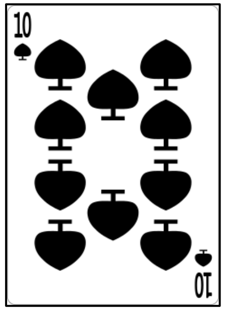
Example 2:

Domain = {52 card deck}
H(a,b,c,d,e) = cards a,b,c,d,e are in a hand together
T(a,b,c) = a,b,c are triples

$\exists a,b,c,d,e.(H(a,b,c,d,e) \wedge \exists a,b,c.T(a,b,c))$

There exists 5 cards in a hand, and there exists a Triple Combo (multiples) within the hand.





Model Exploration

- The Model that was created relies heavily on the principals of object-oriented programming in python.
- There are two .py files and one directory containing photos of playing cards.
- gameobj.py contains: Card object, Deck object, and Game objects.
- Main.py contains a window class that implements a tkinter GUI for a user to interact / play the game Yaniv.

- The window contains 3 deck objects, each deck object is a collection of card objects.

- The Deck object has methods that generate a 52 deck of cards, and another that can gather the groups within a deck into a nested list.

The finalized version of the model can be found at:
https://github.com/ajsib/CISC204_project/tree/main/documents/final/v4

How the model works

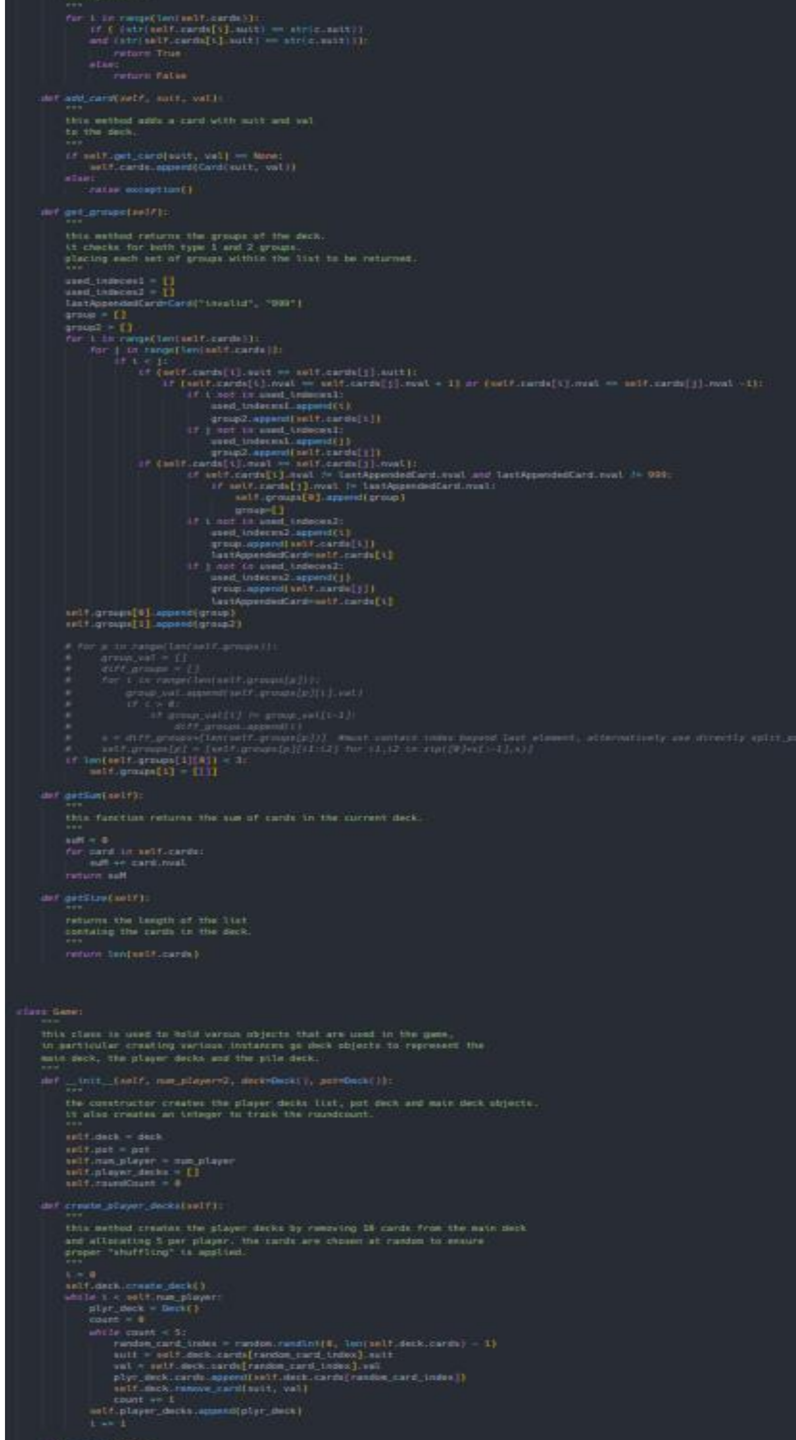
The model that was devised randomly determines the players hands each time the GUI is opened. As such the model is a demonstration of the game.

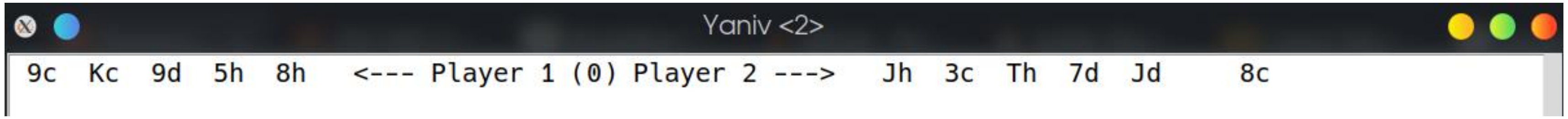
The various methods that are implemented in [game_obj.py]() and [main.py]() represent the logic that was necessary to build a working model of the game in python.

See the github repo under documents/final to see the pydoc documentation for the two python files.

To explore the model we will play a sample game and explore how the use of logic and code was able to produce the GUI that is seen.

We will give more emphasis to the game itself rather than the GUI, since building using tkinter is tedious at best, and group 3 is of the opinion that it is best to skip over that.





- In previous slides, the starting point of the GUI is demonstrated, instead it will be assumed that the user has seen up to that point.
- In the new game the current state is shown in the Get Stats menu:
- Recalling Group3's problem statement: the goal is to find the most efficient move.

The first thing to do is find the groups. In the game the deck.get_groups() method is used for this. In this game, player 1 has two type 1 groups, of the a) and b) variety. That is there are double 9's in the deck and a possible move of double 8's if the player picks up the pot card.

The next thing that is determined is the worth of all groups. In total since the double 9's have the highest value of all groups / cards, this group is chosen for move.

Since the pot card is >= 7, the best probability of a better card is to choose option 2 and take up a new card.
Thus our move is to choose the group -> 9 clubs;9 diamonds; and take up a new card.
From this move a 9 of hearts is picked up.

Thank you

- Ethan, Aidan, Will, Oliver