

Informal Design Document: Generator for Bridge Hands

B. Molinari, DCS

May 21, 1999

1 Specification

The program `bridge_game` will repeatedly deal a random bridge game. A game is presented in the following format.

North

Spades : A, Q, 10, 8, 3

Hearts : J, 8, 7

Diamonds: 5, 3

Clubs : 10, 9, 8

7 points

East

Spades : 9, 4

Hearts : 10, 5, 4, 3

Diamonds: K, J, 10, 7

Clubs : Q, J, 4

7 points

South

Spades : K, 7, 6, 5, 2

Hearts : Q, 9

Diamonds: Q, 9, 4

Clubs : 6, 5, 3

7 points

West

Spades : J

Hearts : A, K, 6, 2

Diamonds: A, 8, 6, 2

Clubs : A, K, 7, 2

20 points

The standard bridge convention is followed, in that each card is given a point value according the following table.

card	points
ace	4
king	3
queen	2
jack	1
others	0

If there is only one card in a suit, 1 further point is added to the point value of the hand. If there are no cards in a suit, 2 further points are added to the point value of the hand.

The program will use a random-number generator and the user will be prompted at the start of the program for a seed value.

2 Top-level Design

A game is generated by the following steps, in sequence.

- A deck of 52 cards is created, with the cards in order.
- This deck is shuffled by a sequence of random card exchanges.
- The cards are dealt from this deck into the four hands.
- Each hand is sorted into the standard bridge format, shown in section 1.
- The game is displayed, including the point value of each hand.

3 Classes and their relations

CARD A card object has two attributes, namely, suit and rank. The latter represents the face value of a card, from 2 to 10 then through the “picture cards” and then the ace.

SUIT_IN_HAND A `suit_in_hand` object has two attributes. One is an array of cards, possibly empty. All these cards belong to the same suit (have the same value for their suit attribute). This suit is recorded as the suit attribute.

HAND A hand-object has one attribute, namely, an array of four suit-objects. The total number of cards involved in these suit-objects is 13.

BRIDGE_GAME A `bridge_game` object has four attributes, namely north, east, south and west, where each is a hand. It also has a deck attribute. Finally it has an attribute that represents a random-number generator.

DECK A deck object has an attribute that is an array of 52 cards.

CARD_CONSTANTS This class defines integer constants to represent suit values and rank values, and gives them appropriate names. These names are used as required in the other classes.

The inheritance relationship between the classes is given by table 1.

The client relationship between the classes is given by table 2.

class	is-inherited-by
CARD_CONSTANTS	CARD SUIT_IN_HAND HAND DECK

Table 1: Inherit relationship

class	is-client-of
BRIDGE_GAME	DECK HAND
HAND	SUIT_IN_HAND CARD
SUIT_IN_HAND	CARD
DECK	CARD

Table 2: Client relationship

4 Functional Design

The creation of a deck object involves a loop through the 4 suits, and for each suit an inner loop through the 13 rank values. Each iteration of the inner loop creates a card-object with the relevant suit and rank attributes.

The deck is shuffled by looping through the 52 cards in the deck object. At each iteration a second card is chosen randomly in the deck, and the two cards are exchanged.

A hand object is created with four empty `suit_in_hand` objects. When a card is dealt to a hand it is added to the relevant `suit_in_hand` object.

A hand object is sorted by, in turn, sorting each of its `suit_in_hand` objects.

The point value for a hand object is computed by summing the point value for each of its `suit_in_hand` objects.

Notes

The design of sections 2, 3 and 4 can be developed from the specification of section 1. The objects and their attributes are described informally. The relationships between the relevant classes can then be stated, *before* any Eiffel is written. Many details remain to be filled in, but the program is starting to take shape. If the later detailed work causes the general design to be modified then this document *must* be changed to be consistent with the new design.