

## 3.6: Digitization Project

Shaurya & Davide

December 8, 2021

### Contents

<b>1</b>	<b>Algorithm:</b>	<b>1</b>
<b>2</b>	<b>Example: 9 of Hearts</b>	<b>2</b>
<b>3</b>	<b>Effeciency of the encoding:</b>	<b>2</b>

### 1 Algorithm:

```
# The first two bits denote the suit.
if the suit clubs then set first two bits = =00=
    if suit is diamonds set first two bits = =11=
    if suit is hearts set first two bits = =10=
    if suit is spades set first two bits =01=.
# The next four bits denote the rank
if rank is a symbol then:
    if the rank is ace, set last 4 bits to =0001=
        if the rank is king set last 4 bits to =1100=
        if the rank is jack set last 4 bits to =1010=
        if the rank is queen set last 4 bits to =1011=
    if the rank is a number then:
        use the binary equivalent (e.g. 10 of spades is the 9th number from 2-
        10, therefore it will end in =1001= (9 in binary))
```

## 2 Example: 9 of Hearts

First two bits = `10` Last four bits = `1000` (8 in binary) Encoded result: `101000`

Theoretically, 52 cards with each needing a unique code in bits. would only require 5.70044 bits ( $2^x = 52$ ), but since each bit needs to be either one or zero, a practical minimum of 6 bits is required

We chose a fixed length encoding, since there are few enough cards that we can fit all possible choices 6 bits. The algorithm encodes the following information:

- Suit of card (first two bits)
- Color of card (first bit, `0` = `black`, `1` = `red`)
- Rank of card (2-10, Ace, Queen, King, Jack)

## 3 Efficiency of the encoding:

The algorithm is efficient, and takes up the least space possible while staying simple and quick to execute. However, the algorithm doesn't scale very well (e.g. if we add the two joker cards as a suit for a total of 54 cards, using this method we would have to use 3 bits to determine the suit of card, resulting in a minimum 7 bits).

An alternate solution would be to assign each card an arbitrary (but unique) 6 bits (similar to ASCII 32), we could theoretically handle 64 cards ( $2^6 = 64$ ) while still using 6 bits of space, at the cost of being harder to decode by humans.