Example Using Bit-Level Ops

- How can we encode std. deck of cards?
  52 cards in 4 suits
- Want to be able to compare cards to find the higher "value"
  Want to identify the suit.
- **Attempt 1:** Use 1 bit/card - set corresponding bit to 1.
  Problems?
- **Attempt 2:** Use 1 bit/suit + 1bit/number
- Can we do better?

Better Representations

- **Attempt 3:** Binary encode all 52 cards.

- **Attempt 4:** Separately encode suit (2bits) and number (4 bits)

Compare Cards: Suits

```
char hand[5];
char card1, card2; / / Cards to compare
card1 = hand[0];
card2 = hand[1];
```

etc.

```
if sameSuit(card1, card2)) { ...}
#define SUITMASK 0x30
int sameSuit(char card1, char card2){
   return (!(card1 & SUITMASK) ^ (card2 & SUITMASK) ); }
```

Compare Cards: Numbers

```
char hand[5];
char card1, card2; / / Cards to compare
card1 = hand[0];
card2 = hand[1];
etc.
if largerNumber(card1, card2)) { ...}
#define NUMMASK = 0x0F {
int largerNumber(char card1, char card2){
   return ( (unsigned int) (card1 & NUMMASK) >
           (unsigned int) (card2 & NUMMASK) )); }
```