

Solitaire Prime - Prog 1
CECS 282-01 – Fall 2021
Due: Monday Sept 13, 2021 @ 8:00 PM

Solitaire is a card game that is played by one person. Solitaire is also known by the name “Patience”. It is estimated that there are more than 1700 different versions of Solitaire... make that 1701 because we are inventing a new Solitaire game today.

Welcome to Solitaire Prime! This game uses one standard deck of cards. Here are the rules:

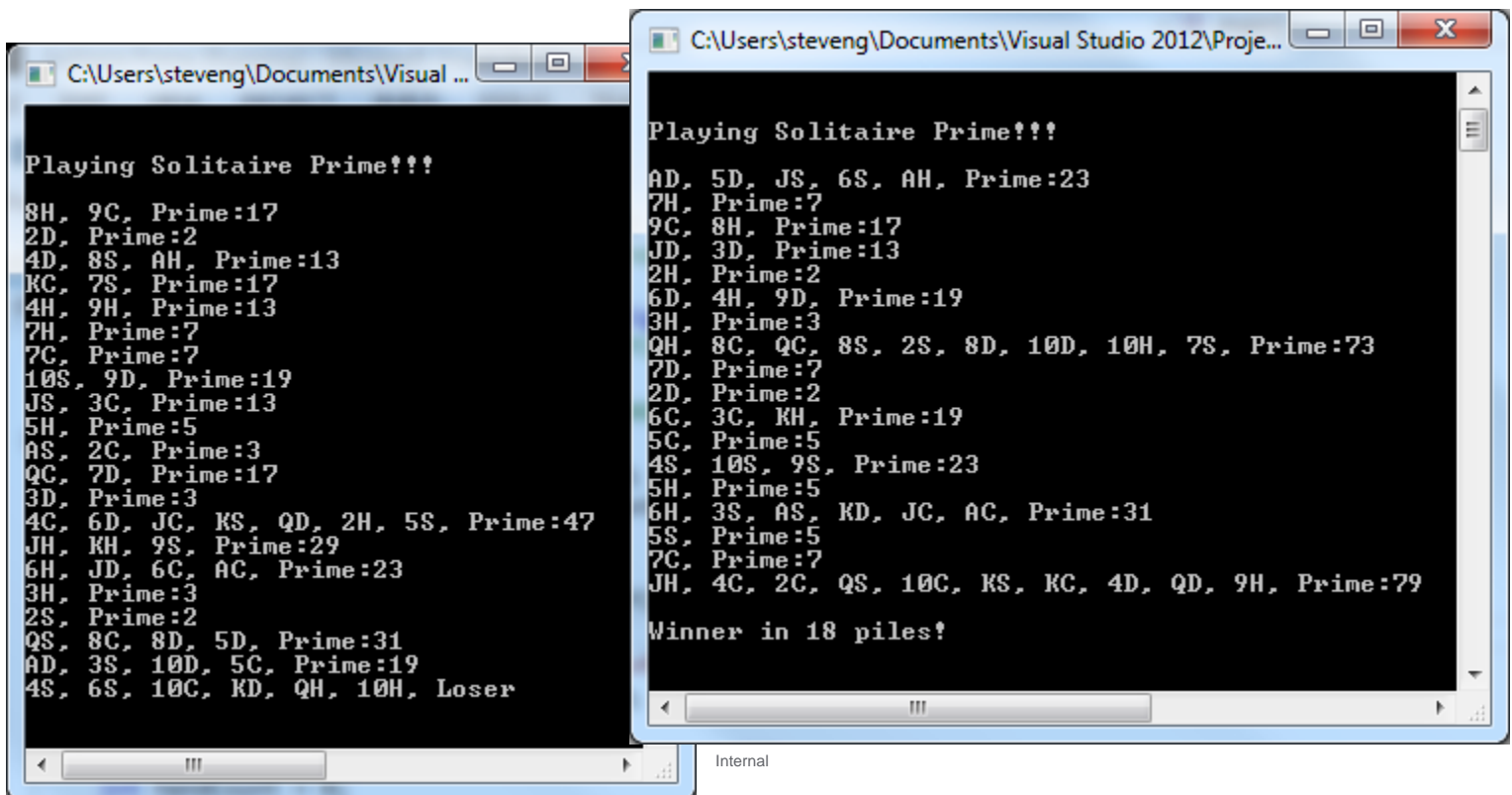
- 1) Take the top card from the deck and place it face up on the table.
- 2) The Sum is now the value of that card (Ace = 1, 2 = 2, ... 10 = 10, Jack = 10, Queen = 10, King = 10)
- 3) If the Sum is a prime number, discard that pile (hand), and start over at instruction #1
- 4) If the Sum is not prime, take the next card from the top of the deck and place it on top of the card stack (pile or hand) on the table.
- 5) The Sum is now the sum of all cards in the stack on the table.
- 6) Go to instruction #3.

Continue to play the game, keeping track of how many piles you have created that are prime.

If the last card from the deck gives you a prime pile, then you win! Write the word “Winner” on the screen and show how many prime piles there were.

If the last card from the deck does not give you a prime pile, then you lose. Write the word “Loser” on the screen.

These 2 screen shots show possible winner and loser hands. When you acquire a Prime pile, print out the value (prime number) and start over on the next line.



```
C:\Users\steveng\Documents\Visual Studio 2012\Proje...
Playing Solitaire Prime!!!
8H, 9C, Prime:17
2D, Prime:2
4D, 8S, AH, Prime:13
KC, 7S, Prime:17
4H, 9H, Prime:13
7H, Prime:7
7C, Prime:7
10S, 9D, Prime:19
JS, 3C, Prime:13
5H, Prime:5
AS, 2C, Prime:3
QC, 7D, Prime:17
3D, Prime:3
4C, 6D, JC, KS, QD, 2H, 5S, Prime:47
JH, KH, 9S, Prime:29
6H, JD, 6C, AC, Prime:23
3H, Prime:3
2S, Prime:2
QS, 8C, 8D, 5D, Prime:31
AD, 3S, 10D, 5C, Prime:19
4S, 6S, 10C, KD, QH, 10H, Loser

C:\Users\steveng\Documents\Visual Studio 2012\Proje...
Playing Solitaire Prime!!!
AD, 5D, JS, 6S, AH, Prime:23
7H, Prime:7
9C, 8H, Prime:17
JD, 3D, Prime:13
2H, Prime:2
6D, 4H, 9D, Prime:19
3H, Prime:3
QH, 8C, QC, 8S, 2S, 8D, 10D, 10H, 7S, Prime:73
7D, Prime:7
2D, Prime:2
6C, 3C, KH, Prime:19
5C, Prime:5
4S, 10S, 9S, Prime:23
5H, Prime:5
6H, 3S, AS, KD, JC, AC, Prime:31
5S, Prime:5
7C, Prime:7
JH, 4C, 2C, QS, 10C, KS, KC, 4D, QD, 9H, Prime:79
Winner in 18 piles!
```

You will have 2 classes:

- 1) The Deck class which will create the deck of cards
- 2) The Card class which creates cards

The main logic of the program will be in the main function. You will use the Card class and the Deck class to play the game.

Here the methods you will need to create. Feel free to add more if you need them.

```
public class Deck
    public Deck() // constructor which creates a deck of 52 cards. Ace of Spades on top, followed by the
rest of the spades in order, followed by Hearts, Diamonds and Clubs.
    public void refreshDeck(); // reset the deck so it looks like a new deck.
    public Card deal() // deal a card from the top of the deck.
    public void shuffle() // shuffle the cards in the deck.
    public int cardsLeft() // return the number of cards left in the deck
    public void showDeck(); // show all the cards in the deck: 13 columns and 4 rows.

public class Card
    public Card() // create a "blank" card
    public Card ( char r, char s ) // constructor to create a card, setting the rank and suit
    public void setCard( char r, char s ) // set an existing blank card to a particular value
    public int getValue() // return the point value of the card. Ace = 1, 2 thru 10, Jack = 10, Queen = 10,
King = 10
    public void showCard() // display the card using 2 fields... Ace of Spade:AS, Ten of Diamond:10D,
Queen of Heart:QH, Three of Club:3C. (If you want to get fancy, you can use these symbols for the suit ♠, ♣, ♥,
♦)
```

In the main function, you will have a menu that looks like this:

Welcome to Solitaire Prime!

- 1) New Deck
- 2) Display Deck
- 3) Shuffle Deck
- 4) Play Solitaire Prime
- 5) Exit

New Deck will create an unshuffled deck in the following order: Spades, Hearts, Diamonds, Clubs... Ace, 2, 3, ..., 10, Jack, Queen, King

Display Deck will display all cards in a grid: 13 columns by 4 rows.

Shuffle Deck will randomly shuffle all cards in the deck.

Play Solitaire will play the game as described above.

Exit will exit the program.

You must create your own function to shuffle (cannot use the `random_shuffle` provided by C++) and you must create your own function called “`isPrime`” (cannot use any C++ library function). **One is NOT a prime number**. You must use array to store your cards in Deck. You cannot use vector for this assignment.

Objectives:

- 1) Understand how to create classes
- 2) Learn how to use arrays in C++
- 3) Learn how to use header files and .cpp file as separate files.
- 4) Learn about preprocessor directives, #include libraries, and “using namespace std”
- 5) Learn how to protect header files by using #ifndef, #define, #endif
- 6) Difference between array of objects in C++ versus Java.
 - a. Java arrays of objects have object references -= not actual objects. No need for a default constructor – example -> array of cards: `Card [] deck = new Card[52];` This array contains 52 Card references but no cards.
 - b. C++ arrays of objects have the actual objects. Need to have a default constructor. Example-> `Card deck[52];` This array contains 52 cards – you need to create a default constructor, then you need to reset all 52 cards to actual values with a function `setCard('A', 'S')` for the Ace of Spades.
- 7) Introduce the `rand()` function for shuffling cards
- 8) Get a random number within a range of numbers