CIS7 Project Documentation Guide

In the documentation, provide at least 2 pages (single-space) that contains the following components of your course project:

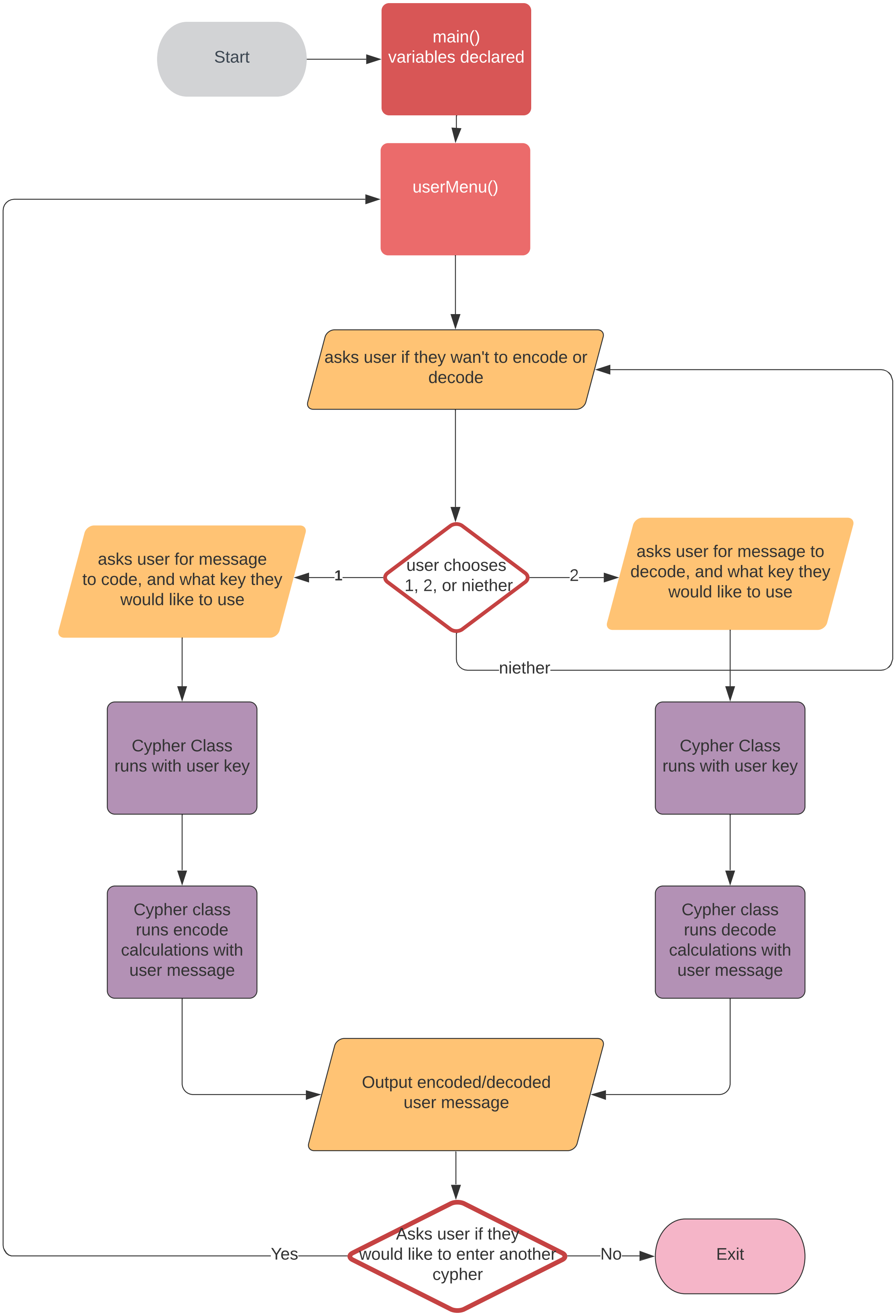
1. Team name, members.
   Andrew Whitfield

2. Project Information and details: (30 points)
   - What problems are you solving in this project?
     - Problem 1: code or decode Cypher,
     - Problem 2: Boredom
   - What solutions are you implementing in the project?
     - A program the encodes or decodes a Vigenere Cypher,
     - A program that is a fun game.
   - Provide explanation of calculations and algorithm implementation.
     - Program 1: Ei = (Pi + Ki) mod 26 is used to get the Vigenere "Shift" of letters, you do have to add 65 to the value though since the formula just gives you the relative point between the two givens. Relative to the ascii scale you need to add 65 to the answer of Ei to get the right letter selction.
     - Program 2: Nothing complicated, array selection is used, but everything is simple addition or subtraction
   - What is the program objectives? Explain how your program is interacting with the user and its purpose.
     - Program 1 – objective: "a program that encrypts and decrypts a message from the user."
     - Program 2 – objective: Make a Black Jack game
   - How is discrete structures implemented in the C++ program?
     - Program 1 : used mod
     - Program 2 : and, or, bool comparisons
   - What are the limitations of the program?
     - Program 1 : Can only use upper and lower case letter.
     - Program 2: No betting, No soft hands, No split hands, Only one player
   - Provide recommendation on improving the limitations of the program.
     - Program 1: add exceptions for spaces
     - Program 2: add features that are limited by the program, as stated above.
3. Flowchart AND Pseudocode. (30 points)
   - Write the pseudocode for the program, from start to finish. Be sure to include decision-making branching.
   - Use standard shapes for flowchart, be sure to include decision-making branching.

Cypher pseudo code

```
Declare userKey, userMessage, encode, decode, inputCheck, userSelection
Call userMenu()
Declare inputCheck(2), userSelection(2)
Choose 1 or 2 (1 for encode, 2 for decode,  or neither then asks for re-input)
User inputs message and key
Class Cypher called with userKey
String k declared and built with userKey values
If user chose 1 -> goes through for loop using formula ($E_i = (P_i + K_i) \bmod 26$) ->
        assign to output variable -> output to user
If user chose 2 ->goes through for loop using decode ($D_i = (P_i + K_i+26) \bmod 26$) ->
        assign to output variable -> output to user
Return to main()
User asked for redo, or exit
y returns to before userMenu() call
n exits the program
```

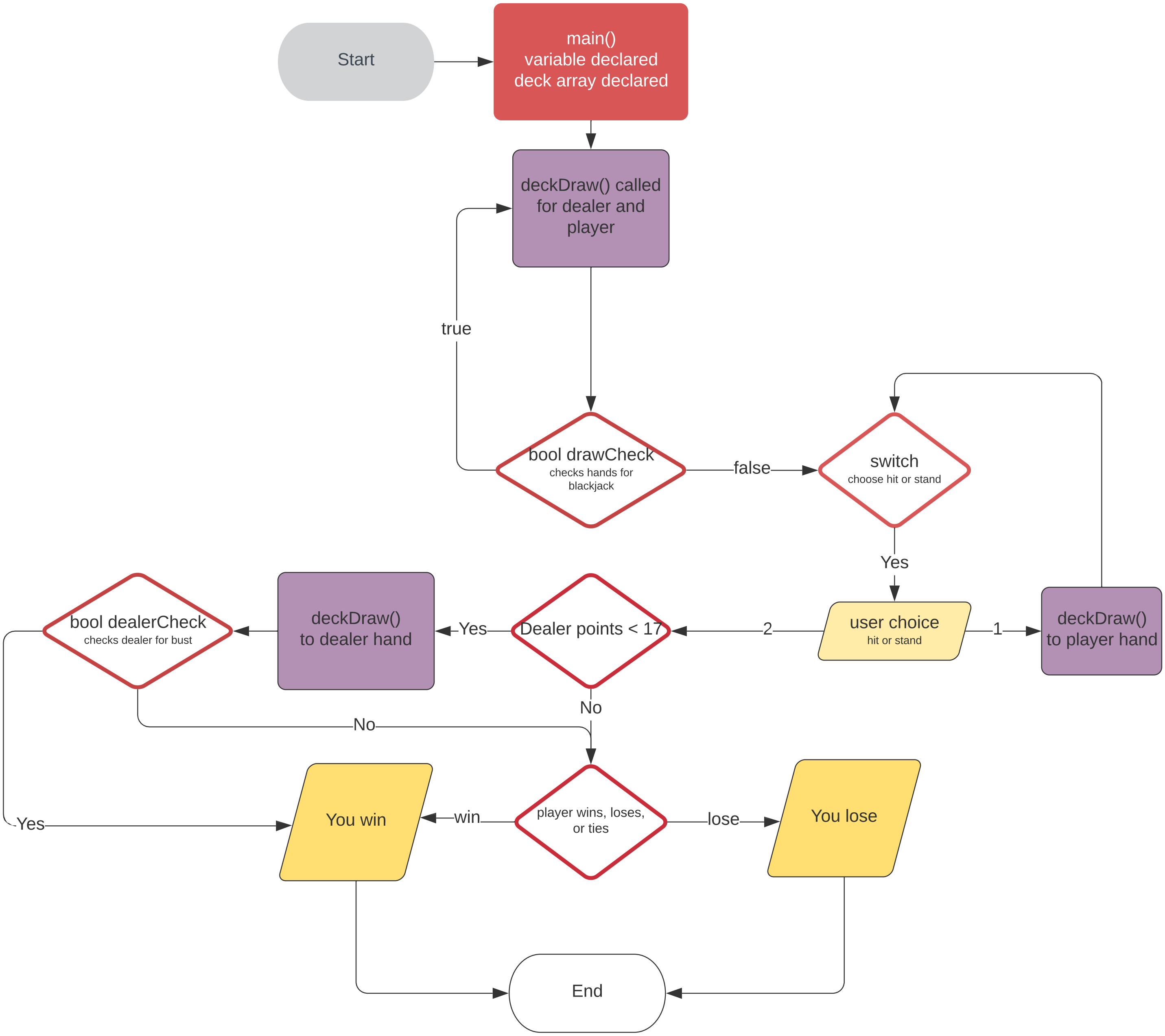
Cypher Program Flowchart


Start
main()
variables declared
userMenu()
asks user if they wan't to encode or decode
user chooses 1, 2, or niether
1
2
niether
asks user for message to code, and what key they would like to use
asks user for message to decode, and what key they would like to use
Cypher Class runs with user key
Cypher Class runs with user key
Cypher class runs encode calculations with user message
Cypher class runs decode calculations with user message
Output encoded/decoded user message
Asks user if they would like to enter another cypher
Yes
No
Exit

Black Jack psuedo code

```
Declare variables drawCheck, userCheck, dealerCheck, cardAvailableCheck, playerChoice, playerTurn, elementHolder, cardholder, dealer, player, player1, dealer1
Declare arrays deck, point
While drawCheck true
 deal hands with drawCard() till no one has blackjack on starting hand
If either has 21
 compares dealer to player for win lose
if the other also has 21, picks tie
While playerChoice true
 will continue to give player option to hit with drawCard(),
till stand is selected.
Checks for bust on player.
While dealerCheck true
 will stand if dealer is 17 or over, will continue to hit till above 16 points.
Checks for dealer bust
Checks for highest hand or tie.
end
```

Black Jack Program Flowchart



Start
main()
variable declared
deck array declared
deckDraw() called for dealer and player
true
bool drawCheck
checks hands for blackjack
false
switch
choose hit or stand
Yes
user choice
hit or stand
1
deckDraw() to player hand
2
Dealer points < 17
Yes
deckDraw() to dealer hand
bool dealerCheck
checks dealer for bust
No
No
Yes
player wins, loses, or ties
win
lose
You win
You lose
End