|  |  |  |
| --- | --- | --- |
|  | TOOP Final Exam  Object Oriented Programming |  |

This section of the final exam is closed book/closed notes and an individual effort. It is to be completed in person on this sheet. You should budget 15 minutes for this section of the exam; however, you are free to take as much or as little time as you wish. Fill in the following blanks with the missing code.

|  |
| --- |
| #include <iostream> // For console IO  #include <string> // for to\_string  using namespace std; // using the standard namespace (std::)  class CPlayingCard {  public:  CPlayingCard();  CPlayingCard(const string& suit, const int& value);  public:  friend ostream& operator<<(ostream&, const CPlayingCard&);  private:  string suit;  int value;  };  // Write a default constructor and a parametrized constructor consistent with the headers provided  // in the class definition. You need no further code; however, there are questions on the next page.  ostream& operator<<(ostream& stream, const CPlayingCard& card) {  string value = "";  if (card.value <= 9) {  value = to\_string(card.value + 1);  } else {  string faces[] = {"Jack", "Queen", "King", "Ace"};  value = faces[card.value - 10];  }  return stream << value << " of " << card.suit;  }  int main() {  CPlayingCard deck[52];  string suits[] = { "Spades", "Diamonds", "Hearts", "Clubs" };  for (int i = 0; i < 52; i++) {  deck[i] = CPlayingCard(suits[(int) floor(i / 13)], i % 13);  cout << deck[i] << endl;  }  return 0;  } |

1. Why is it important that the operator overload for <<, in other words, the to string, be a friend function? Hint: what type of variables is it trying to access?
2. Assume that you were to write a class “CDeck” that would contain many different CPlayingCards. What Object Oriented principles does this describe? Describe why this could be powerful. Hint: Think of methods that involve sorting a deck of cards or drawing a unique card.