**Introduction to Computer Science (MAC 101)**

**Professor: Andi Toce**

**Homework 5**

**Homework:** Write a similar program to *DrawCards.cpp* for a bag that contains the eight objects: Each item has a unique combination of color (red, blue, orange, green) and shape (ball, cube). Every time an object is picked from the bag, it can’t be picked again, so the number of possible choices decreases by one. The logic should be identical to that in *DrawCards.cpp*, but the array settings will differ. You may also want to give your variables different names, such as items\_remaining and (for the integer array) items\_picked.

The program should also deal with the case when we run out of items. Simulate replacing all items in the bag and continue picking items. (Hint: Read the book)

|  |  |
| --- | --- |
| DrawCards.cpp | Output |
| #include <iostream>  #include <cstdlib>  #include <ctime>  #include <cmath>  using namespace std;  int rand\_0toN1(int n);  void draw\_a\_card();  int select\_next\_available(int n);  bool card\_drawn[52];  int cards\_remaining = 52;  char \*suits[4] = {"hearts", "diamonds", "spades", "clubs"};  char \*ranks[13] = {"ace", "two", "three", "four", "five", "six", "seven", "eight", "nine", "ten", "jack", "queen", "king"};  int main() {  int n, i;  srand(time(NULL)); // Set seed for randomizing.  while (1) {  cout << "Enter no. of cards to draw (0 to exit): ";  cin >> n;  if (n == 0)  break;  for (i = 1; i <= n; i++)  draw\_a\_card();  }  return 0;  }  // Draw-a-card function  // Perform a card draw by getting a random 0-4 and a random 0-12.  // Use these to index the strings arrays, ranks and suits.  void draw\_a\_card() {  int r; // Random index (0 thru 12) into  // ranks array  int s; // Random index (0 thru 3) into  // suits array  int n, card;  n = rand\_0toN1(cards\_remaining--);  card = select\_next\_available(n);  r = card % 13; // r = random 0 to 12  s = card / 13; // s = random 0 to 3  cout << ranks[r] << " of " << suits[s] << endl;  }  // Select-next-available-card function.  // Find the Nth element of card\_drawn, skipping over  // those elements already set to true.  int select\_next\_available(int n) {  int i = 0;  // At beginning of deck, skip cards already drawn.  while (card\_drawn[i])  i++;  while (n-- > 0) { // Do the following n times:  i++; // Advance to next card  while (card\_drawn[i]) // Skip past cards  i++;  }  card\_drawn[i] = true;  return i;  }  int rand\_0toN1(int n) {  return rand() % n;  } | Enter no. of cards to draw (0 to exit): 10  seven of hearts  six of diamonds  ace of diamonds  six of hearts  four of diamonds  five of clubs  ten of diamonds  queen of hearts  three of hearts  seven of diamonds  Enter no. of cards to draw (0 to exit): |

**How to submit**

Save the file as LastName.FirstName.HW5.cpp

**E-mail** me the file by the due date. Late submissions will **NOT** be accepted

**Grading**

Correctness – 50%

Efficiency – 30%

Code style and appearance – 20%