

Introduction

If you have ever collected hockey cards (or other sports cards) you may have wondered how many packages you have to buy to get a complete set. In this assignment, we will try and find out.

Let's start with some assumptions to pin down the problem. First, we will assume that a complete set consists of 500 different cards. You buy cards in packages of 7.

We will assume that the frequency of the cards follows a uniform distribution. That is, assume that the manufacturer produces the same quantity of each card so that the probability of getting any specific card is the same as any other. (This assumption may not be realistic, but we make it to simplify the problem.)

Assignment

Your task is to write a program that simulates buying packages of cards until the set is complete. At each step of the program, you should simulate the act of "buying a package of cards". For each card you buy, you should keep track of how many copies of it you have in your collection. An experiment consists of repeating this process until you have a complete set of cards.

For each experiment, you should output the maximum number of duplicates you have of any card. There may be several cards with the same maximum number but you only have to output one of these.

You are to run at least 100 experiments to determine how many packages you expect to have to buy in order to complete the set. When you are finished the experiments, you should output the following information:

1. The average number of packages you had to buy.

2. The maximum number of packages you had to buy to complete your set.
3. The minimum number of packages you had to buy to complete your set.

Methodology

You can use an array to keep track of the individual cards you own and another array to keep track of the cards in the package you just purchased.

To simulate buying a package of cards, you should use a random number generator to generate seven values between 0 and 499. Then you should update the list of cards in your collection by adding these new cards.

To determine whether you have a complete set, you can either keep track of the total number of different cards you have or keep track of the number of cards missing from your set. Each time you update the list you can update this count.

Make sure to seed your random number generator differently for each experiment to ensure that each experiment generates a different collection of cards.

You must define the following functions:

a) A function which generates a package of seven cards. This function has a single array parameter. It should populate an array of seven cards by generating seven distinct values between 0 and 499. If a duplicate value is found, it should be discarded so that the seven

values are all distinct. The random number generator should be seeded in the main program or the numbers generated will always be the same. (Hint: When developing your program you can define this function to allow duplicate cards. Eliminating the duplicates is more difficult and should be done after you get the rest of the program working. You will get part marks if you do not finish that part of the program.)

b) A function to scan the array of cards in your collection to find the most duplicates you bought.

Requirements

Your code must meet these requirements:

- * The program must be written in **C**
- * Your program can write to the standard output or to a file.
- * Use **sensible** variables names.
- * **Comment** and **indent** your code