Background

The purpose of this assignment is to gain experience using a new data structure, the array. Arrays allow you to store multiple values of the same type (`int`, `float`, `double`, `char`, etc.) using a single variable name. Arrays are often used to store multiple values in which the data are in some way related. Because the individual values stored within an array can be accessed using an "index" (or "subscript"), arrays are well suited for repetitive tasks and used in conjunction with loops.

Problem Statement

In this assignment you are helping to analyze the win percentage of teams who won the NBA championship from 2001 to 2016. You are to write a program that will prompt the user to enter data on the command line, store the data into arrays, and perform calculations on the data.

The data that will be entered on the command line during execution of the program is given in Table 1, below. Two arrays will be used to store NBA Championship data for years 2001 to 2016. One array will contain the number of wins and a second array will contain the number of losses. The years do not need to be stored into an array; they can be determined by manipulating the array indices.

Table 1: NBA champion regular season wins and losses for years 2001 to 2016

Year	Number of Wins	Number of Losses
2001	56	26
2002	58	24
2003	60	22
2004	54	28
2005	59	23
2006	52	30
2007	58	24
2008	66	16
2009	65	17
2010	57	25
2011	57	25
2012	46	20
2013	66	16
2014	62	20
2015	67	15
2016	57	25

Instructions

Represent

- Individually, use pseudocode to create a plan for solving the problem.

Plan

- Create a file named **APP_C26_1.cpp**
- Outline the steps your program will take by adding comment statements to your file based on the pseudocode.

Implement

- Write a complete C program **APP_C26_1.cpp** to perform the following tasks:
  - Use a `for` loop to prompt the user to input the data for years 2001 to 2016 (given in the table above).

  - As the data is entered on the command line, store it in two arrays, one for the number of wins and one for the number of losses
  - Compute the winning percentage for each year and store it in an array.
  - Find the minimum and maximum winning percentages and the corresponding years and print both to the screen.
- Compile, link, and run your program.
- Once working modify the program to perform the following tasks:
  - Create and open an output file named **APP_C26_1_result.txt** .
  - Write the same values to this output file that are also printed to the screen (only the maximum and minimum percentages and corresponding years).
  - Close **APP_C26_1_result.txt** .
- Compile, Link, and Run your program.
  - Tip: When debugging, decrease the number of years that you are entering to make it quicker to run/debug.

Evaluate

- Perform a hand calculation to verify and check your results.

Document

- Create a single PDF that includes your planning documentation, your code, the terminal output, the output file, and your verification.
- Submit the PDF to Carmen according to the DAL.

Include the standard comment, **`printf(),`** and **`fprintf()`** statements indicating name, seat number, etc.