CLA CS-101 (A1)
Introduction to Computers
Spring 1995

Homework # 5
Due on Tuesday 3/14/95

1 The program listed at the end of this assignment is proposed to sort a list of student grades stored in a vector (array) called `grade`. Acquaint yourself with the program and then answer the following questions:

a. Assume that the user entered the data shown in the "sample execution" (listed below the program). How many times will the `if (grade[i] < grade[j]) then` instruction be executed? How many times will it be executed if the user specified that the number of students in the class is 50?

b. How would you modify the program below so as to sort the list of grade in an ascending order (i.e. smaller grades come first followed by larger ones)?

2 Answer the following questions:

a. Consider the condition below, where `age` and `weight` are integer variables, `sex` is a char variable, and `smoker` is a boolean variable.

```
(( age>65 and sex='M') or (age>70 and sex='F') or (smoker and weight>300))
```

Give 3 sets of values for `age`, `weight`, `sex`, and `smoker` for which the condition above will evaluate to **True**. And, give 3 sets of values for `age`, `weight`, `sex`, and `smoker` for which the condition above will evaluate to **False**.

b. What are the main differences between the **for**, **while**, and **repeat** loops?

3 In class we discussed a Pascal program that can be used to cast votes in an election and find out the winner. Write such a Program. Assume there are 10 candidates and that the candidate who gets the maximum number of votes wins. Submit a listing of your program as well as a sample execution (showing that it works!)

Note: This problem is optional. If you solve it you could get up to 25 points extra-credit to be added to the lowest grade of any one homework in the first half of this semester (from homework #1 to homework #5).

Good Luck!

Program to sort a list of grades

```
program sort(input,output);  (* Program to sort an array of integers *)

var
  n : integer ;                               (* Total number of students    *)
  i, j : integer ;                            (* Loop indeces                *)
  grade : array [1..100] of integer ;         (* Vector to store grades      *)
  temp : integer ;                            (* Temporary variable          *)

begin
  write('Number of students (100 max): ') ;  (* Prompt user to enter n      *)
  readln(n) ;                                 (* Store user input in n       *)
  writeln() ;

  for i := 1 to n do                          (* Loop for all students       *)
    begin
      write('Grade #', i, ': ') ;             (* Prompt user for grade[i]    *)
      readln(grade[i]) ;                      (* Store user input in grade[i] *)
    end ;

  for i := 1 to n-1 do                        (* Loop to sort names          *)
    for j:= i+1 to n do
      if (grade[i] < grade[j]) then
        begin
          temp := grade[i] ;
          grade[i] := grade[j] ;
          grade[j] := temp ;
        end ;

  writeln() ;                                 (* Print out sorted grades     *)
  writeln('Sorted Grades') ;
  for i := 1 to n do
    writeln(grade[i]) ;
end.
```

Sample execution of program

```
[sphinx ~] pc sort.p
[sphinx ~] a.out
Number of students (100 max): 4

Grade #         1: 23
Grade #         2: 74
Grade #         3: 69
Grade #         4: 92

Sorted Grades
        92
        74
        69
        23
```