9 A one-dimensional (1D) array `Teams[]` contains the names of 10 football teams in a local league. A two-dimensional (2D) array `Results[]` stores, for each team, the total number of:

- games won
- games drawn
- games lost
- points.

The position of any team's data is the same in both arrays. For example the data in index 3 of `Results[]` belongs to the team in index 3 of `Teams[]`

The array data will be used to find the current leader of the league.

The variable `Played` stores the number of games played by each team. Each team plays the same number of games.

Write a program that meets the following requirements:

- allows the number of games played to be input and stored, with a maximum of 18 games
- allows the names of the teams to be input and stored
- allows the number of games won, drawn and lost to be input and stored for each team
- validates the number of games played and the number of games won, drawn or lost against the number of games played
- calculates and stores the number of points for each team using three points for a win and one point for a draw; there are no points for a loss
- sorts the array `Results[]` into descending order of number of points, ensuring the corresponding parallel array `Teams[]` is kept in the same order
- determines how many teams have the highest number of points
- outputs the name(s) of the winning team(s) along with the number of points achieved.

You must use pseudocode or program code **and** add comments to explain how your code works.

You do **not** need to declare any arrays, variables or constants; you may assume that this has already been done.

All inputs and outputs must contain suitable messages.

[15]