CSC 362 Programming Assignment #2
Due Date: Tuesday, September 29

In this assignment, you are to use multiple functions and multiple files. If a function is to compute and return a *single value*, it can do so using a return statement where you would call it using an assignment statement like `x = compute(a, b, c);`. Otherwise, if the function is required to compute or input multiple values, you *must pass* these values through the parameter list. You will have to decide which parameters to pass normally versus as addresses. The primary purpose of this assignment is to test your understanding of parameter passing, so doing so incorrectly will result in a loss of points on your grade. Note that you should only pass parameters as addresses for those parameters that will (or may) *change* in the function, all other parameters should be passed normally. Remember strings are already passed as addresses.

The program's role is to predict football games. It will do so by inputting various ratings about two football teams (all input will come from a data file). Each line of the file contains all information for a single game (two teams' worth). There are multiple rows, one per game. The program will input each line of data using fscanf, process that data and output the result. This repeats until the program reaches the end of the file.

Each row consists of 11 items, each separated by a space. The 11 items are in order: HT HTO HTD HTS HTH HTC VT VTO VTD VTS VTR. HT and VT are the home and visiting team's names respectively (strings), all other items are int values on a scale of 1 to 10 (1 being the worst, 10 being the best rating for that particular team attribute). HTO/VTO stand for home and visiting team's offensive strength, HTD/VTD for home and visiting team's defensive strength, HTS/VTS for home and visiting teams' special teams strength, HTH for team's strength at home, HTC for home field advantage (crowd noise) and VTR is the visiting team's strength on the road.

The 11 inputs are used in main, so input them in a single input function but declare them in main to be passed to other functions. There are 5 values to be computed using the 9 int inputs: preference1, preference2, …, preference5. These 5 values are all doubles (declared in main, computed in a compute function). Given the result of these 5 values, compute a prediction which will be made about which team will win and by how much. This will be computed in a separate prediction function. The result of the prediction will be output in a separate output function. Use a summary function to maintain running totals on the number of games predicted and the number of home teams predicted to win. Finally, outside of the loop which iterates for each game, output the results of the summary.

Your program will consist of the following functions:

- main: declare variables, open the input file, iterate through the file (until EOF), and call the various functions listed below. The following functions are called from within the loop except for summary, which is called after. Outside of the loop, also close the file.
    - **input** the 11 data items from the file
    - **compute** the 5 preferences using the following formulas
      preference1 = HTO * *OFFENSE_FACTOR* - VTD
      preference2 = HTD + 2 – VTO * *OFFENSIVE_FACTOR*
      preference3 = HTS * *SPECIAL_TEAMS_FACTOR* – VTS
      preference4 = HTH + HTC ***** *HOME_FIELD_ADVANTAGE* - VTS
      preference5 = HTO * HTD * HTH * *OVERALL_FACTOR* – VTO * VTD * VTR
      Italicized items above are constants (declared using #define in your header). Compute is a single function which receives the needed parameters (e.g., HTO, VTD, etc) and computes the five preference values. These 5 values then need to be accessible from main. You **must use proper parameter passing**. You may (if you wish) implement each of the 5 preference calculations in five additional functions to be called from the compute function. For instance, you could do preference1=compute1(HTO,VTD); See below for a description of the constants in the above formulas.

- **prediction**: given the five preference values, compute a prediction of who wins, home or visitor and by how much. This function works as follows:

  sum = preference1 * *OFFENSIVE_WORTH* + preference2 * *DEFENSIVE_WORTH* + preference3 * *SPECIAL_WORTH* + preference4 * *HOME_WORTH* + preference5 * *OVERALL_WORTH*

  See below for a description of these 5 additional constants. If sum < 0 then the visiting team wins else the home team wins. The amount that the team is predicted to win by is the absolute value of sum, truncated as an int. If the truncated sum is 0, then the home team is predicted to win by 1. You can implement your own absolute value function or use abs from the math.h library. To truncate, cast the value (a double) as an int. For instance, if sum is -6.5, then the road team is expected to win by 6. If sum is 0.8 (truncated to 0), then the home team is predicted to win by 1.
- **output** the winning team name and the losing team name and the amount, such as `Predicted Bengals over Steelers by 16`.
- **update**: modify 3 running totals: number of games predicted so far, number of home team wins predicted so far, number of visiting team wins predicted so far.

- **summary** outputs the number of games predicted and the percentage of those games in which the home team is predicted to win. NOTE: This must be output as a percentage such as 71.33%.

Divide your functions into multiple files as follows:

1. A file containing your main function and an include statement for your header file
2. A header file that contains all function prototypes, all used include statements for C libraries, and all constants used in your program. The constants are as follows:
   - OFFENSIVE_FACTOR = 1.5
   - SPECIAL_TEAMS_FACTOR = 1.2
   - HOME_FIELD_ADVANTAGE = 1.3
   - OVERALL_FACTOR = 1.15
   - OFFENSIVE_WORTH = 0.32
   - DEFENSIVE_WORTH = 0.28
   - SPECIAL_WORTH = 0.15
   - HOME_WORTH = 0.1
   - OVERALL_WORTH = 0.15

   Prototypes should be commented as your .h file is visible but your .c files may not be
3. A file containing your compute, prediction and update functions (all of the computation functions)
4. A file containing your input, output and summary functions (all of the I/O functions)

Run your program on the two data files on the website. Example output for first file is given below. Hand in your source code and the output of running your program on the second of the two data files.

```
Predicted New_England over Pittsburgh by 20
Predicted St_Louis over Seattle by 1
Predicted Cincinnati over Oakland by 8
Predicted Dallas over NY_Giants by 16
Predicted Green_Bay over Chicago by 23
Predicted Kansas_City over Houston by 4
Predicted Indianapolis over Buffalo by 7
Predicted NY_Jets over Cleveland by 11
Predicted Denver over Baltimore by 2
Predicted Philadelphia over Atlanta by 1
Predicted Minnesota over San_Francisco by 15
Out of 11 games played, 45.45% favor the home team
```