For each of the exercises below, include copy and paste your code as well as a screenshot of the output of your program.

1. Write a function that computes the average and standard deviation of four scores. The standard deviation is defined to be the square root of the average of the four values: , where is average of the four scores and . The function will have six parameters and will call two other functions. Embed the function in a driver program that allows you to test the function again and again until you tell the program you are finished.

**#include <iostream>**

**using namespace std;**

**void calc\_stats(double s1, double s2, double s3, double s4, double &mean,**

**double &sigma)**

**{**

**mean = (s1 + s2 + s3 + s4) / 4.0;**

**sigma = pow(s1 - mean, 2) + pow(s2 - mean, 2) + pow(s3 - mean, 2) +**

**pow(s4 - mean, 2);**

**sigma /= 4.0;**

**sigma = sqrt(sigma);**

**}**

**int main()**

**{**

**double s1, s2, s3, s4, mean, sigma;**

**while (true) {**

**cout << "Enter four values (or -9999 to exit): ";**

**cin >> s1;**

**if (s1 == -9999) {**

**break;**

**}**

**cin >> s2;**

**cin >> s3;**

**cin >> s4;**

**calc\_stats(s1, s2, s3, s4, mean, sigma);**

**cout << "Mean: " << mean << ", Sig: " << sigma << endl;**

**}**

**return 0;**

**}**

A screen shot of a computer code

Description automatically generated

1. Write a program that reads in a length in feet and inches and outputs the equivalent length in meters and centimeters. Use at least three functions: one for input, one or more for calculating, and one for output. Include a loop that lets the user repeat this computation for new input values until the user says he or she wants to end the program. There are 0.3048 meters in a foot, 100 centimeters in a meter, and 12 inches in a foot.

**#include <iostream>**

**using namespace std;**

**const double ft\_to\_meters = 0.3048;**

**const double meters\_to\_cm = 100;**

**const double inches\_to\_ft = 1 / 12.0;**

**bool get\_user\_input(double &ft, double &in);**

**void imperial\_to\_metric(double ft, double in, double &meters, double &cm);**

**void output\_calculation(double meters, double cm);**

**A black screen with white text

Description automatically generatedint main()**

**{**

**double ft, in = 0;**

**double meters, cm = 0;**

**while (get\_user\_input(ft, in)) {**

**imperial\_to\_metric(ft, in, meters, cm);**

**output\_calculation(meters, cm);**

**}**

**return 0;**

**}**

**bool get\_user\_input(double &ft, double &in)**

**{**

**cout << "Enter feet and inches (or -1 to exit): ";**

**cin >> ft;**

**if (ft == -1) {**

**return false;**

**}**

**cin >> in;**

**return true;**

**}**

**void imperial\_to\_metric(double ft, double in, double &meters, double &cm)**

**{**

**meters = ft \* ft\_to\_meters;**

**cm = in \* inches\_to\_ft \* ft\_to\_meters \* meters\_to\_cm;**

**}**

**void output\_calculation(double meters, double cm)**

**{**

**cout << "Meters: " << meters << ", Centimeters: " << cm << "\n\n";**

**}**

1. Write a program like that of the previous exercise that converts from meters and centimeters into feet and inches. Use functions for the subtasks.

**#include <iostream>**

**using namespace std;**

**const double ft\_to\_meters = 0.3048;**

**const double meters\_to\_cm = 100;**

**const double inches\_to\_ft = 1 / 12.0;**

**bool get\_user\_input(double &meters, double &cm);**

**void metric\_to\_imperial(double meters, double cm, double &ft, double &in);**

**void output\_calculation(double ft, double in);**

A black screen with white text

Description automatically generated

**int main()**

**{**

**double meters, cm = 0;**

**double ft, in = 0;**

**while (get\_user\_input(meters, cm)) {**

**metric\_to\_imperial(meters, cm, ft, in);**

**output\_calculation(ft, in);**

**}**

**return 0;**

**}**

**bool get\_user\_input(double &meters, double &cm)**

**{**

**cout << "Enter meters and centimeters (or -1 to exit): ";**

**cin >> meters;**

**if (meters == -1) {**

**return false;**

**}**

**cin >> cm;**

**return true;**

**}**

**void metric\_to\_imperial(double meters, double cm, double &ft, double &in)**

**{**

**ft = meters \* (1.0 / ft\_to\_meters);**

**in = cm \* (1.0 / meters\_to\_cm) \* (1.0 / ft\_to\_meters) \***

**(1.0 / inches\_to\_ft);**

**}**

**void output\_calculation(double ft, double in)**

**{**

**cout << "Feet: " << ft << ", Inches: " << in << "\n\n";**

**}**

1. (You should do the previous two Practice Programs before doing this one.) Write a program that combines the functions in the previous two Practice Programs. The program asks the user if he or she wants to convert from feet and inches to meters and centimeters or from meters and centimeters to feet and inches. The program then performs the desired conversion. Have the user respond by typing the integer 1 for one type of conversion and 2 for the other conversion. The program reads the user’s  
   answer and then executes an *if-else* statement. Each branch of the *if-else* statement will be a function call. The two functions called in the *if-else* statement will have function definitions that are very similar  
   to the programs for the previous two Practice Programs. Thus, they will be function definitions that call other functions in their function bodies. Include a loop that lets the user repeat this computation for new input values until the user says he or she wants to end the program.

**#include <iostream>**

**using namespace std;**

**const double ft\_to\_meters = 0.3048;**

**const double meters\_to\_cm = 100;**

**const double inches\_to\_ft = 1 / 12.0;**

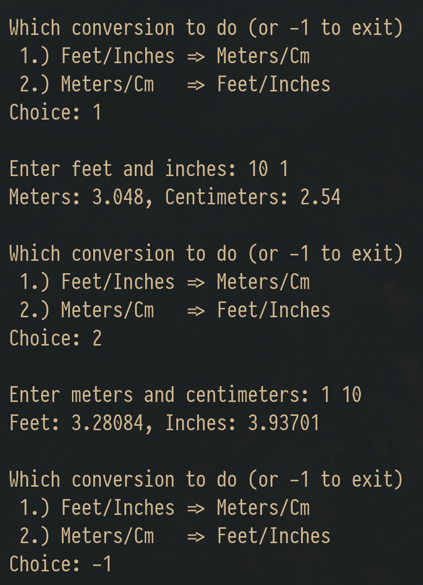
**int get\_user\_input(double &ft, double &in, double &meters, double &cm);**

**void imperial\_to\_metric(double ft, double in, double &meters, double &cm);**

**void metric\_to\_imperial(double meters, double cm, double &ft, double &in);**

**void output\_calculation(double quantity1, double quantity2,**

**int conversion\_type);**



**int main()**

**{**

**cout << endl;**

**double ft, in = 0;**

**double meters, cm = 0;**

**int conversion\_type = -1;**

**while ((conversion\_type = get\_user\_input(ft, in, meters, cm)) != -1) {**

**if (conversion\_type == 1) {**

**imperial\_to\_metric(ft, in, meters, cm);**

**output\_calculation(meters, cm, conversion\_type);**

**} else if (conversion\_type == 2) {**

**metric\_to\_imperial(meters, cm, ft, in);**

**output\_calculation(ft, in, conversion\_type);**

**}**

**}**

**return 0;**

**}**

**int get\_user\_input(double &ft, double &in, double &meters, double &cm)**

**{**

**int choice;**

**cout << "Which conversion to do (or -1 to exit)"**

**<< "\n";**

**cout << " 1.) Feet/Inches => Meters/Cm"**

**<< "\n";**

**cout << " 2.) Meters/Cm => Feet/Inches"**

**<< "\n";**

**cout << "Choice: ";**

**cin >> choice;**

**cout << "\n";**

**if (choice == 1) {**

**cout << "Enter feet and inches: ";**

**cin >> ft >> in;**

**} else if (choice == 2) {**

**cout << "Enter meters and centimeters: ";**

**cin >> meters >> cm;**

**}**

**return choice;**

**}**

**void imperial\_to\_metric(double ft, double in, double &meters, double &cm)**

**{**

**meters = ft \* ft\_to\_meters;**

**cm = in \* inches\_to\_ft \* ft\_to\_meters \* meters\_to\_cm;**

**}**

**void metric\_to\_imperial(double meters, double cm, double &ft, double &in)**

**{**

**ft = meters \* (1.0 / ft\_to\_meters);**

**in = cm \* (1.0 / meters\_to\_cm) \* (1.0 / ft\_to\_meters) \***

**(1.0 / inches\_to\_ft);**

**}**

**void output\_calculation(double quantity1, double quantity2,**

**int conversion\_type)**

**{**

**if (conversion\_type == 1) {**

**cout << "Meters: " << quantity1 << ", Centimeters: " << quantity2**

**<< "\n\n";**

**} else if (conversion\_type == 2) {**

**cout << "Feet: " << quantity1 << ", Inches: " << quantity2**

**<< "\n\n";**

**}**

**}**