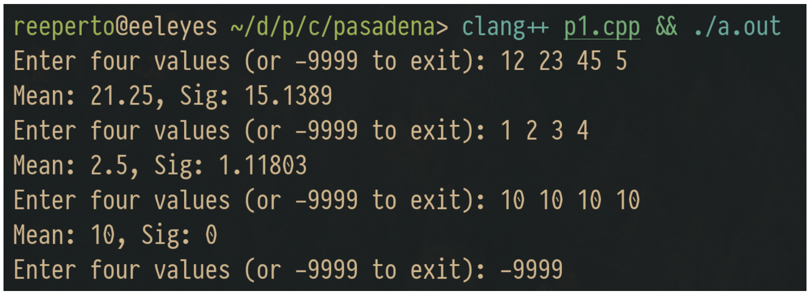
Eli Griffiths and Daniel Su
Checked by groups 1 and 3

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: $(s_i - a)^2$, where a is average of the four scores s_1, s_2, s_3, and s_4. 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.

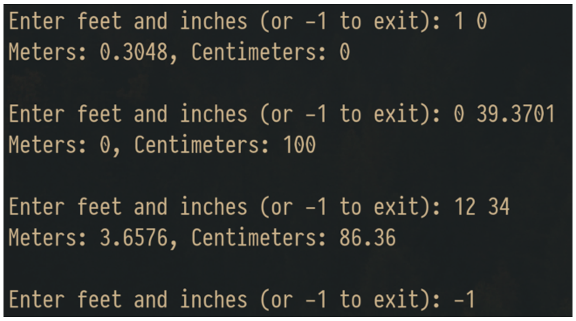
```
#include <iostream>
using namespace std;

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;
}
```



```
reeperto@eeleyes ~/d/p/c/pasadena> clang++ p1.cpp && ./a.out
Enter four values (or -9999 to exit): 12 23 45 5
Mean: 21.25, Sig: 15.1389
Enter four values (or -9999 to exit): 1 2 3 4
Mean: 2.5, Sig: 1.11803
Enter four values (or -9999 to exit): 10 10 10 10
Mean: 10, Sig: 0
Enter four values (or -9999 to exit): -9999
```

2. 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.

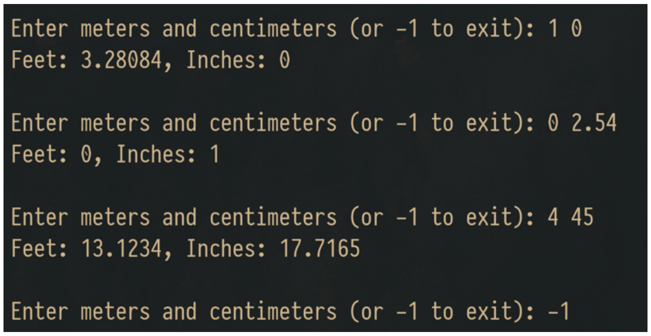
```
#include <iostream>
using namespace std;

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);

int 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";
}
```



```
Enter feet and inches (or -1 to exit): 1 0
Meters: 0.3048, Centimeters: 0

Enter feet and inches (or -1 to exit): 0 39.3701
Meters: 0, Centimeters: 100

Enter feet and inches (or -1 to exit): 12 34
Meters: 3.6576, Centimeters: 86.36

Enter feet and inches (or -1 to exit): -1
```

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

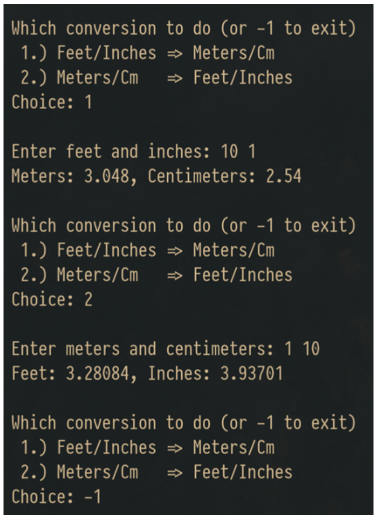
```
#include <iostream>
using namespace std;

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);

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";
}
```



```
Enter meters and centimeters (or -1 to exit): 1 0
Feet: 3.28084, Inches: 0

Enter meters and centimeters (or -1 to exit): 0 2.54
Feet: 0, Inches: 1

Enter meters and centimeters (or -1 to exit): 4 45
Feet: 13.1234, Inches: 17.7165

Enter meters and centimeters (or -1 to exit): -1
```

4. (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;
}
```



```
Which conversion to do (or -1 to exit)
 1.) Feet/Inches => Meters/Cm
 2.) Meters/Cm   => Feet/Inches
Choice: 1

Enter feet and inches: 10 1
Meters: 3.048, Centimeters: 2.54

Which conversion to do (or -1 to exit)
 1.) Feet/Inches => Meters/Cm
 2.) Meters/Cm   => Feet/Inches
Choice: 2

Enter meters and centimeters: 1 10
Feet: 3.28084, Inches: 3.93701

Which conversion to do (or -1 to exit)
 1.) Feet/Inches => Meters/Cm
 2.) Meters/Cm   => Feet/Inches
Choice: -1
```

```
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";
    }
}
```