Lab03

LAB # 3

SECTION # Tuesdays @2:10

Thriambak Giriprakash

2/23/2021

Problem #1

Implement functions which perform unit conversions. The purpose being to learn how to properly implement functions within code.

Analysis

Researched unit conversion values, for example, 1 meter is approximately 3.4 feet.

Design

Implement unit conversion calculation into the functions

Testing

Once the functions were implemented (1), a calculator was used to check the results produced to verify accuracy. They were accurate.

Comments

I learned how to implement a function in this portion of the lab.

Source Code

<Use NPP Exporter to PASTE source code>

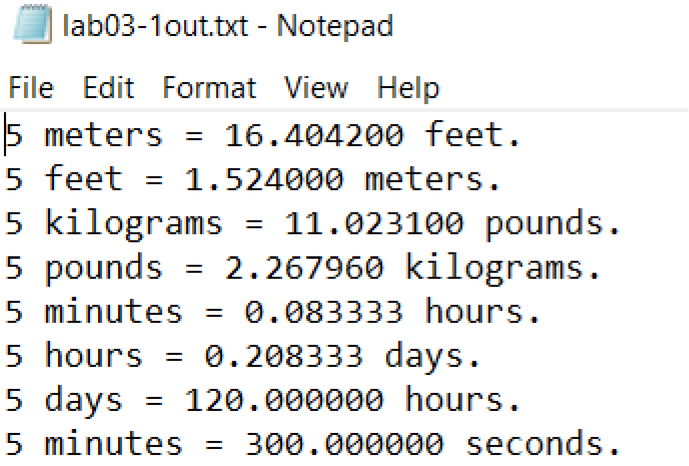
```
/*-------------------------------------------------------------------------
-
-           SE 185: Lab 03 - Introduction to Functions and Branches
-
-   Name:
-
-   Section:
-
-   NetID:
-
-   Date:
-
---------------------------------------------------------------------------
*/

/*-------------------------------------------------------------------------
-
-                               Includes
-
---------------------------------------------------------------------------
*/
#include "stdio.h"

/*-------------------------------------------------------------------------
-
-                               Prototypes
-
---------------------------------------------------------------------------
*/
double meters_to_feet(double meters);

double feet_to_meters(double feet);

double kilograms_to_pounds(double kilograms);

double pounds_to_kilograms(double pounds);

double minutes_to_hours(double minutes);

double hours_to_days(double hours);

double days_to_hours(double days);

double minutes_to_seconds(double minutes);

/*-------------------------------------------------------------------------
-
-                                   Notes
-
---------------------------------------------------------------------------
*/
// Compile with gcc lab03-1.c -o lab03-1
// Run with ./lab03-1
```

```c
/*---------------------------------------------------------------------------
-
-                               Implementation
-
-----------------------------------------------------------------------------
*/

int main(int argc, char *argv[])
{
    /* Test your code by calling your functions here. */
    printf("5 meters = %lf feet.\n", meters_to_feet(5));
    printf("5 feet = %lf meters.\n", feet_to_meters(5));
    printf("5 kilograms = %lf pounds.\n", kilograms_to_pounds(5));
    printf("5 pounds = %lf kilograms.\n", pounds_to_kilograms(5));
    printf("5 minutes = %lf hours.\n", minutes_to_hours(5));
    printf("5 hours = %lf days.\n", hours_to_days(5));
    printf("5 days = %lf hours.\n", days_to_hours(5));
    printf("5 minutes = %lf seconds.\n", minutes_to_seconds(5));

    return 0;
}

/* Implement the functions below. */

/**
 * Part a.
 * Converts given meters into feet.
 *
 * @param meters
 * @return - How many feet are in the given meters.
 */
double meters_to_feet(double meters)
{

    return meters * 3.28084;
}

/**
 * Part b.
 * Converts given feet into meters.
 *
 * @param feet
 * @return - How many meters are in the given feet.
 */
double feet_to_meters(double feet)
{
    return feet * .3048;
}

/**
 * Part c.
 * Converts given kilograms into pounds.
 *
 * @param kilograms
 * @return - How many pounds are in the given kilograms.
 */
```

```
double kilograms_to_pounds(double kilograms)
{
    return kilograms * 2.20462;
}

/**
 * Part d.
 * Converts given pounds into kilograms.
 *
 * @param pounds
 * @return - How many kilograms are in the given pounds.
 */
double pounds_to_kilograms(double pounds)
{
    return pounds* 0.453592;
}

/**
 * Part e.
 * Converts given minutes into hours.
 *
 * @param minutes
 * @return - How many hours are in the given minutes.
 */
double minutes_to_hours(double minutes)
{
    return minutes/60.0;
}

/**
 * Part f.
 * Converts given hours into days.
 *
 * @param hours
 * @return - How many days are in the given hours.
 */
double hours_to_days(double hours)
{
    return hours / 24.0;
}

/**
 * Part g.
 * Converts given days into hours.
 *
 * @param days
 * @return - How many hours are in the given days.
 */
double days_to_hours(double days)
{
    return days * 24.0;
}

/**
 * Part h.
 * Converts given minutes into seconds.
 *
 * @param minutes
 * @return - How many seconds are in the given minutes.
 */
double minutes_to_seconds(double minutes)
{
    return minutes * 60.0;
}
```

Screen Shots

<Number the screenshots and paste here. The point of numbering the screenshots is so that you can refer to them during your discussion in the various parts above. Alternatively, you can include the screenshots in-line with the text above as part of your discussion.>



```
5 meters = 16.404200 feet.
5 feet = 1.524000 meters.
5 kilograms = 11.023100 pounds.
5 pounds = 2.267960 kilograms.
5 minutes = 0.083333 hours.
5 hours = 0.208333 days.
5 days = 120.000000 hours.
5 minutes = 300.000000 seconds.
```

1.

Problem #2

This portion of the lab was to call a previously written function in a separate header file and to put values into it which the user would input.

Analysis

Looked at the header file to see how to call the function as well as what data type it returned.

Design

Followed the comments written on the code.

Testing

Tested a known Fibonacci sequence value on Cygwin to see if it worked (2)

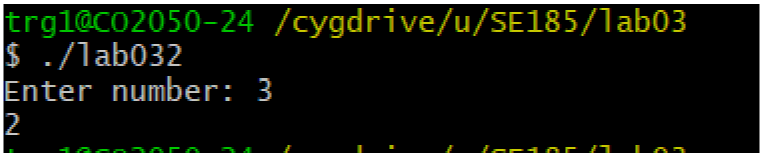
Comments

I learned how to call a function.

Source Code:

```
/*----------------------------------------------------------------------------
-
-           SE 185: Lab 03 - Introduction to Functions and Branches
-
-   Name:
-
-   Section:
-
-   NetID:
-
-   Date:
-
------------------------------------------------------------------------------
*/

/*----------------------------------------------------------------------------
-
-                                Includes
-
------------------------------------------------------------------------------
*/
#include <stdio.h>
#include “lab03.h”

/*----------------------------------------------------------------------------
-
-                                    Notes
-
------------------------------------------------------------------------------
*/
// Compile with gcc lab03-2.c -o lab03-2
// Run with ./lab03-2

/*----------------------------------------------------------------------------
-
-                                Implementation
-
------------------------------------------------------------------------------
*/

int main(int argc, char *argv[])
{
    /* Declare variables here that will be used to store user prompt. */
    int fibNum;
    /* Ask for user input here with a print statement.*/
    printf("Enter number: ");
    /* Scan for user input using scanf. */
    scanf("%d",&fibNum);
    /* Print the requested nth element of the Fibonacci sequence here. */
    printf("%d",fibonacci_sequence(fibNum));
    return 0;
}
```

Screenshots:

2. 

```
trg1@CO2050-24 /cygdrive/u/SE185/lab03
$ ./lab032
Enter number: 3
2
```

Problem #3

This portion of the lab called back on problem 1 and tasked me with creating a more secure unit conversion kit. The code would ask for a letter which corresponded with a certain conversion and once the user inputs that letter and a number, the conversion is performed.

Analysis

No additional analysis required other than study of the output sample provided.

Design

I used a switch case to neatly organize all of the cases in an easy to read manner.

Testing

Tested the sample output provided in the lab document and produced the same results (3)

Comments

This section combined aspects of part 1 and 2 of this lab.

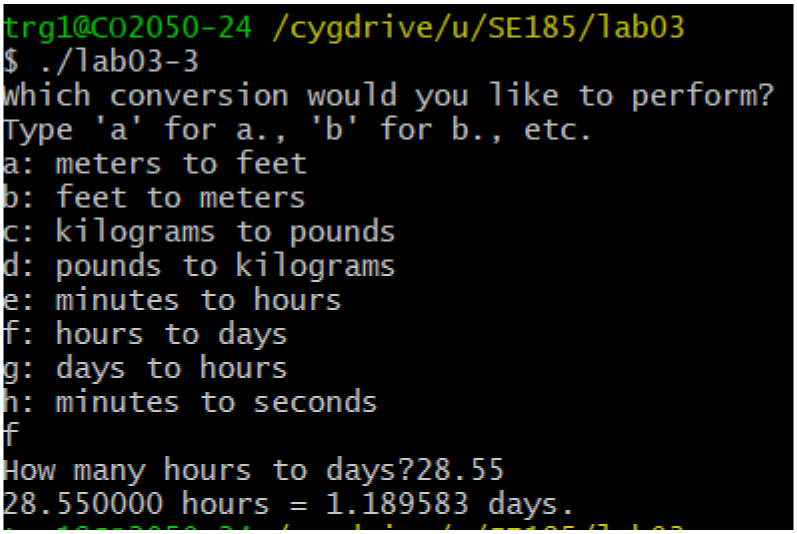
Source Code:

```
/*---------------------------------------------------------------------------
-
-           SE 185: Lab 03 - Introduction to Functions and Branches
-
-   Name:
-
-   Section:
-
-   NetID:
-
-   Date:
-
-----------------------------------------------------------------------------
*/

/*---------------------------------------------------------------------------
-
-                               Includes
-
-----------------------------------------------------------------------------
*/
#include "stdio.h"

/*---------------------------------------------------------------------------
-
-                               Prototypes
-
-----------------------------------------------------------------------------
*/
double meters_to_feet(double meters);

double feet_to_meters(double feet);

double kilograms_to_pounds(double kilograms);

double pounds_to_kilograms(double pounds);

double minutes_to_hours(double minutes);

double hours_to_days(double hours);

double days_to_hours(double days);

double minutes_to_seconds(double minutes);

/*---------------------------------------------------------------------------
-
-                                   Notes
-
-----------------------------------------------------------------------------
*/
// Compile with gcc lab03-1.c -o lab03-1
// Run with ./lab03-1
```

```
/*---------------------------------------------------------------------------
-
-                               Implementation
-
-----------------------------------------------------------------------------
*/

int main(int argc, char *argv[])
{
    /* Test your code by calling your functions here. */
    char sel;
    double convert;

    printf("Which conversion would you like to perform?\nType 'a' for a., 'b'
for b., etc.\na: meters to feet\nb: feet to meters\nc: kilograms to
pounds\nd: pounds to kilograms\ne: minutes to hours\nf: hours to days\ng:
days to hours\nh: minutes to seconds\n");
    scanf("%c",&sel);

    switch (sel)
    {
        case 'a':
            printf("How many meters to feet?");
            scanf("%lf",&convert);
            printf("%lf meters = %lf feet.",convert,
meters_to_feet(convert));
            break;

        case 'b':
            printf("How many feet to meters?");
            scanf("%lf",&convert);
            printf("%lf feet = %lf meters.",convert,
feet_to_meters(convert));
            break;

        case 'c':
            printf("How many kilograms to pounds?");
            scanf("%lf",&convert);
            printf("%lf kilograms = %lf pounds.",convert,
kilograms_to_pounds(convert));
            break;

        case 'd':
            printf("How many pounds to kilograms?");
            scanf("%lf",&convert);
            printf("%lf pounds = %lf kilograms.",convert,
pounds_to_kilograms(convert));
            break;

        case 'e':
            printf("How many minutes to hours?");
            scanf("%lf",&convert);
            printf("%lf minutes = %lf hours.",convert,
minutes_to_hours(convert));
            break;

        case 'f':
            printf("How many hours to days?");
            scanf("%lf",&convert);
            printf("%lf hours = %lf days.",convert, hours_to_days(convert));
            break;

        case 'g':
            printf("How many days to hours?");
            scanf("%lf",&convert);
            printf("%lf days = %lf hours.",convert, days_to_hours(convert));
            break;

        case 'h':
            printf("How many minutes to seconds?");
            scanf("%lf",&convert);
            printf("%lf minutes = %lf seconds.",convert, 
minutes_to_seconds(convert));
            break;


    }

    return 0;
}

/* Implement the functions below. */

/**
 * Part a.
 * Converts given meters into feet.
 *
 * @param meters
 * @return - How many feet are in the given meters.
 */
double meters_to_feet(double meters)
{

    return meters * 3.28084;
}

/**
 * Part b.
 * Converts given feet into meters.
 *
 * @param feet
 * @return - How many meters are in the given feet.
 */
double feet_to_meters(double feet)
{
    return feet * .3048;
}

/**
 * Part c.
 * Converts given kilograms into pounds.
 *
 * @param kilograms
 * @return - How many pounds are in the given kilograms.
 */
double kilograms_to_pounds(double kilograms)
{
    return kilograms * 2.20462;
}

/**
 * Part d.
 * Converts given pounds into kilograms.
 *
 * @param pounds
 * @return - How many kilograms are in the given pounds.
 */
double pounds_to_kilograms(double pounds)
{
    return pounds* 0.453592;
}

/**
 * Part e.
 * Converts given minutes into hours.
 *
 * @param minutes
 * @return - How many hours are in the given minutes.
 */
double minutes_to_hours(double minutes)
{
    return minutes/60.0;
}

/**
 * Part f.
 * Converts given hours into days.
 *
 * @param hours
 * @return - How many days are in the given hours.
 */
double hours_to_days(double hours)
{
    return hours / 24.0;
}

/**
 * Part g.
 * Converts given days into hours.
 *
 * @param days
 * @return - How many hours are in the given days.
 */
double days_to_hours(double days)
{
    return days * 24.0;
}

/**
 * Part h.
 * Converts given minutes into seconds.
 *
 * @param minutes
 * @return - How many seconds are in the given minutes.
 */
double minutes_to_seconds(double minutes)
{
    return minutes * 60.0;
}
```

Screenshots:

3.



```
trg1@CO2050-24 /cygdrive/u/SE185/lab03
$ ./lab03-3
Which conversion would you like to perform?
Type 'a' for a., 'b' for b., etc.
a: meters to feet
b: feet to meters
c: kilograms to pounds
d: pounds to kilograms
e: minutes to hours
f: hours to days
g: days to hours
h: minutes to seconds
f
How many hours to days?28.55
28.550000 hours = 1.189583 days.
```