Programming Fundamental – ENSF 337
Lab 1
M. Moussavi
Jay Chuang
B01
September 14, 2019

```
/* lab1exe_C.c
 * ENSF 337 Fall 2019 Lab 1 Exercise C
 *
 * Completed by:    Jay Chuang
 * Lab Section:     B01
 */

#include <stdio.h>
#include <stdlib.h>

int are_the_same(int x, int y, int z);
int non_decreasing(int x, int y, int z);

int main(void)
{
    int a, b, c, nscan;
    do{

        /* Get values for a, b, and c from user.    */
        /* Quit if the user enters garbage.     */
        printf("\nPlease enter a positive value for the int variable a: ");
        nscan = scanf("%d", &a);
        if (nscan != 1) {
            printf("Your input could not be converted to an int.  I quit.\n\n");
            exit(1);
        }

        if(a < 0) break;
```

```
printf("\nPlease enter a positive value for the int variable b: ");
nscan = scanf("%d", &b);
if (nscan != 1) {
    printf("Your input could not be converted to an int.  I quit.\n\n");
    exit(1);
}
if(b < 0) break;

printf("\nPlease enter a positive value for the int variable c: ");
nscan = scanf("%d", &c);
if (nscan != 1) {
    printf("Your input could not be converted to an int.  I quit.\n\n");
    exit(1);
}

if(c < 0) break;

printf("The numbers are: a = %d, b = %d, c = %d.", a, b, c);

if(are_the_same(a, b, c))
   printf(" They are the same\n\n");
else
    printf(" They are NOT the same\n\n");



// The following 4 lines must be uncommented by the students when testing
// funciton non_decreasing

 if(non_decreasing (a, b, c))
    printf(" And they are in non-decreasing order\n\n");
```

```
          else
             printf(" And they are NOT in non-decreasing order\n\n");

    } while(1);

    printf("\nProgram terminated....");

    return 0;
}

int are_the_same(int x, int y, int z)
{
    if (x == y && x == z && y == z)
        return 1;
    return 0;
}

int non_decreasing(int x, int y, int z)
{
    // Students must remvoe the following line and add the right code
    if (x<=y) {
        if (y<=z) {
            return 1;
        }
    }
    return 0;
}
```

OUTPUT EXERCISE C

```
Please enter a positive value for the int variable a:

Please enter a positive value for the int variable b:

Please enter a positive value for the int variable c: The numbers are: a = 2, b =
2, c = 2. They are the same


 And they are in non-decreasing order



Please enter a positive value for the int variable a:

Please enter a positive value for the int variable b:

Please enter a positive value for the int variable c: The numbers are: a = 3, b =
3, c = 3. They are the same


 And they are in non-decreasing order



Please enter a positive value for the int variable a:

Please enter a positive value for the int variable b:

Please enter a positive value for the int variable c: The numbers are: a = 3, b =
5, c = 6. They are NOT the same


 And they are in non-decreasing order



Please enter a positive value for the int variable a:

Program terminated....
```

```
/*
*  lab1exe_D.c
*  ENSF - Fall 2019 Lab 1, exercise D
*  Completed by: Jay Chuang
*/

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#define G 9.8   /* gravitation acceleration 9.8 m/s^2 */
#define PI 3.141592654

void create_table(double v);
double Projectile_travel_time(double a, double v);
double Projectile_travel_distance(double a, double v);
double degree_to_radian(double d);

void create_table(double v)
{
    double time, distance, angle;

    printf("Angle                  t                  d \n(deg)
(sec)               (m)");

    for (angle=0; angle<=90; angle+=5)
    {
        time = Projectile_travel_time(angle,v);
        distance = Projectile_travel_distance(angle,v);

        printf ("\n%lf           %lf           %lf", angle, time, distance);
```

```
    }
}
double Projectile_travel_time(double a, double v)
{
    double time, angle;

    angle = degree_to_radian(a);
    time = (2*v*sin(angle))/G;

    return time;
}
double Projectile_travel_distance(double a, double v)
{
    double distance, angle;

    angle = degree_to_radian(a);
    distance = (pow(v,2)/G)*sin(2*angle);

    return distance;
}
double degree_to_radian(double d)
{
    double angle;

    angle = (d*PI)/180;

    return angle;
}

int main(void)
```

```
{
    int n;
    double velocity;

    printf ("please enter the velocity at which the projectile is launched
(m/sec): \n");
    n = scanf("%lf" ,&velocity);

    if(n != 1)
    {
        printf("Invlid input. Bye...");
        exit(1);
    }

    while (velocity < 0 )
    {
        printf ("please enter a positive number for velocity: ");
        n = scanf("%lf", &velocity);
        if(n != 1)
        {
            printf("Invlid input. Bye...");
            exit(1);
        }
    }
        create_table(velocity);

    return 0;
}

/* UNCOMMENT THE CALL TO THE create_table IN THE main FUNCTION, AND COMPLETE THE
PROGRAM */
```

OUTPUT EXERCISE D

```
please enter the velocity at which the projectile is launched (m/sec):
Angle                  t                   d
(deg)                (sec)               (m)
0.000000           0.000000           0.000000
5.000000           1.778689           177.192018
10.000000           3.543840           349.000146
15.000000           5.282021           510.204082
20.000000           6.980003           655.905724
25.000000           8.624862           781.678003
30.000000           10.204082           883.699392
35.000000           11.705642           958.870021
40.000000           13.118114           1004.905870
45.000000           14.430751           1020.408163
50.000000           15.633560           1004.905870
55.000000           16.717389           958.870021
60.000000           17.673988           883.699391
65.000000           18.496077           781.678003
70.000000           19.177400           655.905724
75.000000           19.712772           510.204081
80.000000           20.098117           349.000146
85.000000           20.330504           177.192018
90.000000           20.408163           -0.000000
```