

Course: Programming Fundamentals – ENSF 337

Lab #: Lab 1

Instructor: M. Moussavi

Student Name: Derek Braun, 30040032

Lab Section: B01

Date Submitted: Sept. 18, 2018

Exercise C: Using Logical Expressions in C

```
/* lab1exe_C.c
 * ENSF 337 Fall 2018 Lab 1 Exercise C
 *
 * Completed by:  Derek Braun, 30040032
 * Lab Section:  B01
 */

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

int are_the_same(int x, int y, int z);
/* REQUIRES: x, y, and z are positive integers.
 * PROMISES: returns 1 if all three arguments are equal. Otherwise returns zero.
 */

int non_decreasing(int x, int y, int z);
/* REQUIRES: x, y, and z are positive integers.
 * PROMISES: returns 1, if:  $x \leq y \leq z$ . Otherwise returns zero.
 */

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);
        }
    } while(are_the_same(a, b, c) || non_decreasing(a, b, c));
}
```

```

    }

    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 remove the following line and add the right code
    if(x <= y && y <= z){
        return 1;
    }
    else{
        return 0;
    }
}

```

Output:

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....

Exercise D: Projectile Time and Distance Calculator

```
/*
 * lab1exe_D.c
 * ENSF - Fall 2018 Lab 1, exercise D
 * Completed by: Derek Braun, 30040032, B01
 */

#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);

int main(void)
{
    int n;
    double velocity;
```

```

printf ("please enter the velocity at which the projectile is launched (m/sec): ");
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;
}

void create_table(double v){
    int deg = 0;
    double rad;
    double time;
    double dist;

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

    while(deg <= 90){
        rad = degree_to_radian(deg);
        time = Projectile_travel_time(rad, v);
        dist = Projectile_travel_distance(rad, v);
        printf("%lf \t %lf \t %lf\n", rad, time, dist);

        deg += 5;
    }
}

double Projectile_travel_time(double a, double v){
    double t;
    t = 2*v*sin(a);
    t = t/G;
    return t;
}

double Projectile_travel_distance(double a, double v){
    double d;
    d = pow(v, 2)/G;

```

```

        d = d*sin(2*a);
        return d;
    }

double degree_to_radian(double d){
    return ((PI/180)*d);
}

```

Output:

please enter the velocity at which the projectile is launched (m/sec): 42

Angle (deg)	t (sec)	d (m)
0.000000	0.000000	0.000000
0.087266	0.747049	31.256672
0.174533	1.488413	61.563626
0.261799	2.218449	90.000000
0.349066	2.931601	115.701770
0.436332	3.622442	137.888000
0.523599	4.285714	155.884573
0.610865	4.916369	169.144672
0.698132	5.509608	177.265396
0.785398	6.060915	180.000000
0.872665	6.566095	177.265396
0.959931	7.021303	169.144672
1.047198	7.423075	155.884573
1.134464	7.768352	137.888000
1.221730	8.054508	115.701770
1.308997	8.279364	90.000000
1.396263	8.441209	61.563626
1.483530	8.538812	31.256672
1.570796	8.571429	-0.000000