**ECE 3331, Dr. Hebert, Spr 2017 HW 11 due 04/18 at 11:59 pm**

**Problem 1. Section 10.1, Exercise 9**

Struct animal temp, menagerie[200];

**Problem 2. Section 10.1, Exercise 24**

D, E

**Problem 3. Section 10.2, Exercise 3**

#define exp\_c( expc, c ) ( \

Expc.real = exp( c.real ) \* cos( c.imag ), \

Expc.imag = exp( c.real ) \* sin( c. imag ) )

**Problem 4. Section 10.3, Exercise 1**

Typedef struct robot {

Char name[15];

Int limbs;

Int joints;

Float weight;

Float max\_speed;

Char bad\_habits[20][100];

} Robot;

Robot r2, d2;

**Problem 5. Section 10.3, Exercise 3**

house is a tag for a struct while HOUSE is a synonym for struct house

**Problem 6. Section 10.4, Exercise 1**

BIKES bike1 = { “brand”, 32, 123, 2.1, 3.2, 13.24};

BIKES bike2;

Bike2.brand\_name = bike1.brand\_name;

Bike2.spokes\_per\_wheel = bike1.spokes\_per\_wheel;

Bike2.links\_in\_chain = bike1.links\_in\_chain;

Bike2.height = bike1.height;

Bike2.length = bike1.length;

Bike2.price = bike1.price;

**Problem 7. Section 10.4, Exercise 3**

yes

**Problem 8. Section 10.4, Exercise 5**

Struct test {

Char array[2][2];

Int variable;

Float number;

};

Struct test

T\_1 = { ‘A’, ‘B’, ‘C’, ‘D’, 32, 32.3},

T\_2 = {‘L’, ‘J’, ‘K’, ‘M’, 65, 99.3};

**Problem 9. Section 10.5, Exercise 1**

Ptr = &soldier3;

**Problem 10. Chapter 10, Programming Exercises problem 10.4 pg 545.**

--------------------------HEADER FILE-----------------------------------

struct vector {

double x;

double y;

double z;

};

#define lengths(l1, l2, l3, a, b) ( \

l1 = sqrt(a.x\*a.x + a.y\*a.y + a.z\*a.z), \

l2 = sqrt(b.x\*b.x + b.y\*b.y + b.z\*b.z), \

l3 = sqrt((b.x-a.x)\*(b.x-a.x) + (b.y-a.y)\*(b.y-a.y) + (b.z-a.z)\*(b.z-a.z)))

#define dot\_product(d, a, b) d = a.x\*b.x + a.y\*b.y + a.z\*b.z

#define angle(ang, a, b) ang = (a.x\*b.x + a.y\*b.y + a.z\*b.z)/(sqrt(a.x\*a.x + a.y\*a.y + a.z\*a.z)+sqrt(b.x\*b.x + b.y\*b.y + b.z\*b.z))

#define cross\_product( c, a, b ) ( \

c.x = a.y\*b.z - a.z\*b.y, \

c.y = a.z\*b.x - a.x\*b.z, \

c.z = a.x\*b.y - a.y\*b.x )

#define read\_v( v ) scanf( "%lf%lf%lf", &v.x, &v.y, &v.z )

#define print\_v( v ) printf("%f, %f, %f", v.x, v.y, v.z )

---------------------------------------------MAIN FILE---------------------------------------------------

#include <stdio.h>

#include <stdlib.h>

#include "vector.h"

int main()

{

struct vector a, b, c;

float l1, l2, l3, dot, ang;

printf( "Enter v: " );

read\_v( a );

printf( "Enter b: " );

read\_v( b );

lengths( l1, l2, l3, a, b );

printf("\n%f\n%f\n%f", l1, l2, l3 );

dot\_product(dot, a, b);

printf("\nDot Product: %f", dot);

angle(ang, a, b);

printf("\nAngle: %f", ang);

cross\_product(c, a, b);

printf("\n");

print\_v(c);

}

**Problem 11. Section 10.6, Exercise 3.**

TV\_SHOW get\_tv\_data1(void)

{

TV\_SHOW c;

gets( c.title );

gets( c.network );

scanf(“%f”, &c.rating);

scanf(“%f”, &c.share);

return c;

}

**Problem 12. Section 10.6, Exercise 5.**

Void get\_tv\_data2(TV\_SHOW\* data)

{

gets( data -> title);

gets(data -> network);

scanf(“%f”, &(data -> rating));

scanf(“%f”, &(data -> share));

}

**Problem 13. Section 10.6, Exercise 6**

Typedef struct job {

Char\* employer;

Char\* duties;

Float wage;

} JOB;

**Problem 14. Section 10.8, Exercise 2.**

A structure holds multiple types of data simultaneously while a union can hold multiple types of data but only one at a time.

**Problem 15. Section 10.8, Exercise 5.**

If only one variable is required at a time then a union may be preferable over a struct.