ENEE 140, Spring 2015
Midterm Exam

Tuesday, March 31, 2015, 3:30 – 4:45 pm

University of Maryland Honor Pledge

The University is committed to Academic Integrity, and has a nationally recognized Honor Code, administered by the Student Honor Council. In an effort to affirm a community of trust, the Student Honor Council proposed and the University Senate approved an Honor Pledge. The University of Maryland Honor Pledge Reads:

> "I pledge on my honor that I have not given or received any unauthorized assistance on this examination (or assignment)"

Please write the exact wording of the Pledge, followed by your signature, in the space below:

Pledge: ____

Pledge: ____

Pledge: ____

Pledge: ____

Your signature: ______________________________________

Full Name: __________________ Section: ________ Directory ID: ______________

1 (14):	
2 (24):	
3 (5):	
4 (15):	
5 (12):	
6 (10):	
7 (20):	
TOTAL (100):	

Instructions:

- Make sure that your exam is not missing any sheets, then write your full name, your section and your Directory ID on the front.
- Write your answers in the space provided below the problem. If you make a mess, clearly indicate your final answer.
- The exam has a maximum score of 100 points.
- The problems are of varying difficulty. The point value of each problem is indicated. Pile up the easy points quickly and then come back to the harder problems.
- This exam is OPEN BOOK. You may use any books or notes you like. Calculators are allowed, but no other electronic devices. Good luck!

Problem 1. (14 points)

This problem tests your understanding of C types and casts. Explain whether the following variables are implicitly or explicitly casted, and what type they are casted into.

```
int a = 20, b = 25, c;
float d = 1.5, e = -2.5, f;

1. c = a * e;
2. c = (float)a * b;
3. f = (int)d * e;
```

1.	
2.	
3.	

Problem 2. (24 points)

This problem tests your understanding of C types and casts. Assume that variables a, b, c and d are defined as followed:

```
int         a = 11;
unsigned    b = 3;
float       c = 1;
float       d = 2;
```

Fill in all the empty cells in the table below. For each of the C assignment expressions in the left column, state the resulting value of the r2 – r9 variables.

Assignment	Value
`float  r1 = c/d;`	0.5
`int    r2 = a/b;`	
`int    r3 = a%b;`	
`float  r4 = a/b;`	
`int    r5 = (c+d) / 2 + 0.5;`	
`int    r6 = UINT_MAX - 1;`	
`int    r7 = (a - r3) % b;`	
`int    r8 = r3 % b;`	
`int    r9 = (UINT_MAX % 2 + 1) % 2;`	

Problem 3. (5 points)

This problem tests your understanding of C characters and functions. Write the function called `function1` that will take as a parameter an integer from 1–26 and return the lowercase letter that it corresponds to (ex: 1 -> a, 10 -> j, etc.).

```
int main()
{
      int num;
      printf("Enter Value between 1 and 26: ");
      scanf("%d",&num);

      if (d >= 1 && d <= 26) {
            printf("%c",function1(num));
      }

      return 0;
}
```

Problem 4. (15 points)

This problem tests your understanding of loops and if statements.

a. Fill in the blanks to print out the sum of integers up to the int a in the blank ('1 + 2 + 3 + ... + 9 = 45').

```
#include <stdio.h>

int main() {

int sum, i, a=9;

sum = 0;
for (_____________) {
    sum = ________;
    if (______) {
     printf(_____);
    } else {
        printf(_______);
    }
}

return 0;
}
```

b. Rewrite the loop as a while loop.

Problem 5. (12 points)

This problem tests your understanding of loops. The following program prompts the user to enter an integer, then prints a triangular structure made up of * characters.

Examples:

If the user enters 5, the program outputs: * ** *** **** *****	If the user enters 3, the program outputs: * ** ***

Fill in the blanks to complete the program:

```
#include <stdlib.h>

int main()
{
      int i,j;
      int max;

      scanf("%d", &max);

      for (____;____;____) {
            for (____;____;____)  {
                  printf("*");
            }

            printf("\n");
      }
}
```

Problem 6. (10 points)

This problem tests your understanding of random number generation. Write a snippet of code that assigns a random even number between 2 and 16 (including both 2 and 16) to variable `x`.

You may assume that `srand()` has already been invoked and that RAND_MAX = $2^n - 1$ (for some value of n).

```
int x;
```

Problem 7. (20 points)

This problem tests your understanding of C functions and integer arithmetic. Consider a program that determines whether a given date corresponds to a weekday or a weekend. The program prompts the user to specify the date by entering the number of days elapsed since January 1st, 2015. The program determines the correct day of the week using a helper function, which returns 0 for Monday, 1 for Tuesday, 2 for Wednesday, etc. This function takes into account the fact that January 1st, 2015 was a Thursday.

The implementation below contains several bugs. List all the bugs that you can find.

```
#include <stdio.h>

int
main()
{
      int days, wd;

      printf ("Enter number of days since January 1, 2015: ");
      scanf("%d", &days);

      wd = day_of_week(days);

      if (wd <= 5) {
            printf("Weekday\n");
      } else{
            printf("Weekend\n");
      }

      return 0;
}


int day_of_week(int days_since_new_year)

// Returns 0 for Monday, 1 for Tuesday, ...
int
day_of_week(int days_since_new_year)
{
      int thursday_offset = 3;
      (thursday_offset + days_since_new_year) / 7;
}
```