ENEE 140, Spring 2025
Midterm Exam

Date:

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: _____ Course: _____ Directory ID: _____

List of Exam Questions:

Question:	1	2	3	4	5	6	7	8	Total
Points:	16	8	18	16	16	18	8	12	112
Score:									

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 name and section at the bottom of each page as well.
- Write your answers in the space provided below the problem. If you make a mess, clearly indicate your final answer.

- The last question is for EXTRA CREDIT and is worth 12 points. You may receive full credit without answering it, assuming that you answered correctly all the other questions (the exam will be scored out 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. No electronic devices (e.g. laptops, tablets, smartphones, calculators) are allowed. Good luck!

1. (16 points) This problem tests your understanding of C types and casts and of C operators. Assume that variables `a`, `b`, `c` and `d` are defined as follows:

```
int a = 7;
float b = 3.5;
char c = 'A';
unsigned d = 0;
```

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. If an expression results in a compilation error, write ERROR. You may use what you like from limits.h (ex. INT_MIN)

Assignment		Value
unsigned	r1 =++d;	1
unsigned	r2 = a−8;	_____
float	r3 =b/a;	_____
char	r4 = c++;	_____
int	r5 = a % 3;	_____
int	r6 = (**int**)b∗a;	_____
unsigned	r7 = UINT_MAX %2 + 1;	_____
float	r8 = a/(d+2.0);	_____
int	r9 = a % b;	_____

2. (8 points) This is a four-part multiple choice. These questions test the concepts of functions in C.

   (a) In C, what is a function primarily used for?

   A. Decision making

   B. Variable declaration

   C. Printing output

   D. Code organization and reusability

   (a) _____

   (b) What function call returns a integer?

   A. **float** a()

   B. **void** b()

   C. **int** c()

   D. **char** d()

   (b) _____

   (c) In C, can you access a functions local variables in main?

   A. No

   B. Yes

   (c) _____

   (d) What is a function parameter in C?

   A. A global variable

   B. A variable passed to a function when it is called

   C. A reserved keyword

   D. A variable declared within a function

   (d) _____

3. (18 points) This problem tests your understanding of `if-else` statements. What is the output of the following function?

```
#include <stdio.h>

int main() {
    int a = 1, b = 2, c = 3, d = 4, e = 5, f = 6;

    if (a % 2 == 0) {
        b = a + 3;
        c = b * 2;
    } else {
        d = c + 4;
        e = d - 2;
    }

    if (b % 3 == 0) {
        f = c + d;
        a = e * 2;
    } else {
        c = a + b;
        d = f - 1;
    }

    e = a + b + c;
    f = d + e;

    printf("a is %d, b is %d, c is %d, d is %d, e is %d, f is %d\n",
    a, b, c, d, e, f);
    return 0;
}
```

4. (16 points) This problem tests your understanding of C loops, `if-else` statements, prefix and postfix increment and decrement operators, and integer and floating point arithmetic. Write the output of the following program.

```
1  #include <stdio.h>
2
3  int main() {
4
5      int i = 0;
6
7      while (i < 10) {
8          if (i < 3) {
9              printf("%d\n", i++);
10             i++;
11         }
12         else if (i > 3 && i < 6) {
13             printf("%.2f\n", i++/3.0);
14         }
15         else if (i == 6) {
16             printf("%d\n", ++i/4);
17             i++;
18         }
19         else if (i > 8) {
20             printf("%d\n", ++i);
21         }
22         else {
23             printf("%d\n", --i);
24             i+=2;
25         }
26     }
27
28     return 0;
29 }
```

5. (16 points) This question tests your understanding of character manipulation. What is the output of this program?

```
1
2  #include <stdio.h>
3
4  int main() {
5
6      char c = 'B', d = '7';
7
8      printf("%c", c);
9
10     d -= 7;
11
12     printf("%c", d);
13
14     c += 'a'-'A' + 16;
15
16     printf("%c%c", c, c);
17
18     d += 3;
19
20     printf("%c", d);
21
22     c += 'A'-'a' - 16;
23
24     printf("%c", c);
25
26     c += 18 + 'a'-'A';
27
28     printf("%c!\n", c);
29
30     return 0;
31 }
```

6. (18 points) The following code tests your understanding of psuedo-random number generation. Give a variable type for the first box and a numerical value for each of the second boxes.

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

int main(void) {
    srand(time(NULL));
    int a = 1;
    int b = 8;
    float c = 5.4;

    printf("Number 1 is: %d\n", a + rand() % (b-a+1));

    printf("Number 2 is: %d\n", rand() % (b+1));


    printf("Number 3 is: is %f", c*rand()/RAND_MAX);

    return 0;
}
```

Number 1 is a/an _____ between _____ and _____.
Number 2 is a/an _____ between _____ and _____.
Number 3 is a/an _____ between _____ and _____.

7. (8 points) This question tests your ability to debug code with scanf, if-else statements, and floating point arithmetic. The following program should scan in two numbers and compute their sum, quotient, and remainder based on the choice menu provided. You can assume the user inputs valid input. You must find all bugs (hint: there are 4!).

```
1
2  #include <stdio.h>
3
4  int main(void) {
5      int num1, num2, choice;
6      char userinput;
7
8      //Read in user input
9      printf("Enter two integers:");
10     scanf("%d", &num1, &num2);
11     printf("You entered %d and %d. Would you like to compute the:\
12 ----\n\t1: sum\n\t2: quotient\n\t3:the remainder\n", num1, num2);
13     scanf("%d", &choice);
14
15     //Perform computations
16     if (choice = 1){
17         printf("The sum is %d", num1 + num2);
18     } else if (choice == 2) {
19         printf("The quotient is %f", num1/num2);
20     } else if (choice == 3){
21         printf("The remainder is %f", num1 % num2);
22     }
23     return 0;
24 }
```

8. (12 points) (BONUS) This question tests your understanding of C variables and operations. Your goal is track the variable known as POLANDIUM as it progresses through the code. Note the changes to its value, and write down what each of the printf statements will print out. Don't lose it, keep watch of it!

```
1  #include <stdio.h>
2
3  int earth(int a);
4  int siberia(int b);
5  void climate(int c);
6
7  int main() {
8      int POLANDIUM = 101;
9      printf("POLANDIUM = %d\n", POLANDIUM);
10
11     POLANDIUM = earth(POLANDIUM);
12     printf("POLANDIUM = %d\n", POLANDIUM);
13
14     POLANDIUM += 10 * 1;
15     printf("POLANDIUM = %d\n", POLANDIUM);
16
17     climate(POLANDIUM);
18     printf("POLANDIUM = %d\n", POLANDIUM);
19
20     POLANDIUM += ((50 & 63) / 1) + (48 & 7) - 0;
21     printf("POLANDIUM = %d\n", POLANDIUM);
22
23     POLANDIUM = siberia(POLANDIUM);
24     printf("POLANDIUM = %d\n", POLANDIUM);
25
26     return 0;
27 }
28
29 int earth(int a) {
30     int landmass;
31     int sea = 0;
32
33     for (landmass = 0; landmass < 5; landmass++) {
34         sea += landmass * 2;
35     }
36
37     if (sea > 10) {
38         return a + 39;
39     } else {
40         return a + 29;
41     }
42 }
```

```
43
44 void climate(int c) {
45     int system = 1;
46     int elevation = c + 6;
47
48     do {
49         system += elevation + system;
50         elevation = system % 35;
51     } while (system % elevation != 5 && elevation != 10);
52 }
53
54 int siberia(int b) {
55     int freeze;
56     int wind = 0;
57     for (freeze = 2; freeze > 0; freeze--) {
58         wind += (freeze * 3);
59     }
60     if (wind >= 6) {
61         b = 205;
62         return b;
63     } else {
64         b = 222;
65         return b;
66     }
67 }
```