1.) Here's the main function of a program that

- Reads in two matrices by prompting the user for their dimensions, dynamically allocating memory from the heap, and reading the values into the memory using row major ordering for the matrix values.
- Multiplies the two matrices together and puts the result into another dynamically allocated piece of memory (after checking that the dimensions are appropriate for matric multiplication).
- Outputs the two input and the result matrices.

Write the implementations of the functions `input_matrix`, `matrix_multiply`, and `output_matrix` (3.0%).

```
/*--------------------------------------------------------------------------
*/
int main(void) {

    double *m1,*m2,*m3;
    int m1_rows,m1_columns,m2_rows,m2_columns;

    if (((m1 = input_matrix(&m1_rows,&m1_columns,"Matrix 1")) != NULL) &&
((m2 = input_matrix(&m2_rows,&m2_columns,"Matrix 2")) != NULL) &&
((m3 = malloc(m1_rows*m2_columns*sizeof(double))) != NULL)) {
        printf("Matrix 1\n");
        output_matrix(m1,m1_rows,m1_columns);
        printf("Matrix 2\n");
        output_matrix(m2,m2_rows,m2_columns);
        if (matrix_multiply(m1,m1_rows,m1_columns,m2,m2_rows,m2_columns,m3))
{
            printf("Product\n");
            output_matrix(m3,m1_rows,m2_columns);
            free(m1);
            free(m2);
            free(m3);
            return(0);
        } else {
            printf("Error in dimensions\n");
            free(m1);
            free(m2);
            free(m3);
            return(-1);
        }
    } else {
        free(m1);
        free(m2);
        free(m3);
        printf("Error allocating memory\n");
        return(-2);
    }
}
```

2.) A naturalist is off to explore the amazon jungle, and needs a computer program to record information about all the new species discovered. For each new species it is necessary to store the name (max 128 characters), size (a real number), and the type of animal. mammal, insect, bird, or fish).

Here is what a sample run should look like (with the keyboard input shown in italics) ...

```
> NewSpecies
Enter animal information ("exit" to exit)
What is the name : bloatfish
What is the size : 12.47
What is the type : fish
Enter animal information ("exit" to exit)
What is the name : stingybeasty
What is the size : 0.13
What is the type : insect
Enter animal information ("exit" to exit)
What is the name : toothfulsloth
What is the size : 33.33
What is the type : mammal
Enter animal information ("exit" to exit)
What is the name : exit

The following new species were found:
bloatfish            has size  12.47 and is a fish
stingybeasty         has size   0.13 and is a insect
toothfulsloth        has size  33.33 and is a mammal
```

Implement the program in C. (3.0%)

- An array of structures must be used, so that each new species can be recorded in an element of the array.
- The type of animal is represented as an `enum` type, indicating one of mammal, insect, bird, or fish.
- It's not known in advance how many new species will be found, so the program must `malloc` for an initial array of size 1, and use the doubling `realloc` technique (as discussed in class) to get more memory as required. You must always check the return value from `malloc`, as done in the `Malloc` wrapper function (or just use `Malloc` :-).

Remarks:

- The assignment is due next Friday, Apr. 16th, 11:59PM.
- You need to submit two (`matrix.c, naturalist.c`) file with the corresponding C code using submit2 (Please DO NOT submit compiled executables)! Furthermore, the C code needs to compile ... (otherwise the TA's won't look at it)