Lab #4

Summer 2024

Requirements

In this lab, you will cover void pointers and structs. You will **not** be required to read in from a file for this lab. You can initialize your struct array in main however you like. Remember that it is considered good programming practice to separate a larger problem into multiple smaller problems. Creating smaller functions which solve a single problem that you can re-use in other functions is an important part of algorithm development.

In this lab, you will be working with the following struct:

```
typedef struct {
        int ID;
        float weight;
        int stockCount;
        short colors;
}   InventoryItem;
```

1.1 makeArray

```
void * makeArray(int arraySize, int elementSize)
```

Info: This function will take an array size, as well as the size of each element in the array. It will allocate an array with the given size, and store the getSize before the start of the array as an int. If allocating the array was successful, it will return a pointer to the array, otherwise it will return NULL.

1.2 getSize

```
int getSize(void *array)
```

Info: This function takes an array which was allocated with makeArray, and returns the size stored before the array.

1.3 countWithColors

```
int countWithColors(InventoryItem *items, short colorCount)
```

Info: This function takes an array of InventoryItems, and returns the number of items in the array which have a value of "colors" equal to the given colorCount. In this function, you **must not** access the size stored before the array directly. You must call **getSize** to get the size of the array instead.

1.4 freeArray

```
void freeArray(void *array)
```

Info: This function takes an array which was allocated with makeArray, and frees the memory allocated to the array.

Submission Information

Submit this assignment by using the mucsmake command.

Use the following command on Hellbender:

mucsmake <course> <assignment> <filename>

For example:

mucsmake 2050 lab4 lab4.c

Rubric: 11 points

1. Write required *makeArray* function
   * 4 points
2. Write required *getSize* function
   * 2 points
3. Write required *countWithColors* function
   * 3 points
4. Write required *freeArray* function
   * 2 points

Notice:

1. All of your lab submissions **must** include documentation in the form of code comments to receive full points. In addition, your program is expected to have a **comment header** at the top that includes your name, pawprint, the course you are taking, and the lab that you solved. You can refer to the Lab 0 document for an example of the comment header.
2. All of your lab submissions must compile under GCC using the -Wall and -Werror flags to be considered for a grade. These flags will automatically be applied if you use the compile command.
3. Do **NOT** change the given function prototype or anything else in the provided .h file.