PROGRAM 4 / Monster Zoo

CSC 1300 / Spring 2023

# Important Dates

Assignment Date: Wednesday, April 5, 2023

Due Date: Wednesday, April 19, 2023 by 11:59pm

**May be turned in late up through Monday April 24, 2023 11:59pm at a grade reduction of 10 points off per day late.**

# Submission

You should have a **zip file** named **username\_prog4**. This zip file should contain the files listed below. Replace **username** with **your TTU username**.

* **username\_prog4.cpp** – main function
* **username\_functions.cpp** – programmer-defined functions
* **username\_prog4.h** – header file (global constants, structure declarations, function prototypes)
* **monsters.txt** – this should be a text file with at least **FIVE** monsters that **you created** with your program **(data should be separated by #)**. Do not just submit my example text file that I provided you!

# Description

Your assignment is to write a program for a monster zookeeper. The zoo can hold up to **75 monsters**. Information about each monster will be kept and manipulated in this program. The user should be able to load existing monster information from any file he or she chooses, register a new monster, remove a monster from the zoo, print monster information to the screen, or print a cost analysis of each monster and the total cost to house and take care of these monsters.

# PROGRAM SPECIFICATIONS

## Structures

You will need to create two structures. One is called **Cost**. Cost will have the following members:

* The number of hours it takes to take care of a specific monster per week.
* The cost (per week) of taking care of this monster.
* The cost of food to feed this monster for one week.
* The cost of materials/supplies (grooming, medical) for this monster for one week

The second structure is called **Monsters**. Monsters will have the following members:

* **Name** of the monster
* **Description** of the monster
* **Weight** of the monster (in pounds)
* **Height** of the monster (in feet)
* **Location** of where the monster was from before being added to the zoo
* **Danger** level – this is an integer Likert Scale from 1 to 5 where 1 is not dangerous at all (you can pet it) to 5 being extremely dangerous (it will eat you)
* **Cost** structure variable (Cost structure is nested in Monsters structure)

# The Main Function

You will need to create an array of **75** elements of the **Monster** data type.

You will also need to create an integer that will hold the **current number** of monsters housed in the zoo. This number begins the program at zero.

Once you create the variables, print a welcome message to tell the user what the program is and then call the **loadMonstersFromFile()** function, which will get all the monsters currently at the zoo into the program.

Start a loop and display a menu of five options (below).

**What would you like to do?**

**1. Register a New Monster**

**2. Remove a Monster From Zoo**

**3. Print Monsters to Screen**

**4. Print Monster Care Cost Data**

**5. End Program**

**Enter 1, 2, 3, 4, or 5.**

**CHOICE:**

Validate the user’s choice.

Each menu choice will call a function as specified below.

If the user chooses **option 1**, then your program will call the **registerMonster()** function.

If the user chooses **option 2**, then your program will call the **removeMonster()** function.

If the user chooses **option 3**, then your program will call the **printMonsters()** function.

If the user chooses **option 4**, then your program will call the **printCostInfo()** function.

If the user chooses **option 5**, then your program will ask the user if they wish to save the current zoo monster data to a file. If they choose yes, then your program should call the **saveMonstersToFile()** function and then end. If they choose no, then your program should just end.

# Programmer-Defined Functions

## loadMonstersFromFile

The **loadMonstersFromFile()** function takes two parameters: the number of monsters currently loaded in the Monsters array and the Monsters array. The function will return the number of monsters at the zoo after adding all of them from the file. Start the current number of monsters to zero before adding from the file.

Ask the user what the name of the file is that they would like to load the monsters from. Then open their file. Check if the file could open before reading from it.

As long as there is data from the file and we haven’t reached the zoo maximum capacity of 75, read each monster from the file and place the data in the correct element in the Monsters array, making sure that you print the name of the monster added from the file and increment the number of monsters each time a monster is added.

When you are reading from the file, everything read in will have to be read in as a string because the data is all separated by a delimiter, ‘#’ instead of by newlines. Some of the Monster members are strings, so that won’t be a problem. However, some of the Monster members are doubles or integers. So, you will have to first read the data into a temporary string and then convert the string to either a double using the **stod** function or an integer using the **stoi** function. (<https://www.geeksforgeeks.org/stdstod-stdstof-stdstold-c/>)

After adding the monster data from the file, return the updated number of monsters loaded into the array.

Example of this part of the program running below with user input highlighted in yellow.

**What is the name of the file with your monster data? (ex: filename.txt)**

**FILENAME: monsterFile\_4-3-2023.txt**

**Zombiedog has been added.**

**Chucky Chupacabra has been added.**

**Celfy has been added.**

**Fliggerwit Fletcher has been added.**

**Orgodore Borgmore has been added.**

**All creatures from monsterFile\_4-3-2023.txt have been added to the program.**

## registerMonster

The **registerMonster()** function takes two parameters: the number of monsters currently loaded in the Monsters array and the Monsters array. The function will return the updated number of monsters.

Before trying to add any monsters, this function should first check to make sure the number of monsters isn’t already 75. Because if it is, then your program should not add any monsters, but should instead tell the user that their zoo is at full capacity and that they are not able to add monsters. Then the function should end.

Otherwise, your program will ask the user for the following: (make sure you place each bit of information in the correct place in the Monsters array)

* NAME:
* DESCRIPTION:
* WEIGHT (in pounds): **[validate that weight is over zero pounds]**
* HEIGHT (in feet): **[validate that height is taller than zero feet]**
* LAST KNOWN LOCATION:
* DANGER LEVEL (1-5): **[validate that level is 1 through 5]**
* CARE INFORMATION (per week): **[validate that they didn’t enter negative number for all four of these]**
  + Required direct care for monster (in hours)
  + Cost of care $
  + Food cost $
  + Medical & grooming cost $

Then, increment the number of current monsters in the zoo by one.

Last, return the number of monsters.

Example of this part of the program running below with user input highlighted in yellow.

**NAME: Mike Wazowski**

**DESCRIPTION: An actor who played one of the main characters in Monsters, Inc. and other Monsters moviews. His full name is Michael Zazowski. Started acting when he was only 18 years old. Green with one eye and sharp teeth. Eats everything humans eat but hates all green foods.**

**WEIGHT (in pounds): 25.6**

**HEIGHT (in feet): 2.6**

**LAST KNOWN LOCATION: Cookeville, TN**

**DANGER LEVEL (1-5): 1**

**CARE INFORMATION (per week):**

**Required direct care for monster (in hours) 3.45**

**Cost of care $ 33.3**

**Food cost $ 95.2**

**Medical & grooming cost $ 856.76**

**The Mike Wazowski has been added.**

## removeMonster function

The **removeMonster()** function has two parameters: the current number of monsters in the Monsters array and the Monsters array. This function returns the new number of monsters.

First, this function will say “The following is a list of all the monsters in the zoo: “ and then it will print the **name** of each monster. Then, your program will ask the user  
What monster is leaving the zoo?  
MONSTER NAME:

Your program should then read in the name and place it in a temporary string variable.

Use a loop to find the index of the monster that needs to be removed (by trying to find a name match). Once that is found, then you know there is a monster in the Monsters array by that name and that your program will be able to remove it.

If you found the monster in the array, use a loop to overwrite the element with the monster to delete (x) with the next element in the array (x+1), moving each element after the deleted element to the left. Then, decrement the number of monsters in the zoo and print out “You have removed [insert monster name here].”

If you did not find the monster in the array, then print “Sorry, a monster by the name [monster name user entered in] could not be found.”

Last, return the new number of monsters.

Example of this part of the program running below with user input highlighted in yellow.

**The following is a list of all the monsters in the zoo:**

**Zombiedog**

**Chucky Chupacabra**

**Celfy**

**Fliggerwit Fletcher**

**Orgodore Borgmore**

**Mike Wazowski**

**What monster is leaving the zoo?**

**MONSTER NAME: dk**

**Sorry, a monster by the name dk could not be found.**

Another example of this part of the program running below with user input highlighted in yellow.

**The following is a list of all the monsters in the zoo:**

**Zombiedog**

**Chucky Chupacabra**

**Celfy**

**Fliggerwit Fletcher**

**Orgodore Borgmore**

**Mike Wazowski**

**What monster is leaving the zoo?**

**MONSTER NAME: Chucky Chupacabra**

**You have removed Chucky Chupacabra.**

An easy way to test if your function worked is to select either of the print functions next to see if this monster is still listed.

## printMonsters function

The **printMonsters()** function is a void function and contains the following parameters: number of monsters currently in the Monsters array and the Monsters array.

If there are no monsters in the zoo yet, then print “THERE ARE NO MONSTERS AT YOUR ZOO!”. Otherwise, print each monster in a very easy-to read format to the screen. Refer to the sample output below.

Notice that the descriptions in my example are word wrapped so that no words are cut in the middle and also the paragraph is indented. You **get a bonus of 5 points if you can do word wrapping like this for your descriptions**.

Example of this part of the program running below. There is no user input in this section.

**------------------------------------------------------------------------**

**\*\*MONSTER 1\*\***

**Name: Zombiedog**

**Description:**

**Undead dog who died a while back and is now a zombie. Brown.**

**Loves to eat popcorn.**

**Weight: 30.00 pound(s)**

**Height: 2.00 feet**

**Last known location: Unknown**

**Danger level: 1**

**Weekly Care Information:**

**-Hours of care required: 3.00**

**-Cost of care: $ 3.99**

**-Food cost: $ 98.30**

**-Grooming & Supplies Cost: $ 29.30**

**------------------------------------------------------------------------**

**\*\*MONSTER 2\*\***

**Name: Chucky Chupacabra**

**Description:**

**Goat sucker that attacks animals and consumes their blood. Has**

**a bat-looking head, a hound body, alagator feet, pig tail,**

**and large eyes. Grey in color.**

**Weight: 56.50 pound(s)**

**Height: 3.50 feet**

**Last known location: Puerto Rico**

**Danger level: 5**

**Weekly Care Information:**

**-Hours of care required: 0.50**

**-Cost of care: $ 84.40**

**-Food cost: $ 22.90**

**-Grooming & Supplies Cost: $ 3.10**

**------------------------------------------------------------------------**

**\*\*MONSTER 3\*\***

**Name: Celfy**

**Description:**

**Holstein Friesian cow with black and white color who grew up**

**in the North Pole and thought he was an elf. Learned how to**

**create fidget spinners, which made Santa really rich. Celfy**

**is now an indispensable work elf.**

**Weight: 560.30 pound(s)**

**Height: 4.50 feet**

**Last known location: North Pole**

**Danger level: 1**

**Weekly Care Information:**

**-Hours of care required: 3.00**

**-Cost of care: $ 93.10**

**-Food cost: $ 998.50**

**-Grooming & Supplies Cost: $ 4.60**

**------------------------------------------------------------------------**

**\*\*MONSTER 4\*\***

**Name: Fliggerwit Fletcher**

**Description:**

**This monster's description must remain private due to sensitive**

**government information. You must have security level delta-4**

**to see the description or last known location.**

**Weight: 958.00 pound(s)**

**Height: 7.80 feet**

**Last known location: removed**

**Danger level: 5**

**Weekly Care Information:**

**-Hours of care required: 384.30**

**-Cost of care: $ 3948.20**

**-Food cost: $ 33.30**

**-Grooming & Supplies Cost: $ 1.20**

**------------------------------------------------------------------------**

**\*\*MONSTER 5\*\***

**Name: Orgodore Borgmore**

**Description:**

**Purple creature with two arms and three legs. One eye and one**

**large horn protruding from skull. Brain is larger than normal.**

**Dark red eyes that change when noticed to green. Loves**

**to eat Jack Russel Terriers - especially fat ones that are**

**mostly white.**

**Weight: 2000.00 pound(s)**

**Height: 7.20 feet**

**Last known location: Mos Eisley Cantina**

**Danger level: 2**

**Weekly Care Information:**

**-Hours of care required: 10.00**

**-Cost of care: $ 3.99**

**-Food cost: $ 1.99**

**-Grooming & Supplies Cost: $ 394.30**

**------------------------------------------------------------------------**

**\*\*MONSTER 6\*\***

**Name: Mike Wazowski**

**Description:**

**An actor who played one of the main characters in Monsters, Inc.**

**and other Monsters movies. His full name is Michael Wazowski.**

**Started acting when he was only 18 years old. Green**

**with one eye and sharp teeth. Eats everything humans eat**

**but hates all green foods.**

**Weight: 25.60 pound(s)**

**Height: 2.60 feet**

**Last known location: Cookeville, TN**

**Danger level: 1**

**Weekly Care Information:**

**-Hours of care required: 3.45**

**-Cost of care: $ 33.30**

**-Food cost: $ 95.20**

**-Grooming & Supplies Cost: $ 856.76**

## printCostInfo function

The **printCostInfo()** function is a void function and it contains two parameters: the current number of monsters in the Monster array and the Monsters array.

If the number of current monsters in the zoo is zero, then the function should just print “There are no monsters in the zoo so the total cost is $0.00.” and then the function should end.

Otherwise, this function should print out the total cost to care for each monster in a table.

To figure out the cost of a monster, use the formula:

care cost = numHours \* costPerHour + foodCost + materialCost

Then, this function should also keep a running total of the total cost of all the monsters and print the total at the end.

Example of this part of the program running below. There is no user input in this part of the program.

**COST OF EACH MONSTER FOR ONE WEEK:**

**MONSTER CARE COST**

**Zombiedog $ 139.57**

**Chucky Chupacabra $ 68.20**

**Celfy $ 1282.40**

**Fliggerwit Fletcher $ 1517327.75**

**Orgodore Borgmore $ 436.19**

**Mike Wazowski $ 1066.84**

**TOTAL COST: $ 1520321.00**

## saveMonstersToFile function

The **saveMonstersToFile()** function is a void function and it contains two parameters: the current number of monsters in the Monsters array and the Monsters array.

If the number of monsters currently in the zoo is zero, then this function should print to the screen “There are no monsters in the zoo so none can be saved to a file.” and then the function should end.

Ohterwise, the function should ask the user what the name of the file that they wish to save their monsters to. The function should then open that file and write out all the monster data in the following order:

* Name
* Description
* Weight
* Height
* Last known location
* Danger level
* Number of hours to care for monster
* Cost to care for monster
* Food cost
* Material cost

There should be no newlines or endlines in the file. **All data should be separated by a pound sign (#) instead of a space.**

After printing all data from the Monsters array to the file, this function should print out a message to standard output saying “All monsters currently housed in the zoo were successfully saved to the [insert filename here] file.”

CHOICE: 5

Would you like to save your monster list to a file? (y or n) y

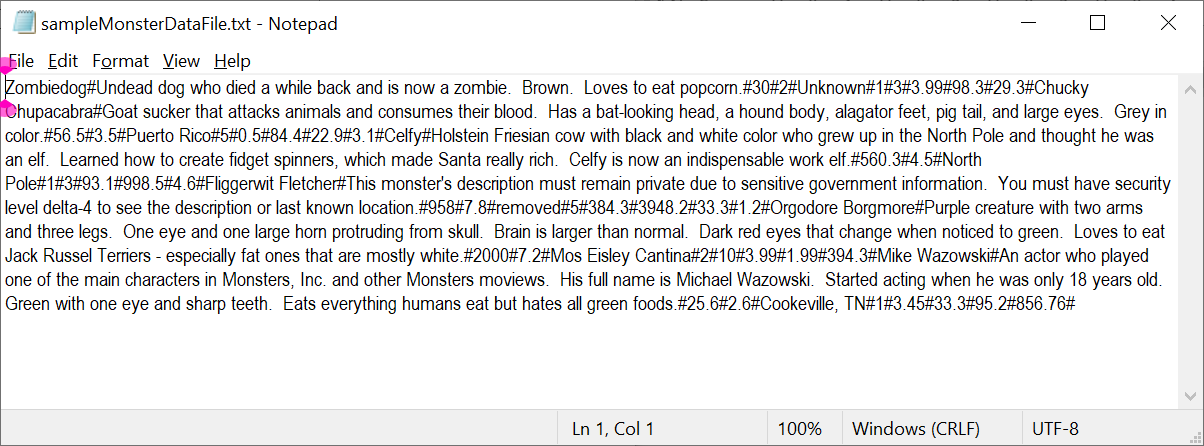
What is the name of the file you want to save your creatures to?

FILENAME: example\_4-3-2023.txt

Your monsters currently housed in the zoo were successfully saved to the example\_4-3-2023.txt file.

# SAMPLE OUTPUT

Here is the sampleMonsterDataFile.txt:



Example run of program below. User input highlighted in yellow.

**WELCOME TO THE MONSTER ZOO PROGRAM!!**

**What is the name of the file with your monster data? (ex: filename.txt)**

**FILENAME: sampleMonsterDataFile.txt**

**Zombiedog has been added.**

**Chucky Chupacabra has been added.**

**Celfy has been added.**

**Fliggerwit Fletcher has been added.**

**Orgodore Borgmore has been added.**

**Mike Wazowski has been added.**

**All creatures from sampleMonsterDataFile.txt have been added to the program.**

**What would you like to do?**

**1. Register a New Monster**

**2. Remove a Monster From Zoo**

**3. Print Monsters to Screen**

**4. Print Monster Care Cost Data**

**5. End Program**

**Enter 1, 2, 3, 4, or 5.**

**CHOICE: 1**

**NAME: Bigol Nerd**

**DESCRIPTION: Was made fun of their whole life because they loved to program computers and learn about ethical hacking. Wears beautiful flowy pants with button-up, hooded shirts. Has extremely long brilliant red hair. Four eyes - one pair in front of skull and the other in back. Loves to eat orange sherbert and double roasted pumpkin seeds. Sometimes can be found hiding in a large hole in the ground.**

**WEIGHT (in pounds): 164**

**HEIGHT (in feet): 6.2**

**LAST KNOWN LOCATION: Beckley, West Virginia**

**DANGER LEVEL (1-5): 3**

**CARE INFORMATION (per week):**

**Required direct care for monster (in hours) 2.3**

**Cost of care $ 39.4**

**Food cost $ 9.2**

**Medical & grooming cost $ 384.88**

**The Bigol Nerd has been added.**

**What would you like to do?**

**1. Register a New Monster**

**2. Remove a Monster From Zoo**

**3. Print Monsters to Screen**

**4. Print Monster Care Cost Data**

**5. End Program**

**Enter 1, 2, 3, 4, or 5.**

**CHOICE: 2**

**The following is a list of all the monsters in the zoo:**

**Zombiedog**

**Chucky Chupacabra**

**Celfy**

**Fliggerwit Fletcher**

**Orgodore Borgmore**

**Mike Wazowski**

**Bigol Nerd**

**What monster is leaving the zoo?**

**MONSTER NAME: Mike Wazowski**

**You have removed Mike Wazowski.**

**What would you like to do?**

**1. Register a New Monster**

**2. Remove a Monster From Zoo**

**3. Print Monsters to Screen**

**4. Print Monster Care Cost Data**

**5. End Program**

**Enter 1, 2, 3, 4, or 5.**

**CHOICE: 3**

**------------------------------------------------------------------------**

**\*\*MONSTER 1\*\***

**Name: Zombiedog**

**Description:**

**Undead dog who died a while back and is now a zombie. Brown.**

**Loves to eat popcorn.**

**Weight: 30.00 pound(s)**

**Height: 2.00 feet**

**Last known location: Unknown**

**Danger level: 1**

**Weekly Care Information:**

**-Hours of care required: 3.00**

**-Cost of care: $ 3.99**

**-Food cost: $ 98.30**

**-Grooming & Supplies Cost: $ 29.30**

**------------------------------------------------------------------------**

**\*\*MONSTER 2\*\***

**Name: Chucky Chupacabra**

**Description:**

**Goat sucker that attacks animals and consumes their blood. Has**

**a bat-looking head, a hound body, alagator feet, pig tail,**

**and large eyes. Grey in color.**

**Weight: 56.50 pound(s)**

**Height: 3.50 feet**

**Last known location: Puerto Rico**

**Danger level: 5**

**Weekly Care Information:**

**-Hours of care required: 0.50**

**-Cost of care: $ 84.40**

**-Food cost: $ 22.90**

**-Grooming & Supplies Cost: $ 3.10**

**------------------------------------------------------------------------**

**\*\*MONSTER 3\*\***

**Name: Celfy**

**Description:**

**Holstein Friesian cow with black and white color who grew up**

**in the North Pole and thought he was an elf. Learned how to**

**create fidget spinners, which made Santa really rich. Celfy**

**is now an indispensable work elf.**

**Weight: 560.30 pound(s)**

**Height: 4.50 feet**

**Last known location: North Pole**

**Danger level: 1**

**Weekly Care Information:**

**-Hours of care required: 3.00**

**-Cost of care: $ 93.10**

**-Food cost: $ 998.50**

**-Grooming & Supplies Cost: $ 4.60**

**------------------------------------------------------------------------**

**\*\*MONSTER 4\*\***

**Name: Fliggerwit Fletcher**

**Description:**

**This monster's description must remain private due to sensitive**

**government information. You must have security level delta-4**

**to see the description or last known location.**

**Weight: 958.00 pound(s)**

**Height: 7.80 feet**

**Last known location: removed**

**Danger level: 5**

**Weekly Care Information:**

**-Hours of care required: 384.30**

**-Cost of care: $ 3948.20**

**-Food cost: $ 33.30**

**-Grooming & Supplies Cost: $ 1.20**

**------------------------------------------------------------------------**

**\*\*MONSTER 5\*\***

**Name: Orgodore Borgmore**

**Description:**

**Purple creature with two arms and three legs. One eye and one**

**large horn protruding from skull. Brain is larger than normal.**

**Dark red eyes that change when noticed to green. Loves**

**to eat Jack Russel Terriers - especially fat ones that are**

**mostly white.**

**Weight: 2000.00 pound(s)**

**Height: 7.20 feet**

**Last known location: Mos Eisley Cantina**

**Danger level: 2**

**Weekly Care Information:**

**-Hours of care required: 10.00**

**-Cost of care: $ 3.99**

**-Food cost: $ 1.99**

**-Grooming & Supplies Cost: $ 394.30**

**------------------------------------------------------------------------**

**\*\*MONSTER 6\*\***

**Name: Bigol Nerd**

**Description:**

**Was made fun of their whole life because they loved to program**

**computers and learn about ethical hacking. Wears beautiful**

**flowy pants with button-up, hooded shirts. Has extremely**

**long brilliant red hair. Four eyes - one pair in front of skull**

**and the other in back. Loves to eat orange sherbert and**

**double roasted pumpkin seeds. Sometimes can be found hiding**

**in a large hole in the ground.**

**Weight: 164.00 pound(s)**

**Height: 6.20 feet**

**Last known location: Beckley, West Virginia**

**Danger level: 3**

**Weekly Care Information:**

**-Hours of care required: 2.30**

**-Cost of care: $ 39.40**

**-Food cost: $ 9.20**

**-Grooming & Supplies Cost: $ 384.88**

**What would you like to do?**

**1. Register a New Monster**

**2. Remove a Monster From Zoo**

**3. Print Monsters to Screen**

**4. Print Monster Care Cost Data**

**5. End Program**

**Enter 1, 2, 3, 4, or 5.**

**CHOICE: 4**

**COST OF EACH MONSTER FOR ONE WEEK:**

**MONSTER CARE COST**

**Zombiedog $ 139.57**

**Chucky Chupacabra $ 68.20**

**Celfy $ 1282.40**

**Fliggerwit Fletcher $ 1517327.75**

**Orgodore Borgmore $ 436.19**

**Bigol Nerd $ 484.70**

**TOTAL COST: $ 1519738.88**

**What would you like to do?**

**1. Register a New Monster**

**2. Remove a Monster From Zoo**

**3. Print Monsters to Screen**

**4. Print Monster Care Cost Data**

**5. End Program**

**Enter 1, 2, 3, 4, or 5.**

**CHOICE: 5**

**Would you like to save your monster list to a file? (y or n) y**

**What is the name of the file you want to save your creatures to?**

**FILENAME: anotherFile\_4-3-2023.txt**

**Your monsters currently housed in the zoo were successfully saved to the anotherFile\_4-3-2023.txt file.**

**GOODBYE!**

