
Program name : Project2_Design_Reflection_Schaefer_Kristin.pdf
Author : Kristin Schaefer
Date : 04-29-2019

Design Reflection for Project2

Changes in design

For Project2, my final program had few changes in relation to the original design plan. The majority of changes involved adding extra functions in order to avoid having functions such as `playGame()` become too long. For example, I broke apart the `randomEvent()` function into several functions to make the function more legible and easier to manage.

Regarding the use of Inheritance in this project, I found that I made very few changes in the derived Tiger, Turtle and Penguin classes thanks to the previous lab assignment. I had anticipated needing to use virtual functions in the design plan, however I found that setting all of the different variables in each Animal's constructor was sufficient.

For the `randomEvent()` function, the previous lab assignment was also a great help, as I could reuse the structure of my previous assignment with few alterations or errors.

The trickiest part where I had to do some shuffling, was with the use of the dynamic arrays. The order of copying one array to a temp array and then deleting the old array memory was quite tricky.

Problems encountered (in testing) and how problems were solved

For this project I had a much easier time debugging my program as I used a piecemeal approach this time. First I created the Animal class and its derived classes and verified everything was working through the main function. Then I created the Zoo class and started to check that the constructor of the Zoo object was working. Then I added menu functions and checked the loops of the game in the main function. After I took several steps to implement the `playGame()` and `randomEvent()` functions, as these were the most challenging functions to create.

For design testing I had no problems this time as I moved my input validation into two different functions `validate1or2()` and `validate1or2or3()`. This was a great time saver!

The biggest problem that I encountered in this assignment was not dealing with Inheritance, but with dynamic memory allocation. I had segmentation faults in my program at times, and I managed to fix this problem by placing my seed values for the random functions inside the functions themselves and not in the main function. After running `valgrind` several times and making adjustments, I had an error "Conditional jump or move depends on uninitialized value(s)" that I was unfortunately unable to resolve. I traced through the sequence of allocation, but I was not able to find the error. Unfortunately I was out of time to discuss the problem on Piazza or with an instructor. To understand the issue going forward, I will ask an instructor if they can help me with the error.