|  |  |  |
| --- | --- | --- |
| **Date** | **Time** | **Task** |
| 3/26/2020 | 2.5 hours | Design document and program refactoring |
| 3/27/2020 | .5 | Added GetFileName() & implemented it |
| 3/27/2020 | .5 | Updated structure chart to include GetFileName() |
| 4/3/2020 | 1.75 | Added breakpoint functions |
| 4/3/2020 | .5 | Added cake function, verified calculations |
| 4/3/2020 | .5 | Revision work, working on calculating and outputting number of full and half sheets correctly |
| 4/4/2020 | .5 | Fixed cake calculations, removed cake function |
| 4/4/2020 | .5 | Design revisions to make work done on program |
| 4/8/2020 | .5 | Final documentation and running the report |
|  |  |  |
|  |  |  |
| Total | 7.75 hrs |  |

Summary of program experience:

Having the convenience of a preexisting design document to work from made modifying this program a fairly simple task. Modifying the costs of each item was as easy as changing the constants. Adding a function to have the user enter the cost of the cake was simple because of the existing functions to assign input to variables.

The revisions were challenging. There was some refactoring, for instance I changed the cost per pack cost constants to variables. There was also some iteration, as I added functions with selection to calculate the proper cost per pack for a given item. I kept the math to figure out how many packs were needed in the main function but passed that information to new breakpoint functions to determine the item’s final price.

I then removed the cake input function by commenting out the call, declaration and definition. Calculating the proper amount of full sheet cakes was simply a matter of using modulo division with the number of children mod 50. A quick check on whether there were remaining guests determined whether there was more cake to buy and checking how many were left (more or less than 25) helped determine if that addition cake was a full or half sheet.

In all, Program 4 was a welcome challenge and demonstrated well the maintenance phase of the program development cycle. I look forward to being able to begin a new program at a more progressive step in the design process, having now experienced what it means to iterate through an existing algorithm.