**CSI-120 Fall 2022 Final Exam**

**(Extra Credit) Handle All User Input Exceptions using While Loop and TryParse (Extra Credit)**

1. Declare variables for each of the following values. Assign the values to them, make sure to choose the data type that is the **best fit** for the value. Omit any commas, they are just there to make the number easier to read.
   1. ‘Z’
   2. 3.7
   3. 2 < 5
   4. $12.25
   5. 300,000
   6. -9,123,456,798,123
   7. 15.12345678910
2. A local dairy has asked you to calculate their daily shipping needs, profits, and costs. A shipping carton of milk holds 3.78 liters of milk. Each morning, a dairy farm ships cartons of milk to a local grocery store. The cost of producing a liter of milk is $0.38. The profit for each liter sold is $0.27. Write code which does the following:
   1. **Prompts the user** to enter the total amount of milk produced in the morning.
   2. **Outputs the number of milk cartons** needed to hold milk. (Round your answer to the nearest integer.)
   3. **Outputs the cost of producing milk.**
   4. **Outputs the profit for producing milk.**
3. Write a loop to add all the consecutive numbers between 5 and 45 except the 12, 25 and 40. Display the final sum.
4. Request two integer numbers from the user. Using nested for Loops Print out a grid of \* that has as many **ROWS** as the first number and as many **COLUMNS** as the second number.
5. The freezing point of alcohol is -173 degrees Fahrenheit. The boiling point is 172 degrees Fahrenheit. **Create a method** that takes in the **temperature** in Fahrenheit and **returns** either “below freezing”, “above boiling” or “in between”. In main Prompt the user to input a temperature and call your method with the temperature. Display the result from main.
6. Using the formula to convert to Celsius. C = (F - 32) \* 5/9, modify your method for #5 to **take two parameters**, the **temperature** and a **bool** which indicates whether the temperature was passed as Fahrenheit or Celsius. Return the result as normal after converting to the correct units if needed. Your solution should work with either a F or C temperature passed to it.
7. Create an **empty array of doubles** with size of **7** to store how many hours an employee worked for each day of the week. The **first index storing hours for Monday** and the **last index storing the hours for Sunday**. Prompt the user to input a value for each day filling the array with values. **Do not allow a number < 0 or > 24** to be placed in the array. When complete the array should look something like this (Your values do not need to match exactly).

Diagram

Description automatically generated with medium confidence

1. **Create a method** to **display** an employee’s hours worked. The method will take a **double array** as a parameter will display each element from the array as well as the day of the week starting with Monday. Call the method from main passing the array you made for #7.

Text

Description automatically generated

1. **Create a method** to return an employee’s total weekly paycheck. The method will take a **double array** and an **hourlyWage** as parameters. Employees of this company earn **1.5x their normal rate for any hours worked on Saturday** and **1.75x their normal rate for any hours worked on Sunday**. Call the method to the weekly paycheck for the double array you created in #7, choose a number for the **hourlyWage**, R**eturn** the amount to main, Display the results from main.
   1. Using the Example Schedule from number7 and hourly wage of $10 results in $457.50
2. **Create a method** that takes a int[] as a parameter and reverses the array. (Do not use any built-in methods for this, you must code it by hand, do not simply print the array backwards, this is not reversing it). Call the method and pass an int[]. Return the reversed array to main and display the result.
   1. For example, if the Original array is [5, 7, 8, 1, 4] the output should be 4 1 8 7 5
3. **Create a method** that takes a int[] as a parameter and a **searchValue**. Search through the array for the **searchValue** and **return the index** of the value if it exists in the array. If it does not exists return -1. Call the method from main and display the result.
4. **Create a method** that takes an int as a parameter. The int is going to be a number between 0 and 999 (You can assume that the value is in the correct range, you do not have to check for this). The method should covert the number to words, corresponding to the English pronunciation. For Example:
   1. 5: "Five"
   2. 13: "Thirteen"
   3. 87: "Eighty seven"
   4. 428: "Four hundred twenty eight"
   5. 600: "Six hundred"
   6. 201: "Two hundred and one"
   7. 311: "Three hundred and eleven"

Hint: if you are thinking of starting with:

If (number == 1)

{

Console.WriteLine(“One”);

}

You are approaching this problem incorrectly, how can you isolate the individual digits of the number??