ENGI 1331 – Exam 1 Review #2

Background: *Aluminum alloys are made by adding other elements to aluminum to improve its properties, such as hardness or tensile strength. Table 1 shows the composition of a set of commonly used alloys which are known by alloy number (2024, 6061, and so on) and the percent of that alloy made up by specific elements other than aluminum.*

Table 1. Aluminum Alloy Composition

Alloy	Composition of Aluminum Alloy (%)				
	Cu	Mg	Mn	Si	Zn
2024	4.4	1.5	0.6	0	0
6061	0	1	0	0.6	0
7005	0	1.4	0	0	4.5
7075	1.6	2.5	0.2	0	5.6
3560	0	0.3	0	7	0

Exam1_summer2020.mat *file provided has three variables that contains the information in the table above:*

- **Element:** contains the element name (string array)
- **Alloy**: contains the alloy number of each composition (vector)
- **Percent**: contain a different combination of the percent of each element for the alloy numbers (matrix)

TASK 1

Function: `function RESULT = Task1(Percent,SelE)`

- For the selected element, determine the average and median percent composition for each alloy and the range of values (minimum and maximum value).

Main script:

- Load the data **Exam1.mat** into your script.
- Create a menu to select a number from 1 to number of column in the matrix **Percent** and assign it to a variable named **SelE**.
- Use your function Task1 to get the resulting column vector
- Export the results as a column vector named **Task1.csv** in the following order: (1) minimum, (2) average, (3) median, and (4) maximum.

TASK 2

Function: `function [OverallMax,MAXalloy,MAXelement, PercentNorm] = Task2(Element,Alloy,Percent)`

- Determine the overall maximum percent value from **Percent** and associated **Alloy** and **Element**.
- Using the overall maximum value, create a normalized Percent matrix and store as **PercentNorm.**
  NOTE: To normalize the matrix, divide all values by the overall maximum and multiple by 100.
  This will make the overall maximum value equal to 100.

Main script:

- Create a formatted output showing the overall maximum information (see sample output).
- Export the **PercentNorm** variable as a .mat file named **Task2.mat.**

TASK 3

Function: `function [Max1,Element1,Max2,Element2] = Task3(SelA,Percent,Element)`

- For the selected alloy, determine the two highest percent compositions and associated elements from **Element**.

Main script:

- Create a menu to select an alloy number from **Alloy**.
- Create a formatted output for values and elements as shown in the sample output.

TASK 4

Function: `function [AlloyMax,maxNUM,ELEMENTless] = Task4(USERpercent,Element,Alloy,Percent)`

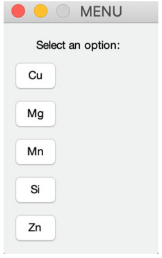
- Determine which alloy number has the most elements less than the user entered percent and greater than zero. For this alloy number, determine the number of elements that met the condition and identify the associated elements.
- Store a string array of only the element that met the condition and name as **ELEMENTless.**

Main script:

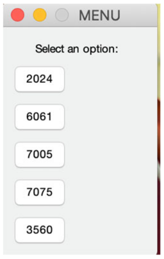
- Ask the user to enter a percent and store as **USERpercent**.
- Create a formatted output for the results above (see sample output).

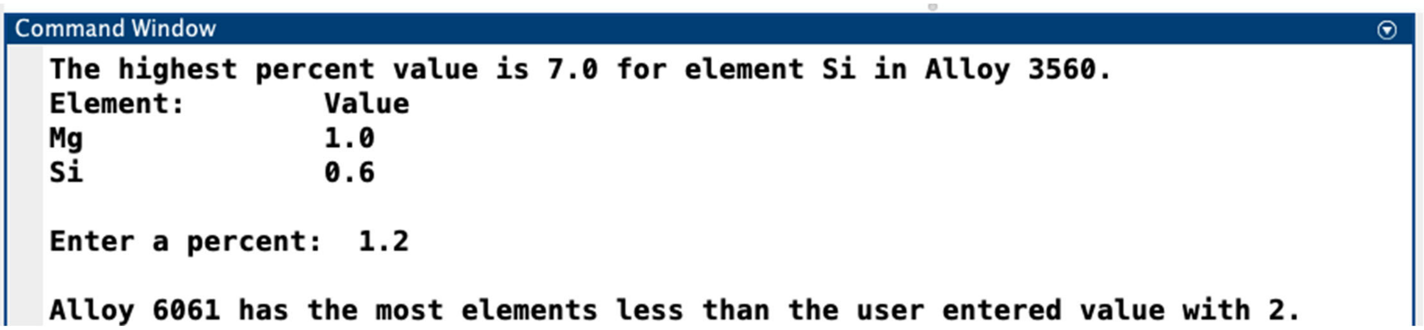
Sample Output

Task 1: Select Si



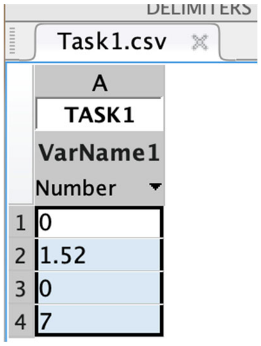
Task 3: Select 6061





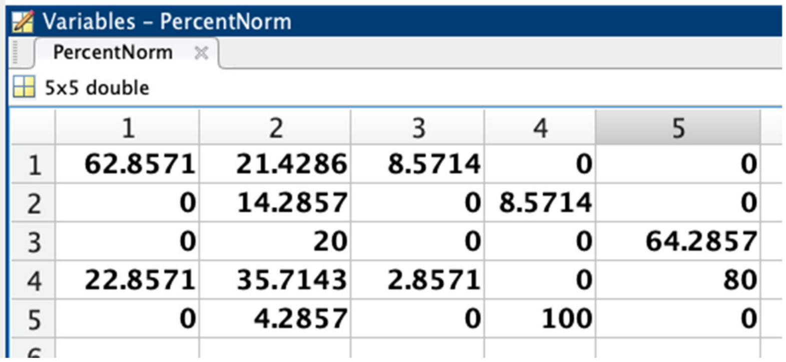
```
The highest percent value is 7.0 for element Si in Alloy 3560.
Element:        Value
Mg              1.0
Si              0.6

Enter a percent:  1.2

Alloy 6061 has the most elements less than the user entered value with 2.
```

Task1.csv (Task 1):



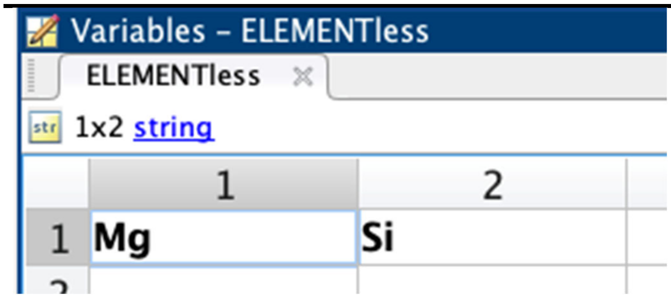
	A
	TASK1
	VarName1
	Number
1	0
2	1.52
3	0
4	7

PercentNorm (Task 2):



	1	2	3	4	5
1	62.8571	21.4286	8.5714	0	0
2	0	14.2857	0	8.5714	0
3	0	20	0	0	64.2857
4	22.8571	35.7143	2.8571	0	80
5	0	4.2857	0	100	0

ELEMENTless (Task 4): Percent entered = 1.2



	1	2
1	Mg	Si