

Date : 13/05/2020

Name Surname :

ID :

Education : 1st 2nd

Asst. Prof. Mehmet MERKEPÇİ

Asst. Prof. Tolga Ulaş GÜRBÜZ

Q1 (25 points)

Using Octave, write a function called "is_perfect" that determines whether the entered number is the perfect number or not. The function will return logical 1 if the input is a perfect number, and logical 0 otherwise.

```
function Out = is_perfect(Number)
sum = 0;
for i = 1:(Number-1)
    if rem(Number,i) == 0
        sum = sum + i;
    end
end
if sum == Number
    Out=true;
else
    Out=false;
end
```

Q2 (25 points)

Using Octave, write a function called "mean_median" that returns the mean and median of a given array of numbers.

```
function [mean, median] = mean_median(Numbers)
sum = 0;
for j=1:length(Numbers)
    sum = sum + Numbers(j);
end
mean = sum/length(Numbers);
for i=1:length(Numbers)
    temp = Numbers(i);
    for m=1:length(Numbers)
        if Numbers(m)>temp
            Numbers(i) = Numbers(m);
            Numbers(m) = temp;
            temp = Numbers(i);
        end
    end
end
if rem(length(Numbers),2) == 0
    median = (Numbers(length(Numbers)/2) +
Numbers((length(Numbers)/2)+1))/2;
else
    median = Numbers((length(Numbers)+1)/2);
end
end
```

Q3 (25 points)

Assume that at each row of a 2D character array, a name exists between two asterisk signs (*). At each row, along with other characters, there is only one name and there are only two asterisk signs. A name can be anywhere in the row.

Using Octave, write a function called "longest_name" that will accept an array with above-given properties as input and return the longest name in this array.

Example:

If Records = ['123*Ali*36' ; '*Veli*1783' ; '99*Zeynep*']
and Out = longest_name(Records),
Out will be 'Zeynep'.

```
function Out = longest_name(Records)

n_rows = size(Records,1);
n_columns = size(Records,2);
longest = 0;

for i = 1:n_rows
    found = 0;
    k=1;
    name = '';
    for j = 1:n_columns
        if Records(i,j)=='*' && found==0
            found = 1;
        elseif Records(i,j)~='*' && found==1
            name(k) = Records(i,j);
            k=k+1;
        elseif Records(i,j)=='*' && found==1
            break
        end
    end
    if (length(name)>length(longest))
        longest = name;
    end
end
Out = longest;
end
```

Q4 (25 points)

Consider a matrix of numbers which has positive elements.

Using Octave, write a function called "mean_of_multiples" whose first input argument is the matrix defined above and the second input argument is a positive integer. The function will return the mean of the elements that are the multiples of the second input argument. The output will be '0' if there are no multiples of the second input argument in the matrix.

Example:

If Numbers = [2 3 4; 6 15 1; 12 7 8]

and Out = mean_of_multiples (Numbers, 3)

Out will be equal to 9 since $(3 + 6 + 15 + 12) / 4 = 9$.

```
function Out = mean_of_multiples(Numbers, Num)

n_rows = size(Numbers,1);
n_columns = size(Numbers,2);
count = 0;
sum = 0;

for i = 1:n_rows
    for j = 1:n_columns
        if rem(Numbers(i,j), Num) == 0
            count = count + 1;
            sum = sum + Numbers(i,j);
        end
    end
end

if count == 0
    Out = 0;
else
    Out = sum / count;
End

end
```