Coding Competition – 1

- Recall the `while` loop example from previous weak:

- We have written a function named createSeries, that takes an integer input argument "x ", and creates a row vector (A) as output ,consisting of exponentially increasing values of x such that element are $[x^1\ x^2\ x^3\ x^4\ x^5\ \ldots]$) and stops when the summation of elements exceed 100.

- Function takes base value **(x) as input** and gives the row vector **(A) as output**.

Function With While Loop

```
function [A] = createSeries (x)
sumEl = 0; % set the summation of elements to zero
idx = 1; % set index value to 1
while sumEl + x^(idx) < 100
    sumEl = sumEl + x^(idx); % update sumEl
    A(1, idx) = x^(idx); %store new element in array A
    idx = idx + 1; % increase index value
end
```

```
>> createSeries(2)
A =
     2     4     8    16    32
```


Coding Competition – 1

- This coding competition's task is to write two variations of this function by using `for` and `while` loops. You are going to write two functions in which different requirements exist.

- Notice that there is one additional requirement for both of these functions.

Coding Competition – 1

a) Write a function named createSeries1, that takes an integer input argument "x", and creates a row vector (A) as output, consisting of exponentially increasing values of x such that element are $[x^1\ x^2\ x^3\ \ldots]$ and **stops when the summation of elements exceeds 100** or **number of elements exceeds 4.**

- Use a single **`for`** loop.

b) In addition to these requirements, write a function named createSeries2, in which **2nd power of the base is not included** in vector A such that elements are $[x^1\ x^3\ x^4\ \ldots]$. Function should again **stop when the summation of elements exceeds 100** or **number of elements exceeds 4.**

- Use a single **`while`** loop.

  - Both functions should take base value **(x) as input** and give row vector **(A) as output**. Do not use nested loops, use a single loop for each question.

Example

- Try your functions with the following:

a)

```
>> A=createSeries1(2)
A =
     2      4      8     16
```

```
>> A=createSeries1(3)
A =
     3      9     27
```

b)

```
>> A=createSeries2(2)
A =
     2      8     16     32
```

```
>> A=createSeries1(3)
A =
     3     27
```


Coding Competition – 1

- Submission is due on October 2nd 2016 (Wednesday) @23:55.
- Upload your solutions to ODTUCLASS in a compressed file.
- Remember to include screenshots of your results.