Midterm II
Format:
# name , surname, registration number

It will be assumed that the first two lines of your programs are always:(do not rewrite them)

```
import numpy as np
import matplotlib.pyplot as plt
```

1. (a) Write a function called fact that will return an array of factorials. Example : fact(4) will return np.array([1,1,2,6,24]). These numbers are [0!,,1!,2!,3!,4!].

   (b) Write a function `my_sinh(x,n)` that will return

   $$\sum_{k=0}^{k=n-1} \frac{x^{2k+1}}{(2k+1)!}$$

   You may call function `fact` with in `my_sinh`. I prefer that you call it only once. In your main program compute $sinh(2.)$ using 10 terms of the series above.

2. Plot function $\cosh(x{*}x)$ in the region where $\cosh(x{*}x) < 10{*}x{*}x$. Method: divide the interval [-8,8] to 1000 equal part(you may use `linspace` from numpy). Using boolean array extract a subarray where the inequality will be satisfied.

3. In this problem you will compute projectile motion in two dimensions. You analysis will include air resistance. All numerical value are given SI units.

   Create a numpy array M of 4 rows, N columns (M[i,j], first index i must be less than 4,the second index is less than N). Take N=10000. Upon the execution of your program the n^{th} column of this matrix will be x, y, v_x, v_y where x, y are position of a particle at instant $t = nh$ and v_x, v_y are its velocity(at instant nh). The first column of the matrix is
   $[0.0, 0.0, v_0 \cos\pi/4, v_0 \sin\pi/4]$ where v_0 is 200(initial condition)
   The recurrence relation between two consecutive column is:(superscript and subscripts n and $n-1$ refer to instants)

   $$x_n = x_{n-1} + h * v_x^{n-1}$$

   $$y_n = y_{n-1} + h * v_y^{n-1}$$

   $$v_x^n = v_x^{n-1} - h * (v^{n-1} * *2) * \gamma * \cos\theta_{n-1}$$

   $$v_y^n = v_y^{n-1} - h * 9.8 - h * (v^{n-1} * *2) * \gamma * \sin\theta_{n-1}$$

where v^{n-1} is the magnitude of the velocity at instant $(n-1)*h$ and $\theta_{n-1} = \arccos\left(\frac{v_x^{n-1}}{v^{n-1}}\right)$:the angle that velocity make with x axis. Note that $v^{n-1} **2$ is square of the speed. Take γ equal to 1.0e-04.

(a) If air resistance is ignored the flight time will be 30 sec approximately. Take h=30/N=30/10 000 and compute matrix M.

(b) Determine when y becomes negative (projectile hitting the ground) . Find approximately τ (flight time).

(c) Plot y as a function of x for $y > 0$.

4. (a) Function `integrate(f,a,b,n)` returns an approximation of the integral $\int_a^b dx f(x)$ provided by the Riemann sum

$$\frac{b-a}{n} \sum_{k=0}^{n-1} f\left(a + k\frac{b-a}{n}\right)$$

where f accepts a numpy array as argument. Program function integrate without using for loop.

(b) Function `integrate_multi(f,a,b,n,arg=[])` will integrate function f with respect to its first argument (f is a multivariable function). The optional argument arg is a list containing exra arguments of f. Example: If wish to integrate $g(x, y, z) = x^2 + yx + z$ with respect to x from 0 to 10 with value of y=3, and z=4 then the command to call inegration will be `integrate_multi(g,0.,10.,10000,[3.,4.])` where 10 000 is the number of points used in Riemann sum. Program `integrate_multi` and call it to integrate g from 0 to 10 using 10000 points with values of extra argument fixed to 3 and 4.