Intro to Programming

0. Make sure that you have completed all the previous labs.

1. Write a function my_reverse(x) that takes a list of integers and reverses the list “in place”. In other words the function does not create a new list which is a reverse of x and return that. It causes the original list being passed to it to be modified. The function “reverse” does not return anything. It will just modify the list passed to it.

   For this problem **you may not** use the built-in reverse function for lists.

   For example:

   ```
   z=[1,2,3,4]
   my_reverse(z)
   print(z)
   ```

   will print [4,3,2,1]

2. Do the problems on pages page 112 (middle of page) and top of page 113.

3. Write a function PrettyPrint(x) that takes as an argument a two dimensional mXn “matrix” – list of lists, and prints it as a two dimensional table.

   For example, give list z

   ```
   z=[[1,2,3],[4,5,6],[7,8,9],[10,11,12]]
   ```

   the function call

   ```
   PrettyPrint(z)
   ```

   produces

   ```
   >>>
   1   2   3
   4   5   6
   7   8   9
   10  11  12
   >>> |
   ```
4. Write a function `sum_of_odd_cols(x)`, that takes a 2-dimensional list of integers and returns the sum of the values in the odd columns of `x`.

5. Write a function `sum_of_even_cols(x)`, that takes a 2-dimensional list of integers and returns the sum of the values in the even columns of `x`.

6. Write a function `max_pos(x)`, that takes a 2-dimensional list of integers and returns a tuple, `(a,b,c)` where `a` is the maximum value in `x`, and `b` and `c` are the row and column of `x` where that value was found.

7. Write a list comprehension that creates a list `x` with tuples `(a,b)` where `a` and `b` are consecutive odd numbers and `1<=a<b<100`. So `x` would be `[(1,3), (3,5), (5,7) ... (97,99)]`

8. Write a list comprehension that creates a list `x` of all leap years from year 4 to year `n`. 