CS235 HW1 - Teng Xu

1) (3 pts) Implement a second algorithm for generating (pseudo)random numbers, suggested at the slide 10 of Lecture 1. Try different seeds and show the first 10 numbers of the produced sequence. Does the above method produce repeating numbers? What would be the range (possible values) of the numbers? Are the obtained numbers appear to be spread out over the range?

No, the method does not produce repeating numbers.
The range is form 0 to 9999.
Yes, the obtained numbers appear to be spread out over the range.
Output:

```
In [15]: lst(2333,10)
Out[15]: [2333, 2889, 6321, 5041, 1681, 5761, 9121, 2641, 4881, 4161]

In [16]: lst(1997,10)
Out[16]: [1997, 8009, 4081, 4561, 2721, 3841, 3281, 4961, 1521, 3441]

In [17]: lst(1996,10)
Out[17]: [1996, 4016, 8256, 1536, 9296, 5616, 9456, 5936, 6096, 1216]
```

2) (3 pts) Implement the Python function that returns a set of all prime numbers in the range from 2 to n.

Output:

```
In [50]: prime(10)
Out[50]: {2, 3, 5, 7, 9}

In [51]: prime(25)
Out[51]: {2, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25}

In [52]: prime(38)
Out[52]: {2, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37}
```

3) (3 pts) Given a set of year numbers. Perform the following tasks: check that all the numbers are 4 digits numbers starting with digit 1 or 2 (hint: use checkAll). Remove all the numbers that are not in this format. (Hint: you could use functions filter and conversions from list to set and vice versa). Create a new set: leap years - a subset of year numbers that are divisible by 4 but not divisible by 100, or divisible by 400. Check if current year is in this list.
Output:
Checkall:

```
In [24]: check({2222,3333,2345,1111,999})
Out[24]: False

In [25]: check({2222,2345,1111})
Out[25]: True
```

Remove:

```
In [19]: removeset({2222,3333})
Out[19]: {2222}

In [20]: removeset({2222,3333,2345,1111,999})
Out[20]: {1111, 2222, 2345}

In [21]: removeset({999,3999})
Out[21]: set()
```

Leapyear:

```
In [37]: leapyear({2000,2001,2010,2100,2220})
Out[37]: {2000, 2220}

In [38]: leapyear({1000,2222,4444,8888,8000})
Out[38]: {4444, 8000, 8888}
```

Current Year is Leapyear

```
In [3]: leapyear({2016})
Out[3]: {2016}
```