

Report

2019년 11월 29일

1 교체 알고리즘 별 성능 비교 분석

다양한 입력 형태에 따른, 교체 알고리즘의 성능 분석 결과를 도표로 작성

비교 기준

- 가능한 데이터 개수
- 입력 패턴
- 슬롯 개수

으로 적중률과 수행시간 비교

1.1 1. 함수 설명

필요한 라이브러리

```
In [98]: import LRU, FIFO, LFU, RND
         from random import *
         import matplotlib.pyplot as plt
         import matplotlib.gridspec as gridspec
         import numpy as np
         import pandas as pd
         from datetime import datetime
```

적중률 데이터 가져오는 함수 + 수행시간 가져오는 함수

Parameter 설명

- input_size : 가능한 데이터 개수
- input_max_value : 입력 패턴 0 ~ input_max_value사이의 난수

- slot_size : 슬롯 개수

```
In [99]: def get_hit_result(input_size, input_max_size, slot_max_size):
    LRU_list = []
    LFU_list = []
    RND_list = []
    FIFO_list = []

    for slot_size in range(3, slot_max_size):

        tmp_LRU = 0
        tmp_LFU = 0
        tmp_RND = 0
        tmp_FIFO = 0
        for i in range(10):

            input_list = []

            for j in range(input_size):
                input_list.append(randint(0, input_max_size))

            tmp_LRU += LRU.LRU_implement(input_list, slot_size)
            tmp_LFU += LFU.LFU_implement(input_list, slot_size)
            tmp_RND += RND.Random_implement(input_list, slot_size)
            tmp_FIFO += FIFO.FIFO_implement(input_list, slot_size)

            LRU_list.append(tmp_LRU / 10)
            LFU_list.append(tmp_LFU / 10)
            RND_list.append(tmp_RND / 10)
            FIFO_list.append(tmp_FIFO / 10)

        #index = [i for i in range(3,slot_max_size)]
        #index = np.array(index)

    return LRU_list, LFU_list, RND_list, FIFO_list
```

```
In [100]: def get_time_result(input_size, input_max_size, slot_max_size):
```

```

LRU_list = []
LFU_list = []
RND_list = []
FIFO_list = []

for slot_size in range(3, slot_max_size):

    time_LRU = 0
    time_LFU = 0
    time_RND = 0
    time_FIFO = 0
    for i in range(10):

        input_list = []

        for j in range(input_size):
            input_list.append(randint(0, input_max_size))

        starttime = datetime.now()
        LRU.LRU_implement(input_list, slot_size)
        tmpTime_LRU = (datetime.now()-starttime).microseconds

        starttime = datetime.now()
        LFU.LFU_implement(input_list, slot_size)
        tmpTime_LFU = (datetime.now()-starttime).microseconds

        starttime = datetime.now()
        RND.Random_implement(input_list, slot_size)
        tmpTime_FIFO = (datetime.now()-starttime).microseconds

        starttime = datetime.now()
        FIFO.FIFO_implement(input_list, slot_size)
        tmpTime_RND = (datetime.now()-starttime).microseconds

    time_LRU += tmpTime_LRU
    time_LFU += tmpTime_LFU

```

```

        time_RND += tmpTime_RND
        time_FIFO += tmpTime_FIFO

    LRU_list.append(time_LRU / 10)
    LFU_list.append(time_LFU / 10)
    RND_list.append(time_RND / 10)
    FIFO_list.append(time_FIFO / 10)

    #index = [i for i in range(3,slot_max_size)]
    #index = np.array(index)

    return LRU_list, LFU_list, RND_list, FIFO_list

```

도표 작성하는 함수

```

In [101]: def visualize(input_size, input_max_size):

    hit_avg, time_avg = [], []

    fig = plt.figure(figsize = (11,20))
    gs = gridspec.GridSpec(4,2)

    slot_list = [25,50,75,100]

    for i,slot_size in zip(range(4), slot_list):

        LRU_list, LFU_list, RND_list, FIFO_list = get_hit_result(input_size, input_max_size)
        LRU_tlist, LFU_tlist, RND_tlist, FIFO_tlist = get_time_result(input_size, input_max_size)

        def mean(list):
            return round(sum(list) / len(list), 2)

        hit_list = [LRU_list, LFU_list, RND_list, FIFO_list]
        time_list = [LRU_tlist, LFU_tlist, RND_tlist, FIFO_tlist]

        for hit, time in zip(hit_list, time_list):
            hit_avg.append(mean(hit))

```

```

        time_avg.append(mean(time))

index = ['LRU', 'LFU', 'RND', 'FIFO']
columns = [i for i in range(3,slot_size)]

ax1 = fig.add_subplot(gs[i,0])
plt.plot(columns, LRU_list, label = 'LRU')
plt.plot(columns, LFU_list, label = 'LFU')
plt.plot(columns, FIFO_list, label = 'FIFO')
plt.plot(columns, RND_list, label = 'RND')
plt.scatter(columns, LRU_list)
plt.scatter(columns, LFU_list)
plt.scatter(columns, RND_list)
plt.scatter(columns, FIFO_list)
plt.title("Slot Size : " + str(slot_size))
plt.ylabel("HIT Ratio")
plt.xlabel("Slot Size")
plt.legend(loc='best')

ax2 = fig.add_subplot(gs[i,1])
plt.plot(columns, LRU_tlist, label = 'LRU')
plt.plot(columns, LFU_tlist, label = 'LFU')
plt.plot(columns, FIFO_tlist, label = 'FIFO')
plt.plot(columns, RND_tlist, label = 'RND')
plt.scatter(columns, LRU_tlist)
plt.scatter(columns, LFU_tlist)
plt.scatter(columns, FIFO_tlist)
plt.scatter(columns, RND_tlist)
plt.title("Slot Size : " + str(slot_size))
plt.ylabel("Excution Time")
plt.xlabel("Slot Size")
plt.legend(loc='best')

index = [
    'LRU / 25', 'LFU / 25', 'RND / 25', 'FIFO / 25',
    'LRU / 50', 'LFU / 50', 'RND / 50', 'FIFO / 50',

```

```

        'LRU / 75', 'LFU / 75', 'RND / 75', 'FIFO / 75',
        'LRU / 100', 'LFU / 100', 'RND / 100', 'FIFO / 100',
    ]
    print("가능한 입력 수가 %d이며 난수의 최대 수가 %d일 때" % (input_size, input_max_size))
    return pd.DataFrame([hit_avg, time_avg], columns = index, index = ['HIT', 'TIME'])
plt.show()

```

In []:

1.2 2. 알고리즘 별 비교

2.1 가능한 입력 수가 25일 때

2.1.1 난수의 최대 수가 25일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

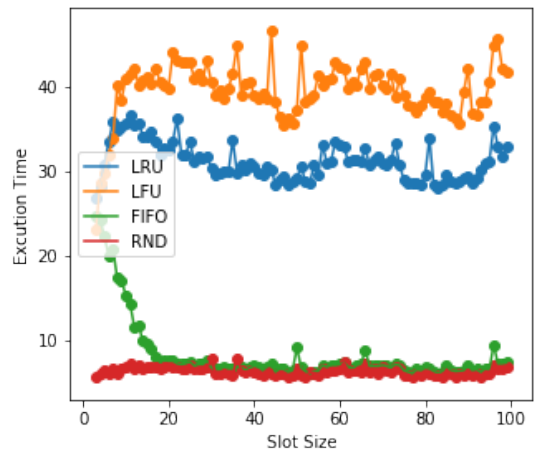
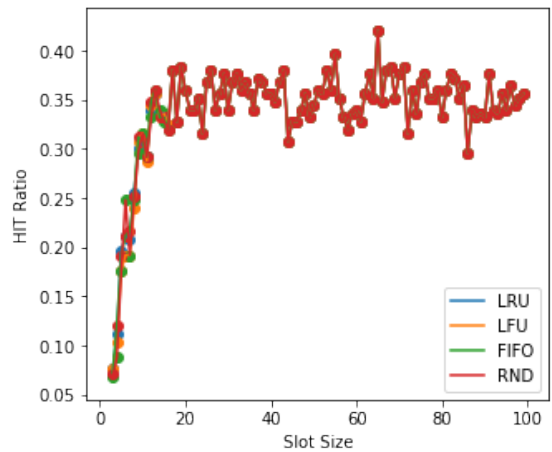
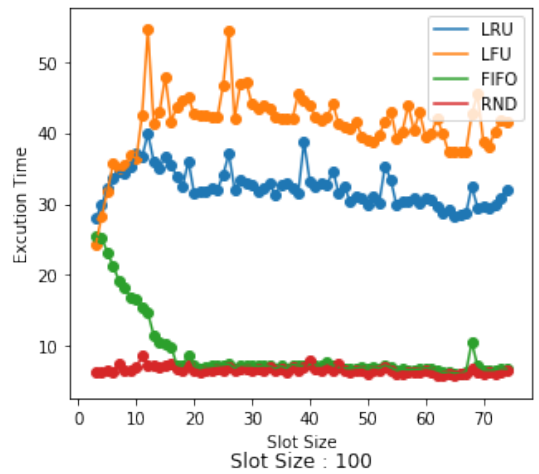
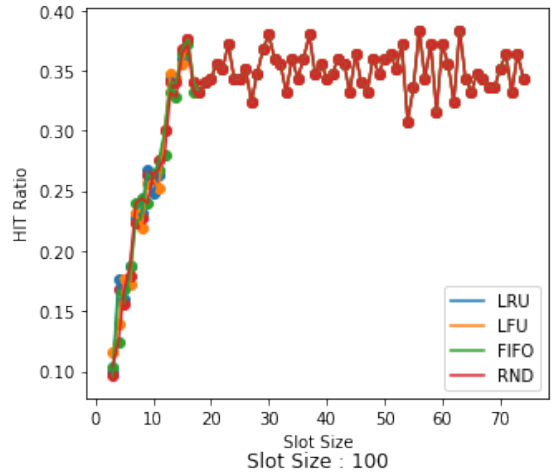
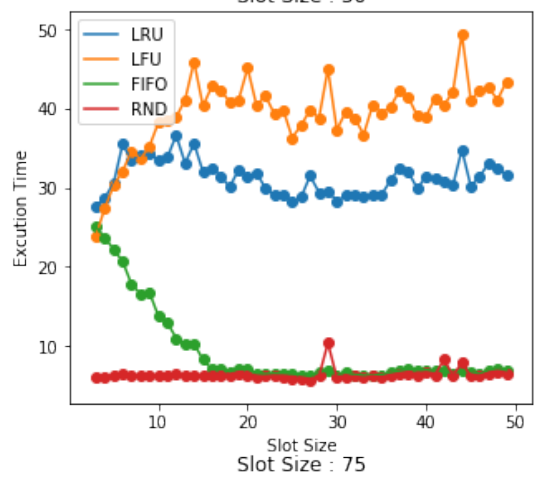
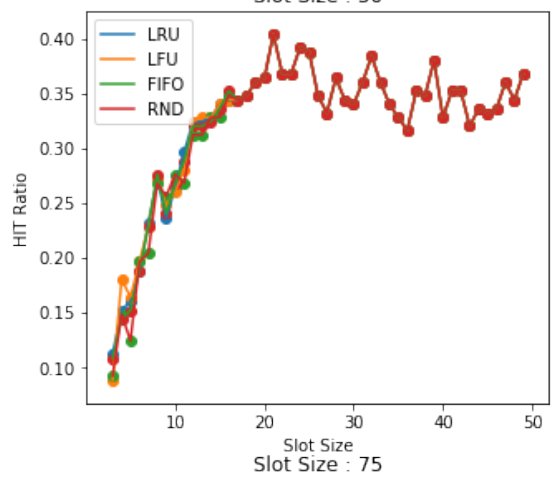
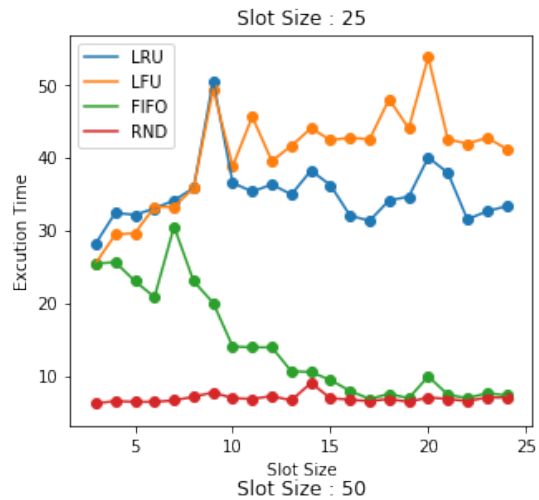
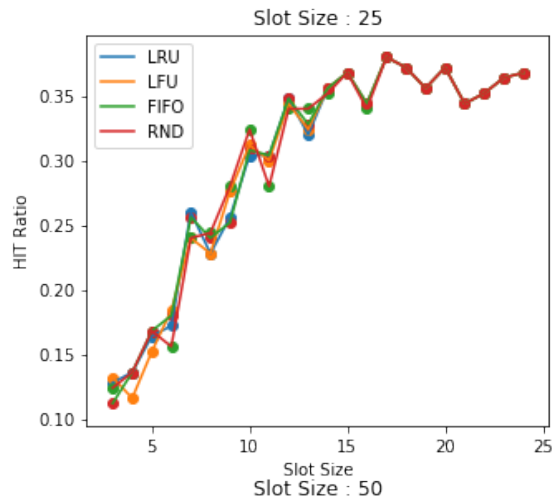
In [118]: visualize(input_size = 25, input_max_size = 25)

가능한 입력 수가 25이며 난수의 최대 수가 25일 때

```

Out[118]:
           HIT  TIME
LRU / 25    0.30  35.04
LFU / 25    0.30  40.33
RND / 25    0.30   6.86
FIFO / 25    0.30  14.04
LRU / 50    0.32  31.27
LFU / 50    0.32  39.34
RND / 50    0.32   6.43
FIFO / 50    0.32   9.34
LRU / 75    0.33  32.33
LFU / 75    0.33  41.39
RND / 75    0.33   6.57
FIFO / 75    0.33   8.93
LRU / 100   0.34  31.16
LFU / 100   0.34  39.54
RND / 100   0.34   6.33
FIFO / 100   0.34   8.21

```



In []:

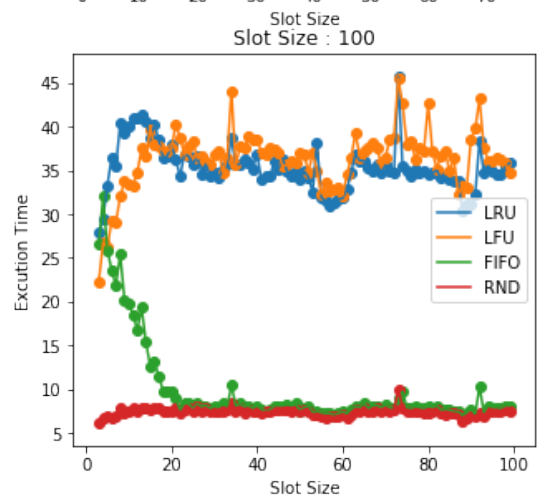
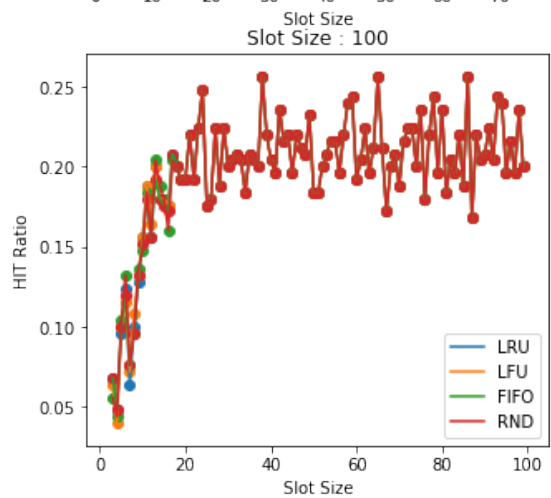
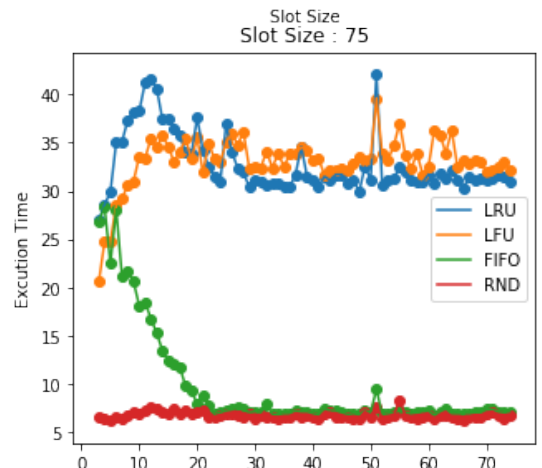
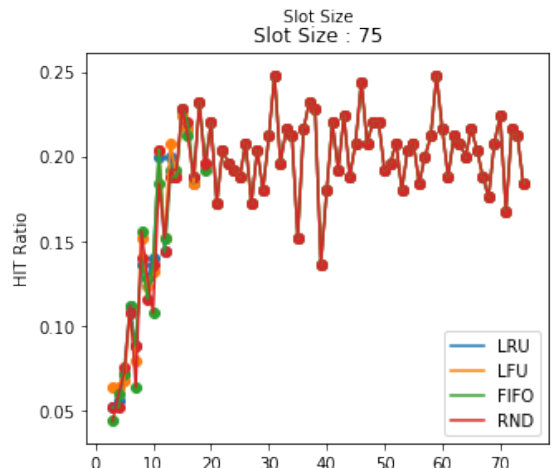
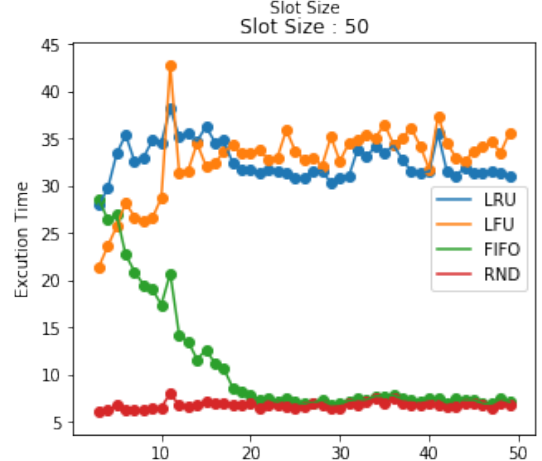
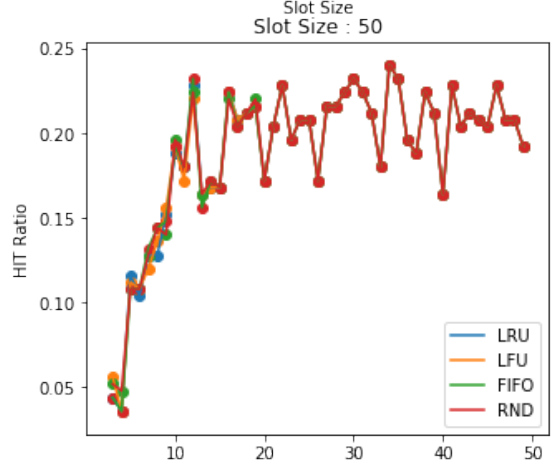
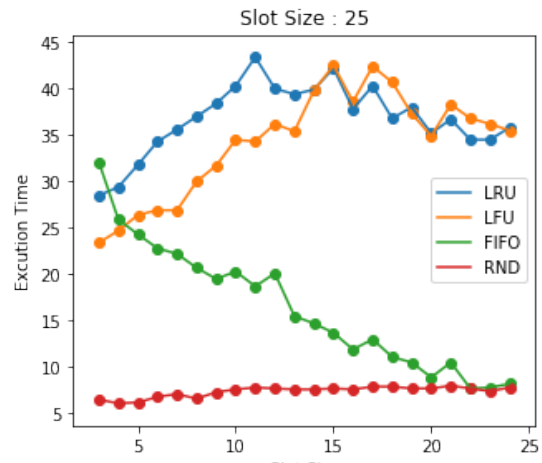
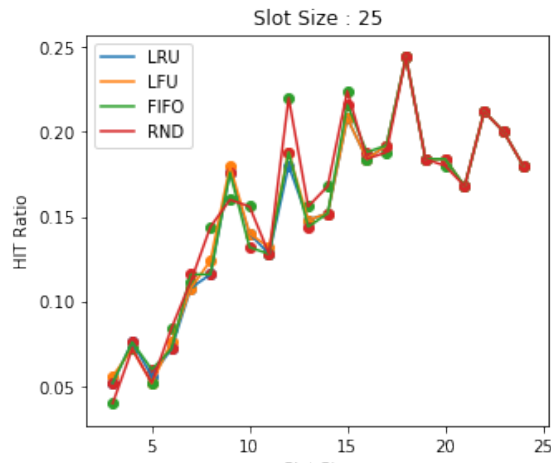
2.1.2 난수의 최대 수가 50일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

In [103]: visualize(input_size = 25, input_max_size = 50)

가능한 입력 수가 25이며 난수의 최대 수가 50일 때

Out [103]:

	HIT	TIME
LRU / 25	0.15	36.81
LFU / 25	0.15	34.25
RND / 25	0.16	7.38
FIFO / 25	0.15	16.37
LRU / 50	0.19	32.61
LFU / 50	0.19	32.74
RND / 50	0.19	6.75
FIFO / 50	0.19	10.87
LRU / 75	0.19	32.76
LFU / 75	0.19	33.07
RND / 75	0.19	6.82
FIFO / 75	0.19	9.81
LRU / 100	0.20	35.34
LFU / 100	0.20	36.25
RND / 100	0.20	7.41
FIFO / 100	0.20	9.88



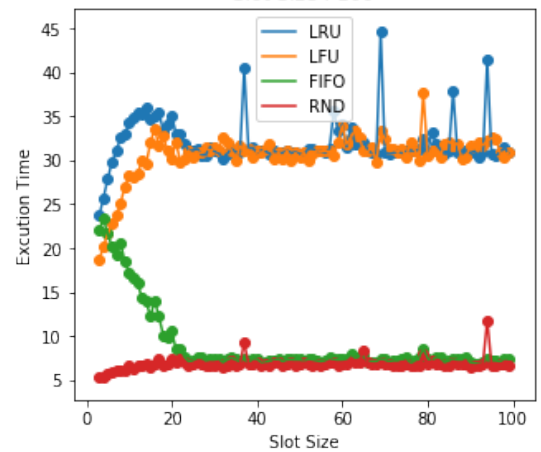
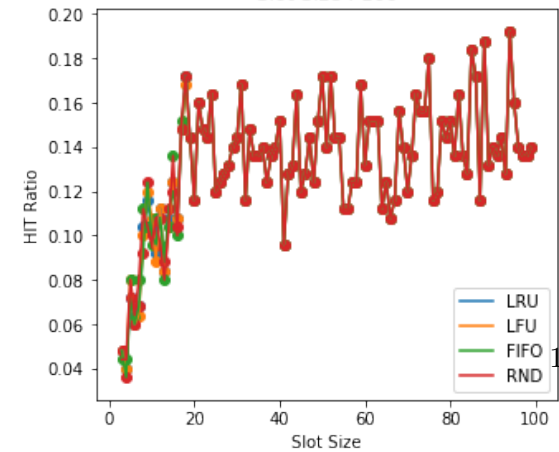
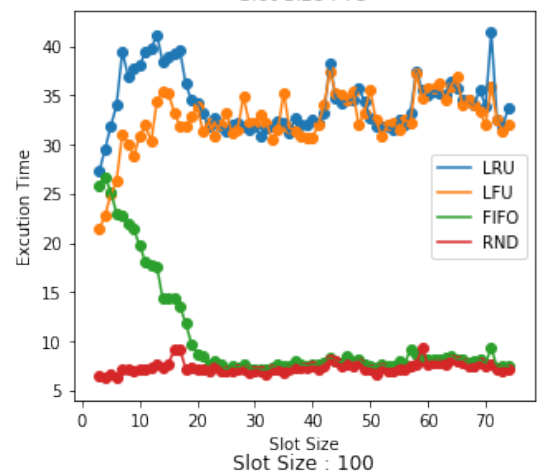
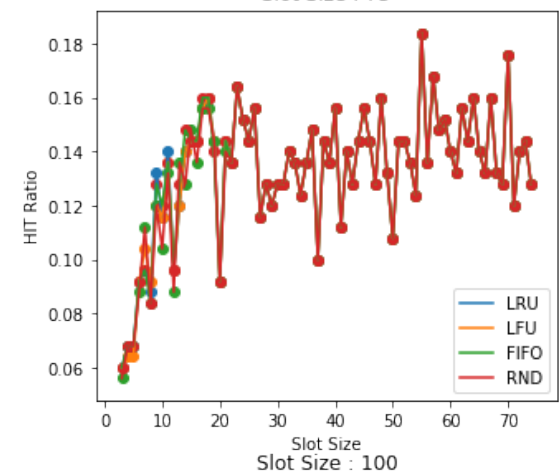
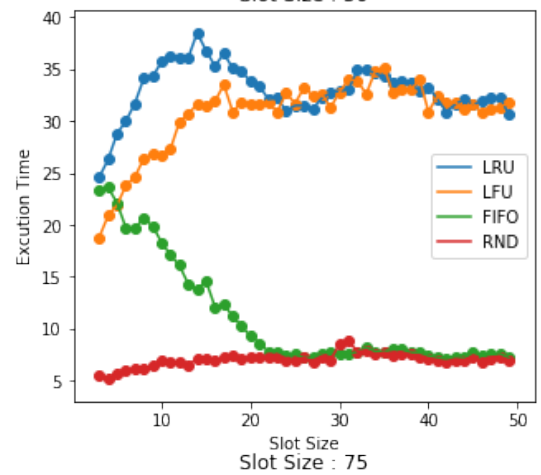
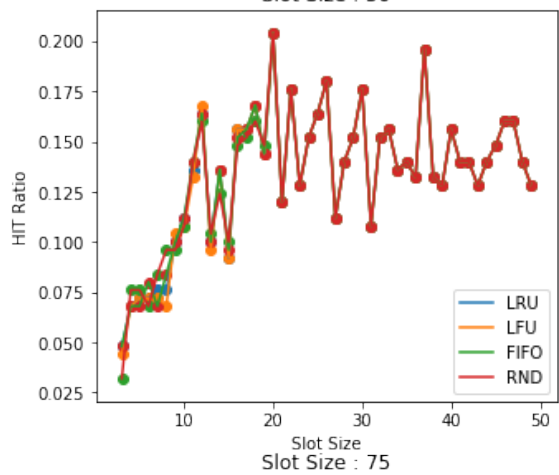
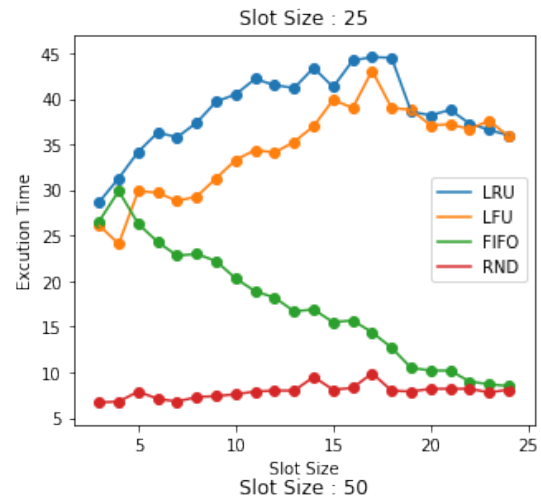
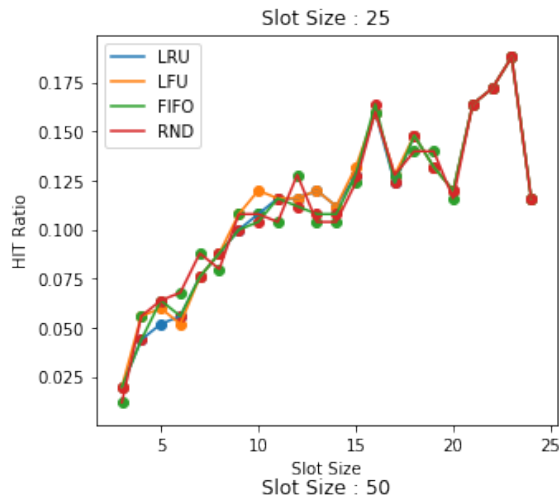
2.1.3 난수의 최대 수가 75일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [104]: visualize(input_size = 25, input_max_size = 75)
```

가능한 입력 수가 25이며 난수의 최대 수가 75일 때

```
Out[104]:
```

	HIT	TIME
LRU / 25	0.11	38.74
LFU / 25	0.11	34.44
RND / 25	0.11	7.90
FIFO / 25	0.11	17.34
LRU / 50	0.13	32.98
LFU / 50	0.13	30.64
RND / 50	0.13	7.01
FIFO / 50	0.13	11.04
LRU / 75	0.13	34.23
LFU / 75	0.13	32.55
RND / 75	0.13	7.34
FIFO / 75	0.13	10.39
LRU / 100	0.13	31.92
LFU / 100	0.13	30.59
RND / 100	0.13	6.85
FIFO / 100	0.13	8.98



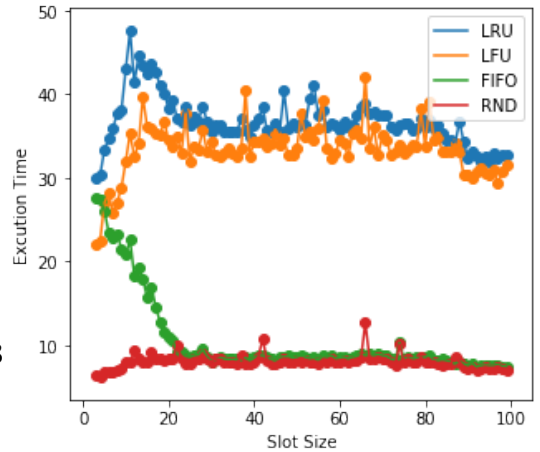
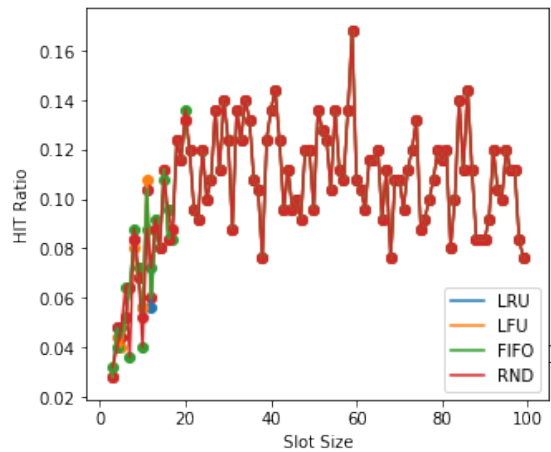
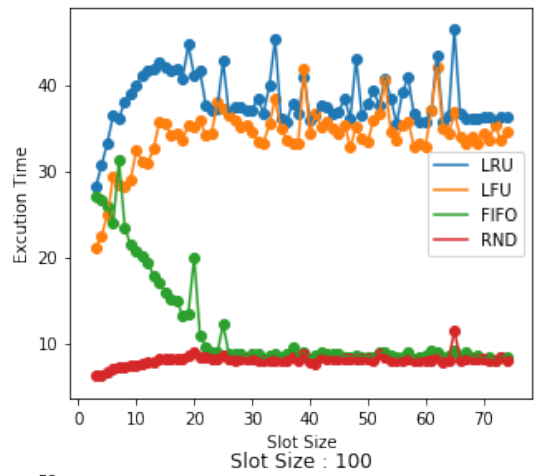
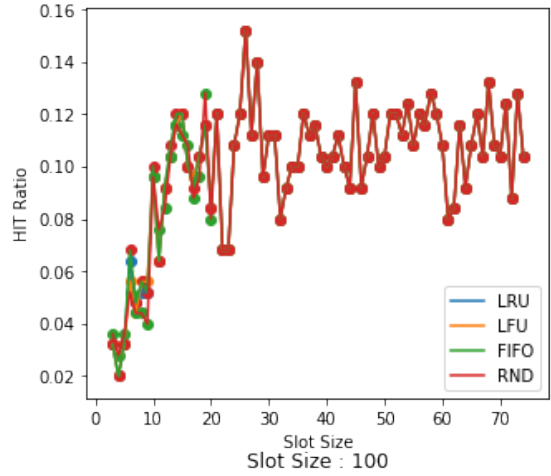
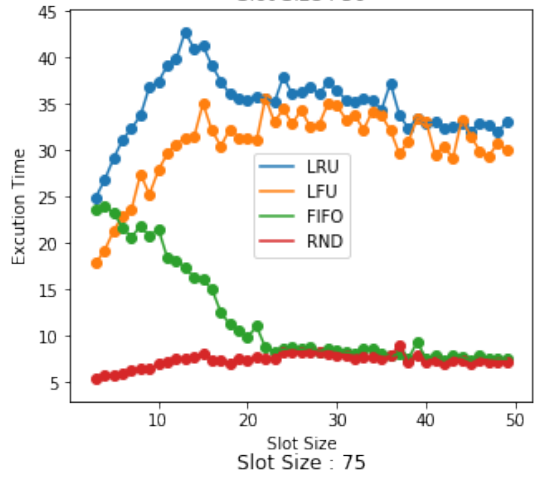
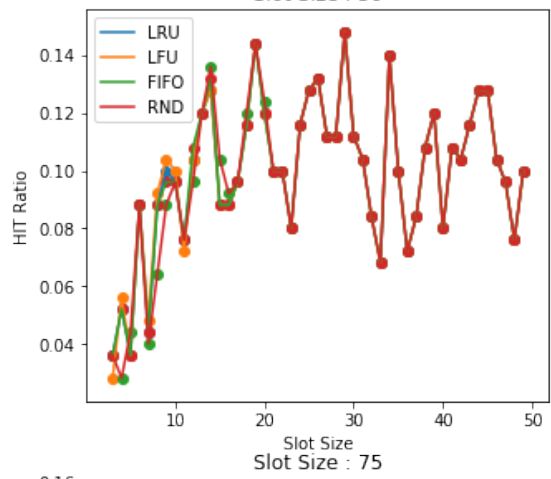
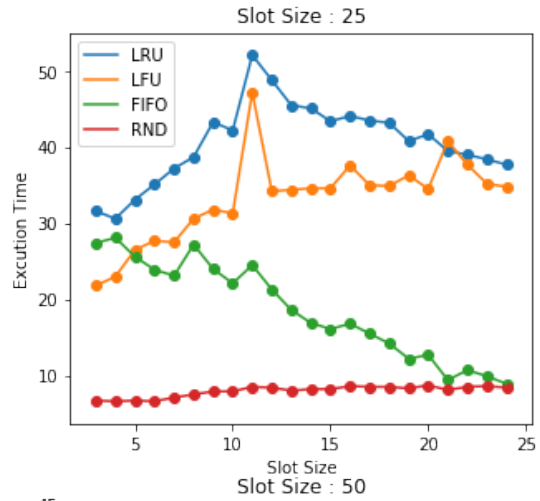
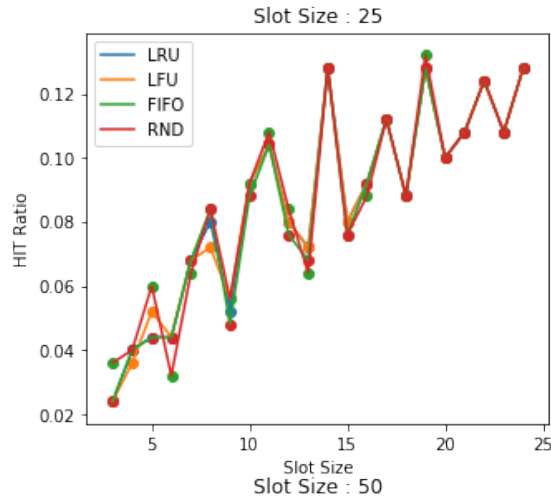
2.1.4 난수의 최대 수가 100일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [105]: visualize(input_size = 25, input_max_size = 100)
```

가능한 입력 수가 25이며 난수의 최대 수가 100일 때

```
Out[105]:
```

	HIT	TIME
LRU / 25	0.09	40.66
LFU / 25	0.09	33.27
RND / 25	0.09	7.93
FIFO / 25	0.09	18.60
LRU / 50	0.10	34.91
LFU / 50	0.10	30.56
RND / 50	0.10	7.36
FIFO / 50	0.10	11.91
LRU / 75	0.10	38.19
LFU / 75	0.10	34.08
RND / 75	0.10	8.10
FIFO / 75	0.10	11.68
LRU / 100	0.11	36.76
LFU / 100	0.11	33.44
RND / 100	0.11	8.10
FIFO / 100	0.11	10.53



2.2 가능한 입력 수가 50일 때

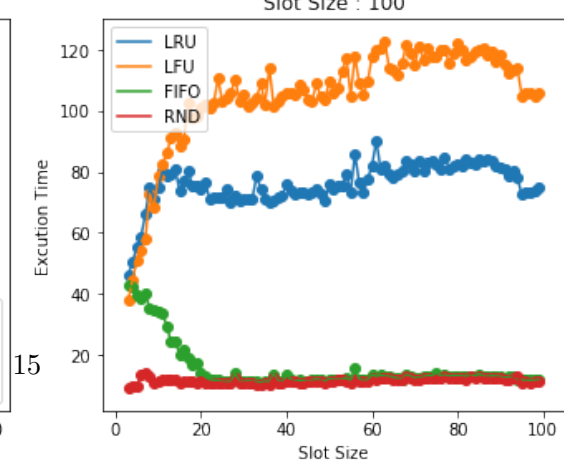
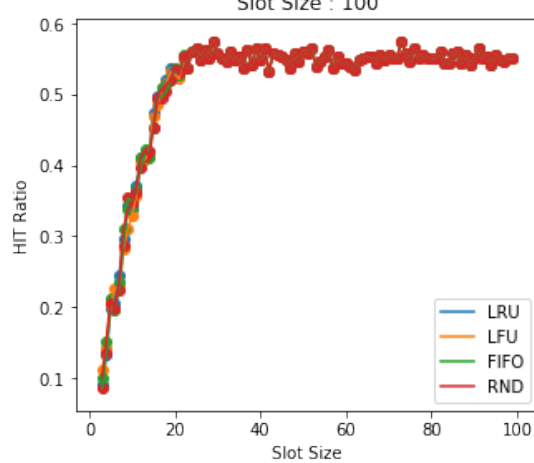
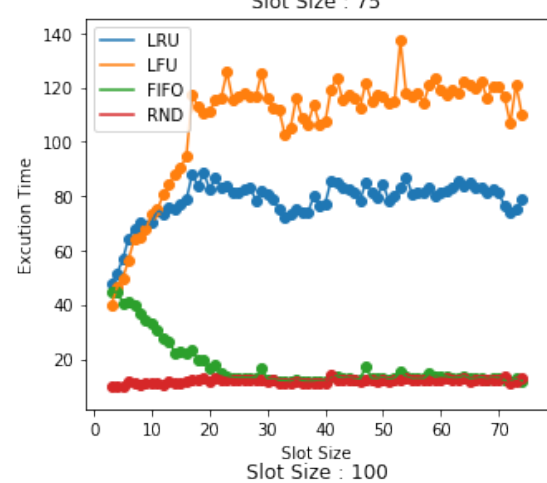
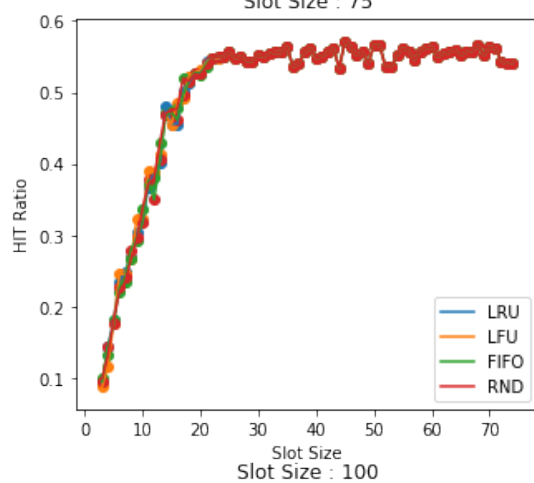
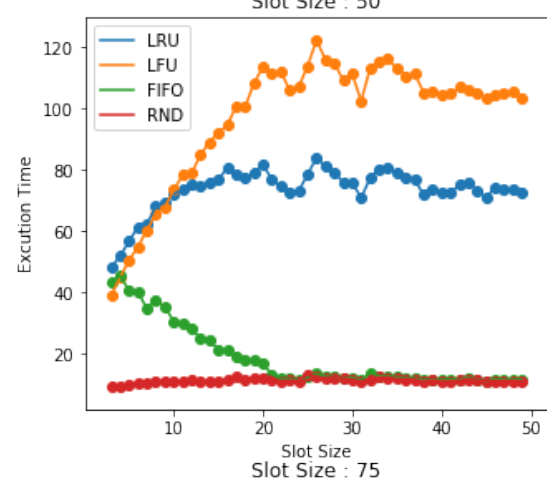
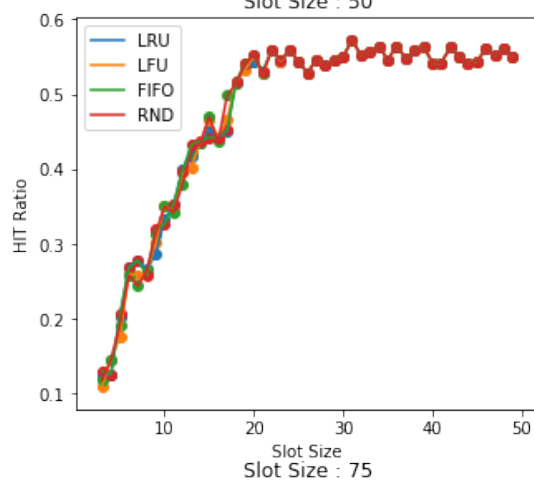
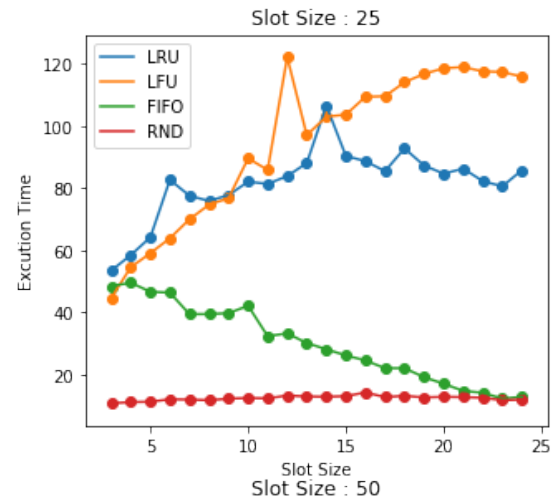
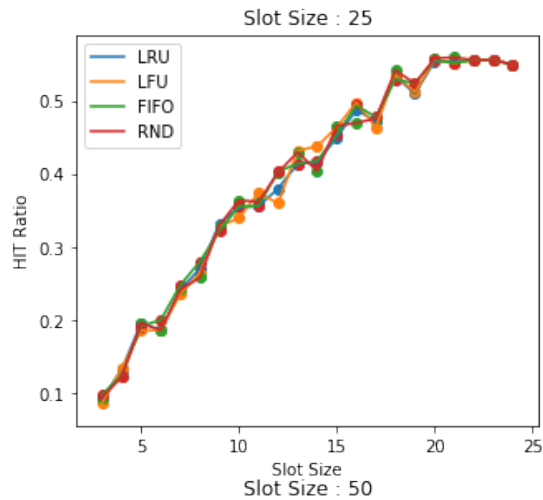
2.2.1 난수의 최대 수가 25일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [106]: visualize(input_size = 50, input_max_size = 25)
```

가능한 입력 수가 50이며 난수의 최대 수가 25일 때

```
Out [106]:
```

	HIT	TIME
LRU / 25	0.39	81.67
LFU / 25	0.39	94.71
RND / 25	0.39	12.45
FIFO / 25	0.39	30.08
LRU / 50	0.48	73.56
LFU / 50	0.48	97.05
RND / 50	0.48	11.43
FIFO / 50	0.48	18.79
LRU / 75	0.50	78.57
LFU / 75	0.50	107.31
RND / 75	0.50	12.06
FIFO / 75	0.50	17.48
LRU / 100	0.52	76.15
LFU / 100	0.52	105.12
RND / 100	0.52	11.69
FIFO / 100	0.52	15.57



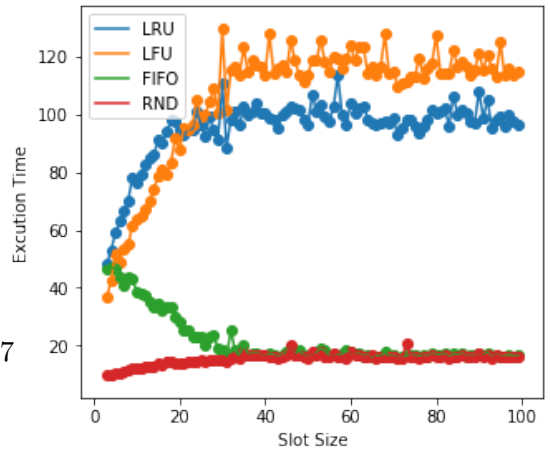
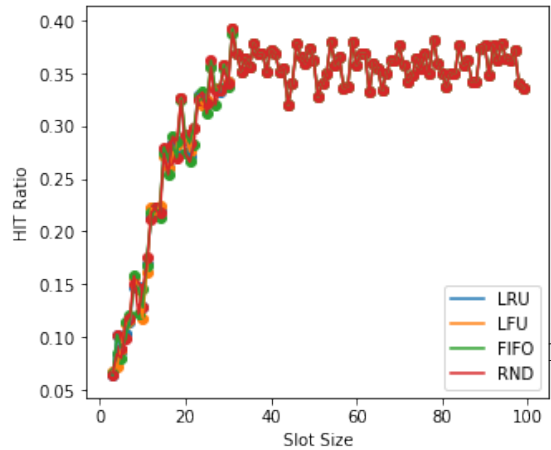
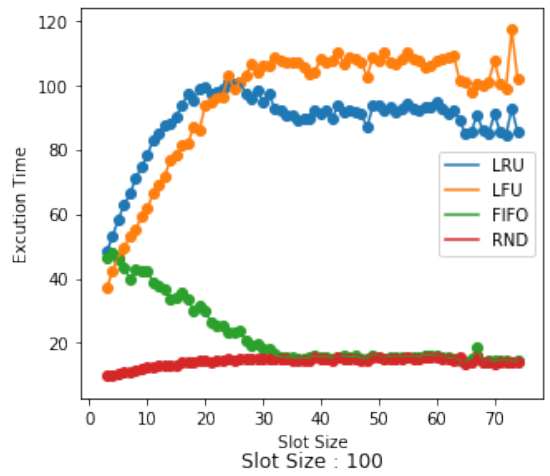
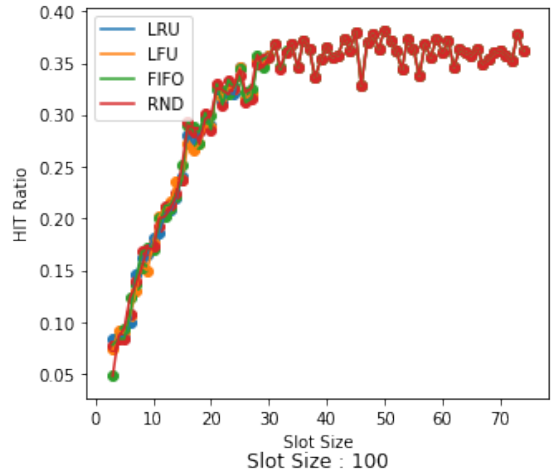
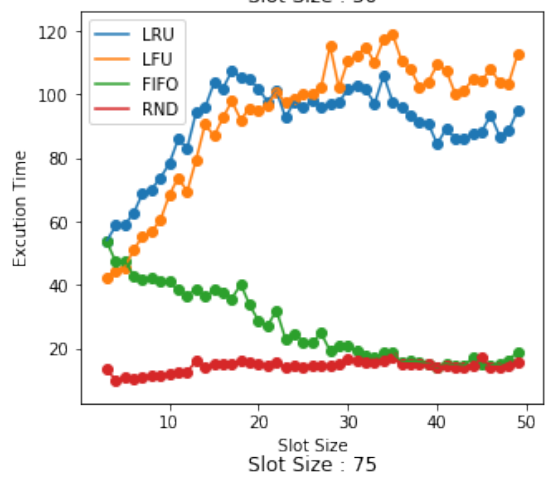
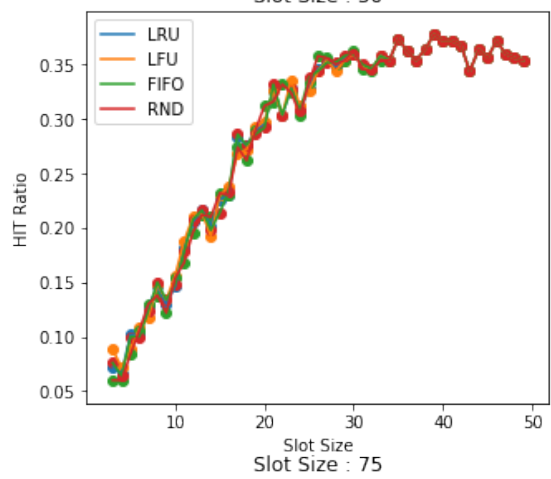
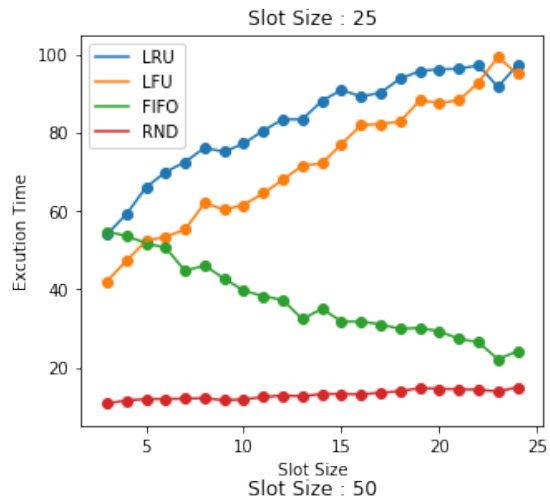
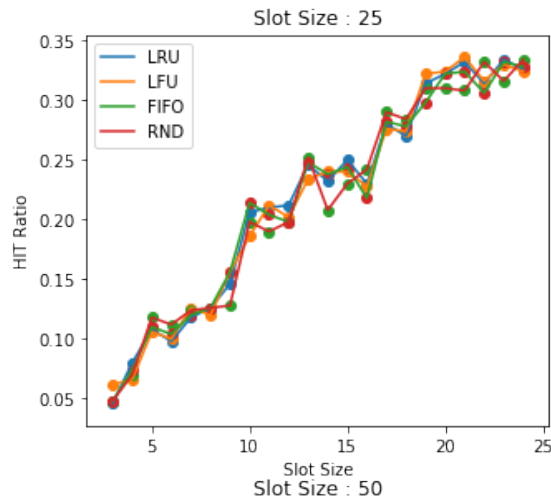
2.2.2 난수의 최대 수가 50일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [107]: visualize(input_size = 50, input_max_size = 50)
```

가능한 입력 수가 50이며 난수의 최대 수가 50일 때

```
Out[107]:
```

	HIT	TIME
LRU / 25	0.22	82.90
LFU / 25	0.22	72.05
RND / 25	0.21	12.95
FIFO / 25	0.22	36.80
LRU / 50	0.29	90.36
LFU / 50	0.29	93.10
RND / 50	0.29	14.39
FIFO / 50	0.29	26.99
LRU / 75	0.32	89.14
LFU / 75	0.32	95.58
RND / 75	0.32	14.28
FIFO / 75	0.32	22.24
LRU / 100	0.32	95.32
LFU / 100	0.32	106.07
RND / 100	0.32	15.51
FIFO / 100	0.32	21.53



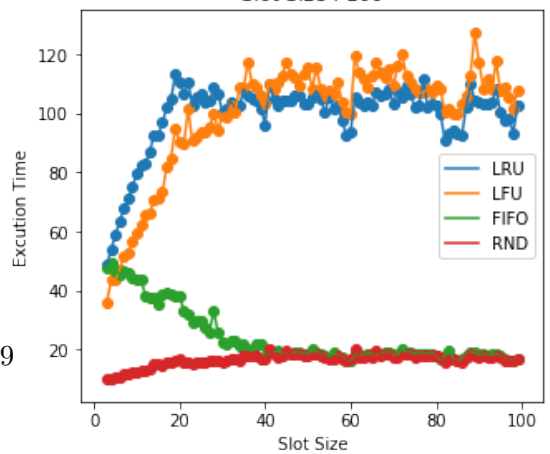
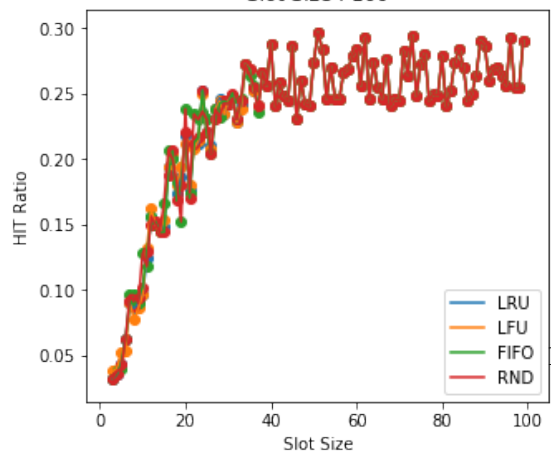
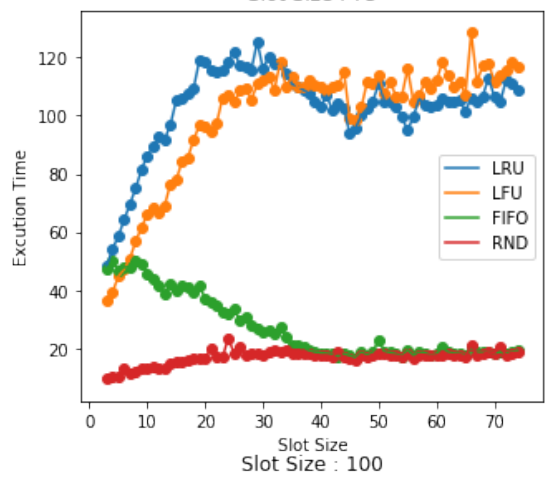
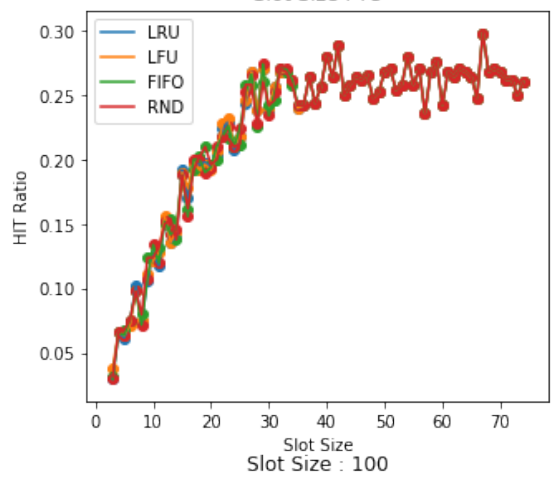
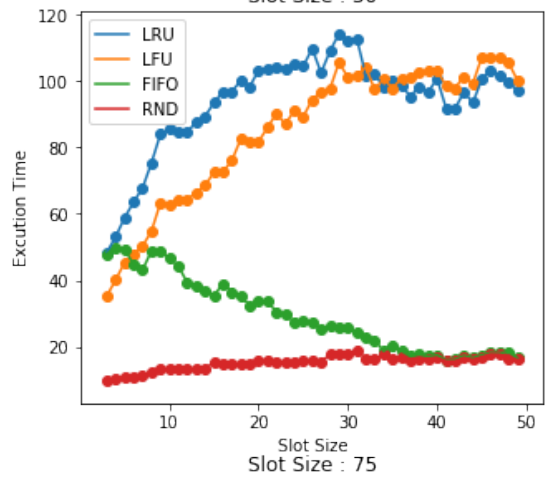
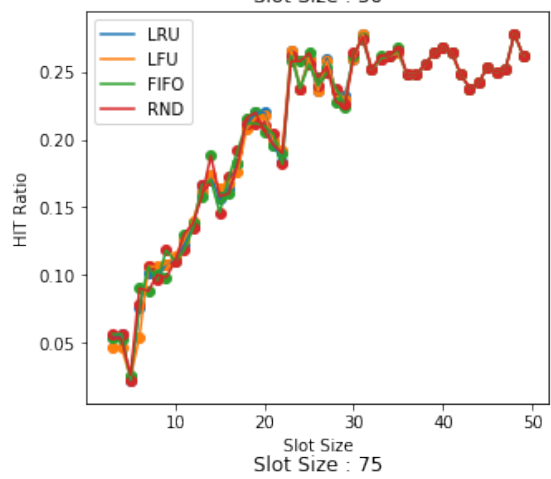
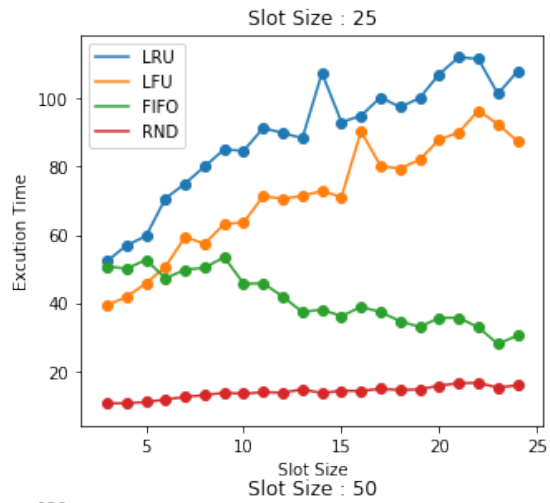
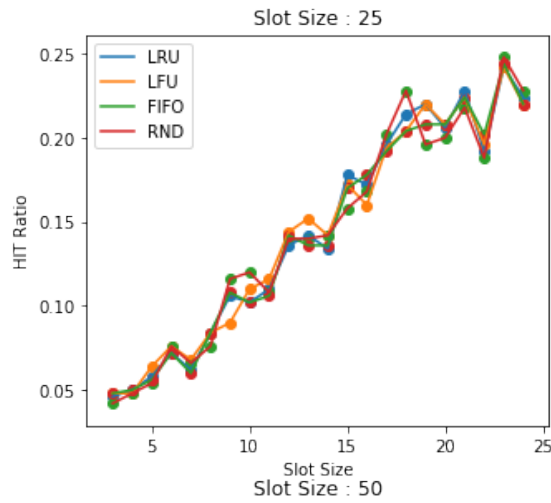
2.2.3 난수의 최대 수가 75일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [108]: visualize(input_size = 50, input_max_size = 75)
```

가능한 입력 수가 50이며 난수의 최대 수가 75일 때

```
Out[108]:
```

	HIT	TIME
LRU / 25	0.14	89.47
LFU / 25	0.14	71.17
RND / 25	0.14	13.86
FIFO / 25	0.14	41.20
LRU / 50	0.20	94.11
LFU / 50	0.20	85.27
RND / 50	0.21	15.17
FIFO / 50	0.20	29.38
LRU / 75	0.23	102.91
LFU / 75	0.23	99.62
RND / 75	0.23	17.27
FIFO / 75	0.23	27.32
LRU / 100	0.23	99.27
LFU / 100	0.23	99.50
RND / 100	0.23	16.61
FIFO / 100	0.23	24.02



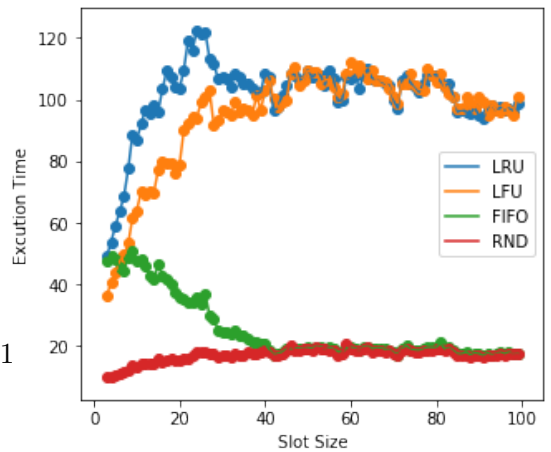
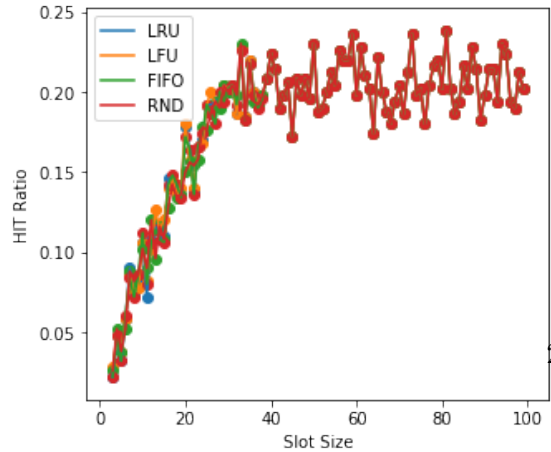
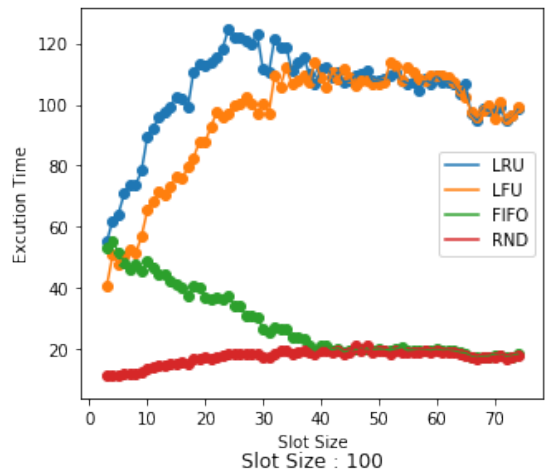
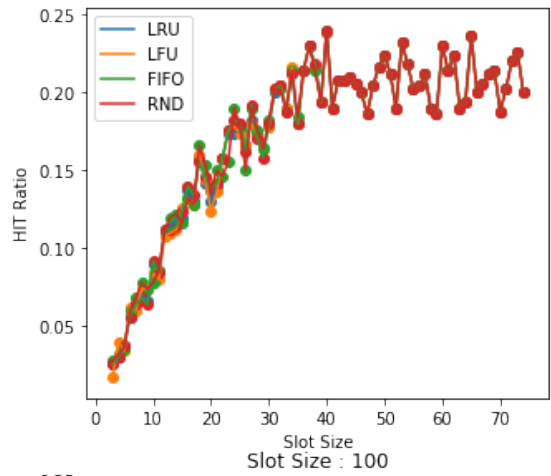
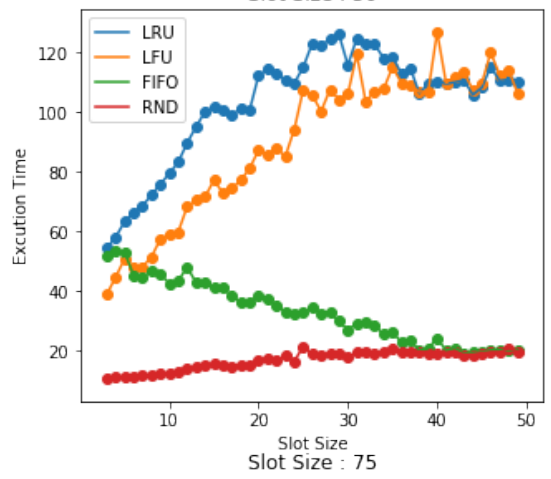
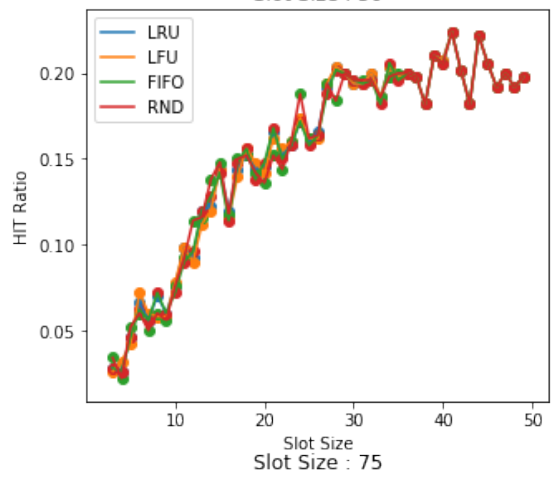
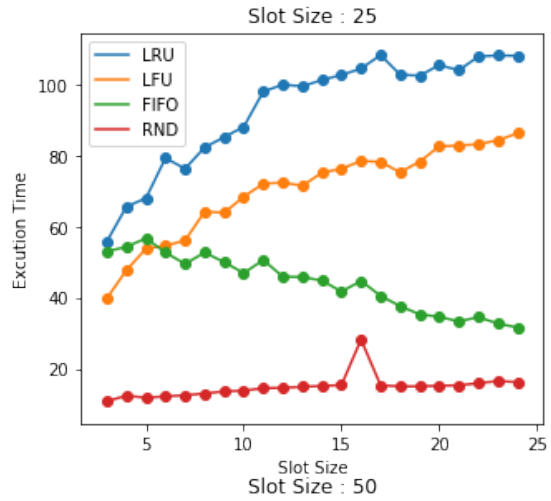
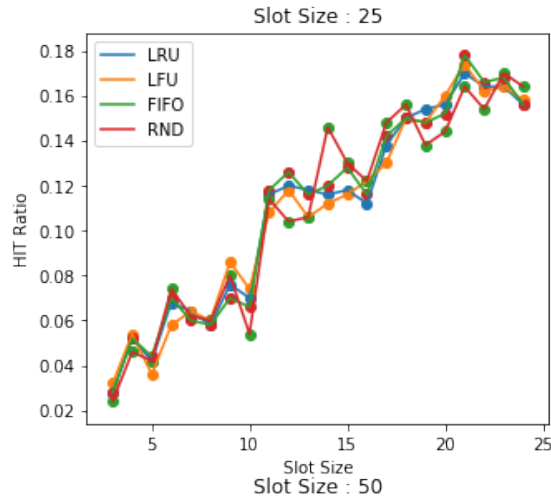
2.2.4 난수의 최대 수가 100일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [109]: visualize(input_size = 50, input_max_size = 100)
```

가능한 입력 수가 50이며 난수의 최대 수가 100일 때

```
Out[109]:
```

	HIT	TIME
LRU / 25	0.11	93.46
LFU / 25	0.11	70.35
RND / 25	0.11	14.97
FIFO / 25	0.11	44.16
LRU / 50	0.15	103.25
LFU / 50	0.15	90.26
RND / 50	0.15	16.80
FIFO / 50	0.15	32.85
LRU / 75	0.17	104.27
LFU / 75	0.17	95.14
RND / 75	0.17	17.32
FIFO / 75	0.17	28.18
LRU / 100	0.18	101.36
LFU / 100	0.18	94.69
RND / 100	0.18	17.08
FIFO / 100	0.18	25.24



2.3 가능한 입력 수가 75일 때

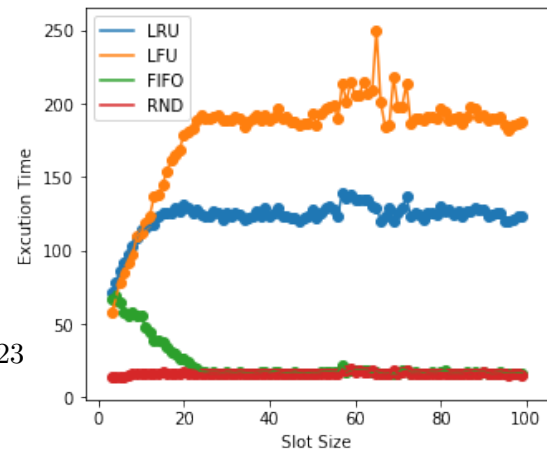
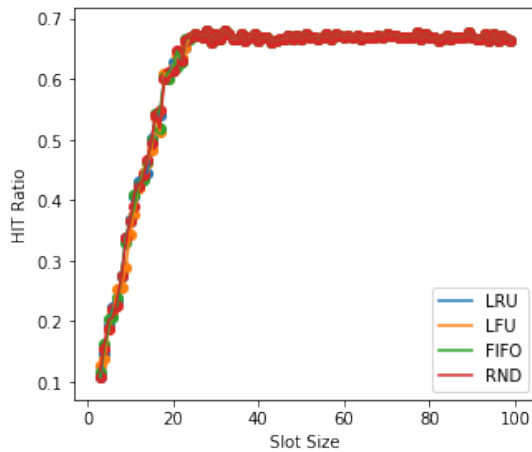
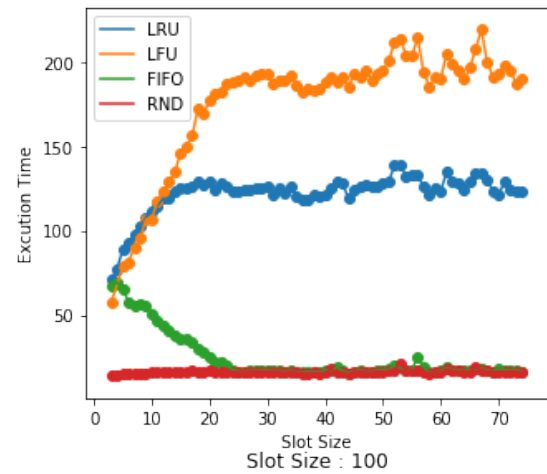
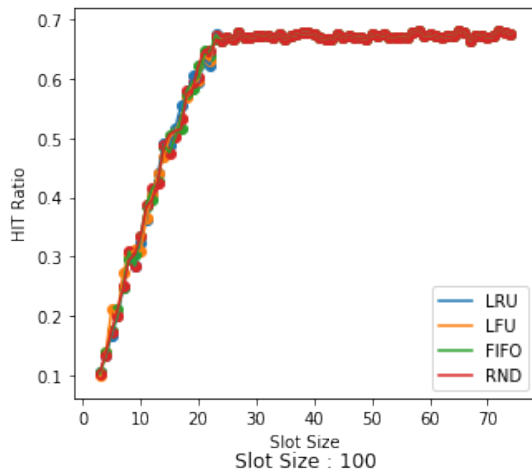
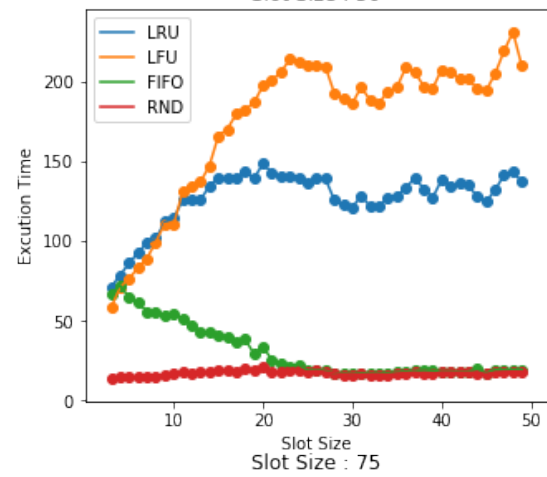
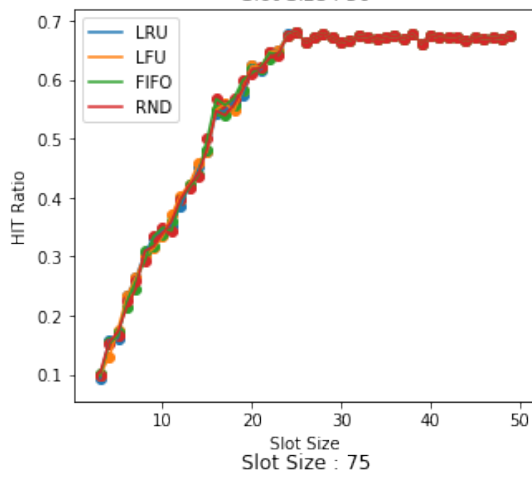
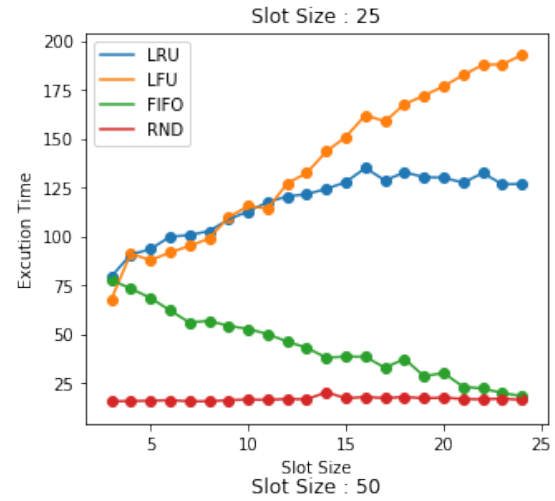
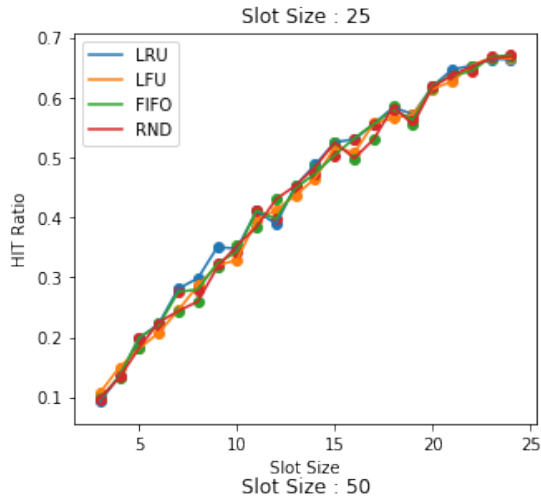
2.3.1 난수의 최대 수가 25일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

In [110]: visualize(input_size = 75, input_max_size = 25)

가능한 입력 수가 75이며 난수의 최대 수가 25일 때

Out [110]:

	HIT	TIME
LRU / 25	0.44	116.73
LFU / 25	0.43	136.91
RND / 25	0.43	16.76
FIFO / 25	0.44	43.94
LRU / 50	0.56	126.95
LFU / 50	0.56	174.27
RND / 50	0.56	17.19
FIFO / 50	0.56	30.33
LRU / 75	0.60	122.40
LFU / 75	0.60	175.26
RND / 75	0.60	16.38
FIFO / 75	0.60	24.83
LRU / 100	0.62	123.22
LFU / 100	0.62	180.20
RND / 100	0.62	16.40
FIFO / 100	0.62	22.66



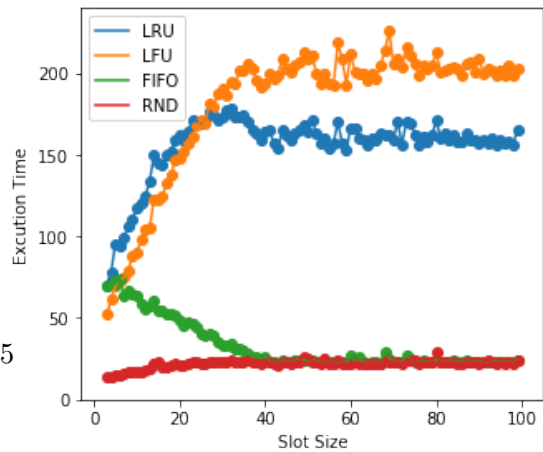
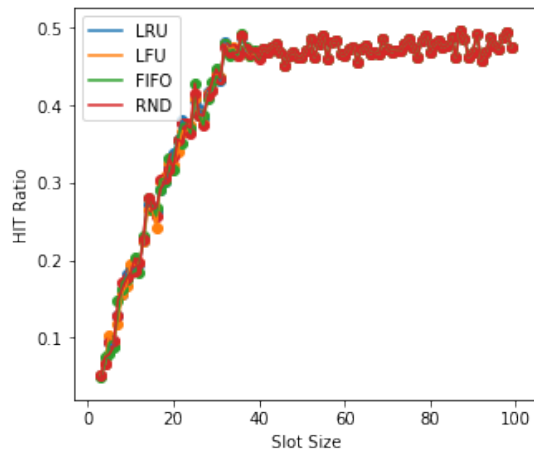
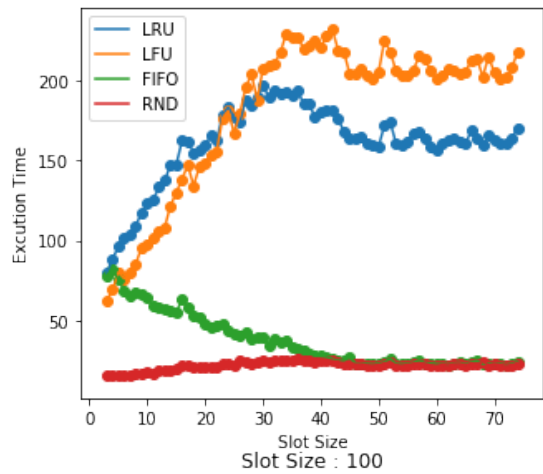
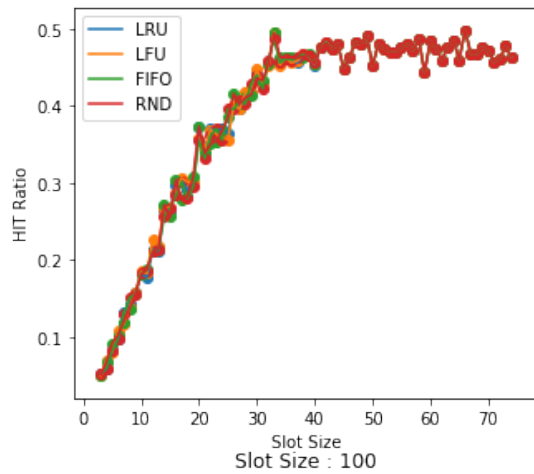
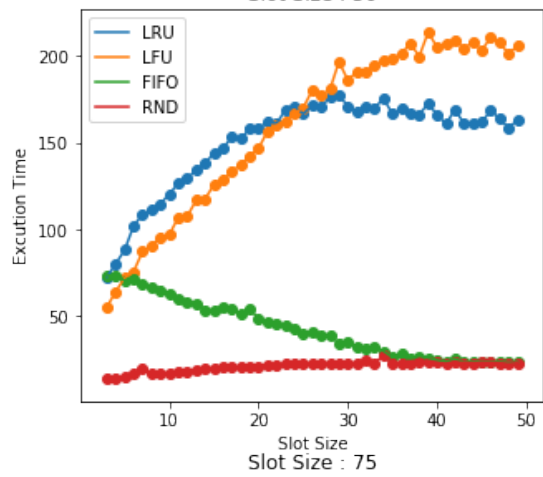
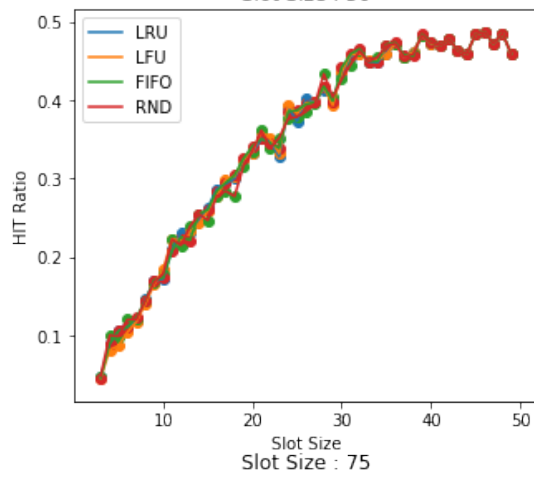
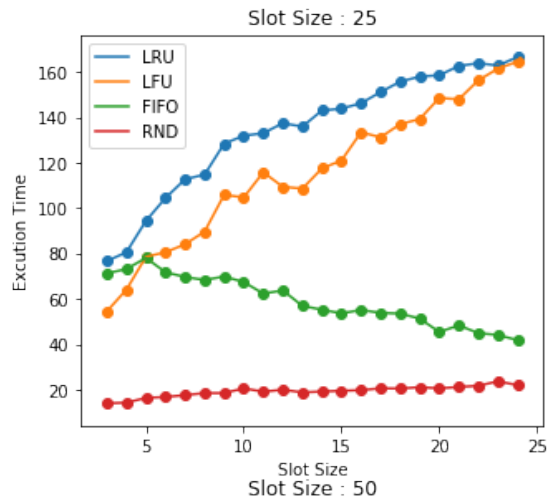
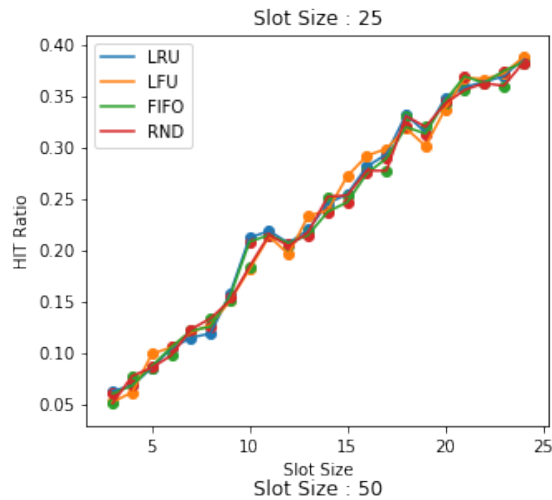
2.3.2 난수의 최대 수가 50일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [111]: visualize(input_size = 75, input_max_size = 50)
```

가능한 입력 수가 75이며 난수의 최대 수가 50일 때

```
Out[111]:
```

	HIT	TIME
LRU / 25	0.23	134.65
LFU / 25	0.23	116.09
RND / 25	0.23	19.40
FIFO / 25	0.23	59.15
LRU / 50	0.35	150.79
LFU / 50	0.35	159.27
RND / 50	0.35	21.21
FIFO / 50	0.35	42.55
LRU / 75	0.39	161.14
LFU / 75	0.39	181.60
RND / 75	0.39	22.53
FIFO / 75	0.39	37.79
LRU / 100	0.41	155.82
LFU / 100	0.41	180.71
RND / 100	0.41	21.99
FIFO / 100	0.41	32.73



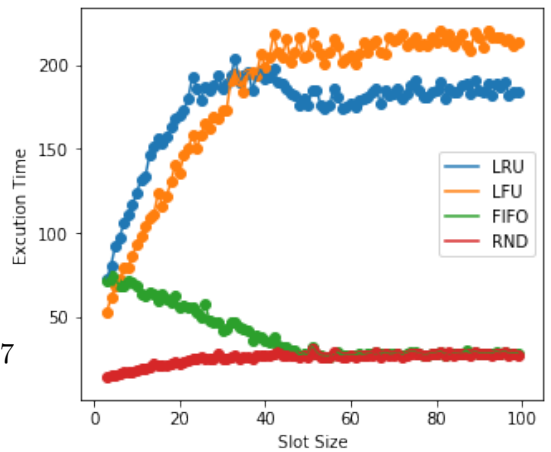
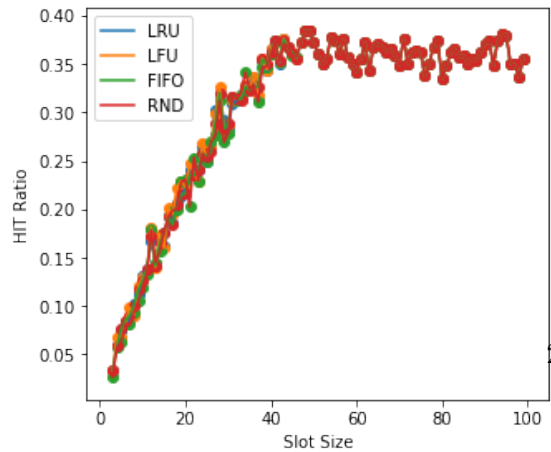
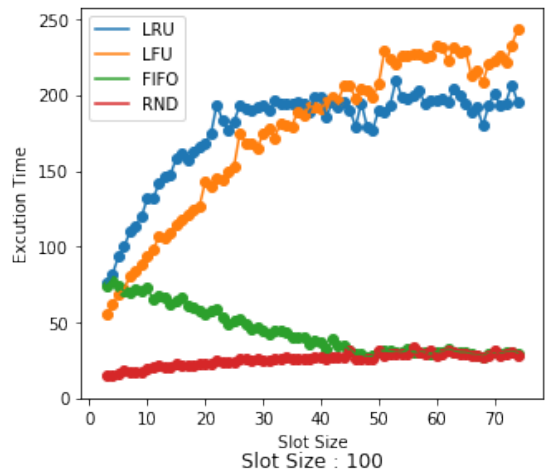
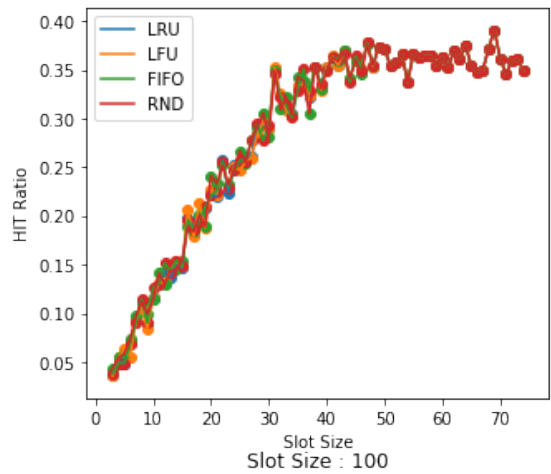
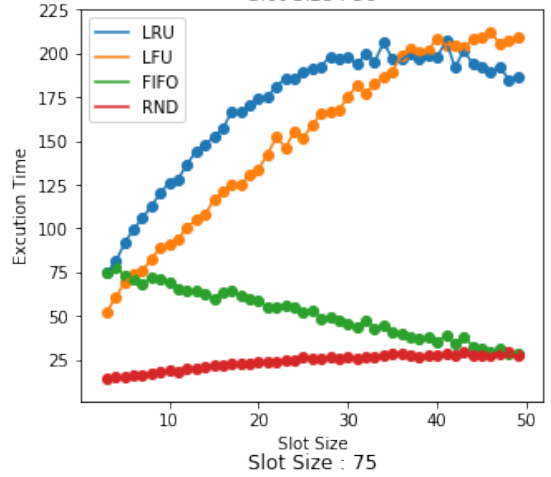
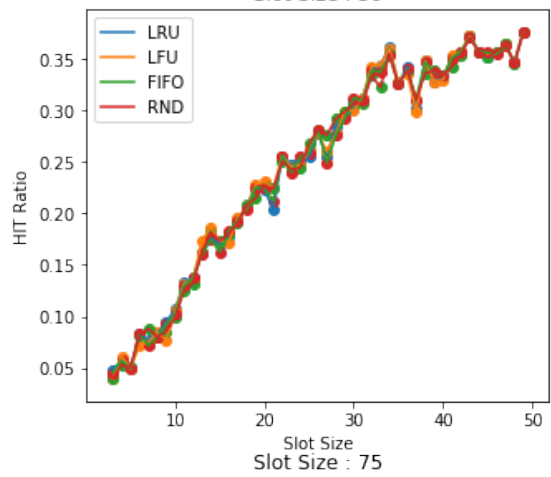
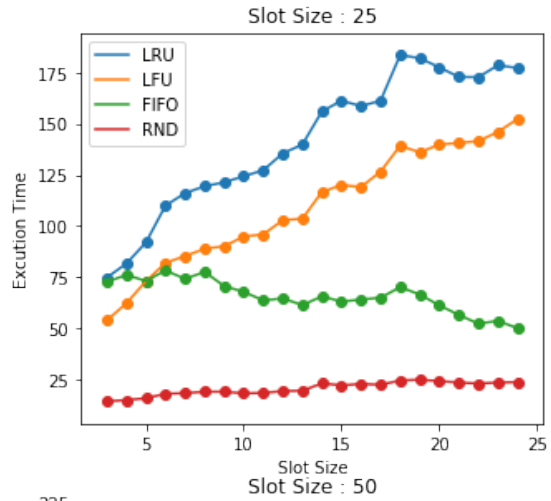
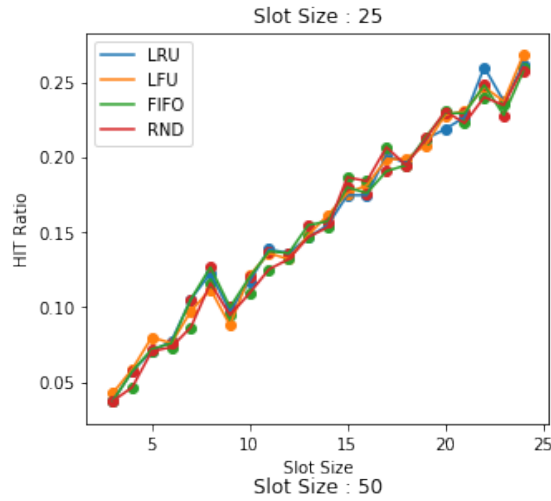
2.3.3 난수의 최대 수가 75일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [112]: visualize(input_size = 75, input_max_size = 75)
```

가능한 입력 수가 75이며 난수의 최대 수가 75일 때

```
Out [112]:
```

	HIT	TIME
LRU / 25	0.16	142.08
LFU / 25	0.16	109.60
RND / 25	0.15	20.46
FIFO / 25	0.16	65.79
LRU / 50	0.25	169.64
LFU / 50	0.25	151.71
RND / 50	0.25	23.86
FIFO / 50	0.25	51.51
LRU / 75	0.29	177.50
LFU / 75	0.28	175.35
RND / 75	0.29	25.66
FIFO / 75	0.29	43.96
LRU / 100	0.31	174.73
LFU / 100	0.31	183.65
RND / 100	0.31	25.39
FIFO / 100	0.31	39.01



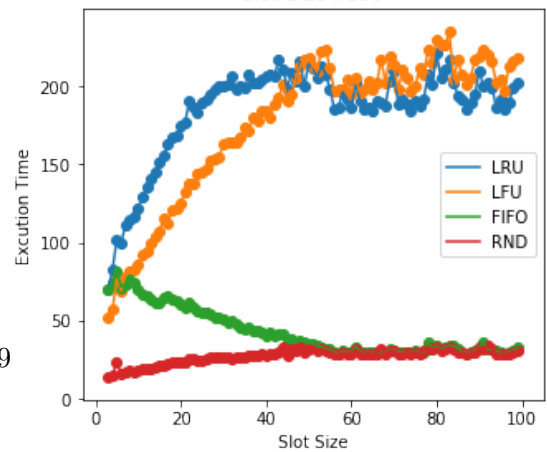
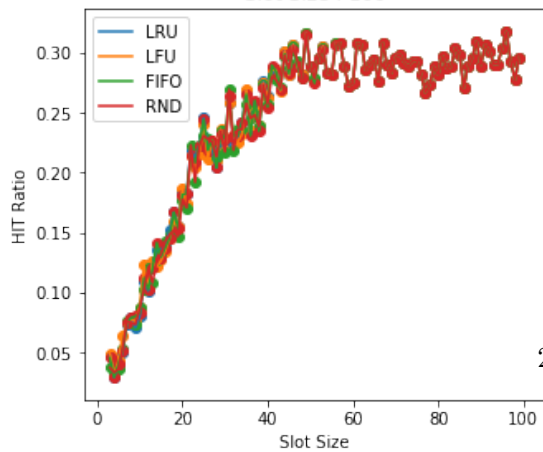
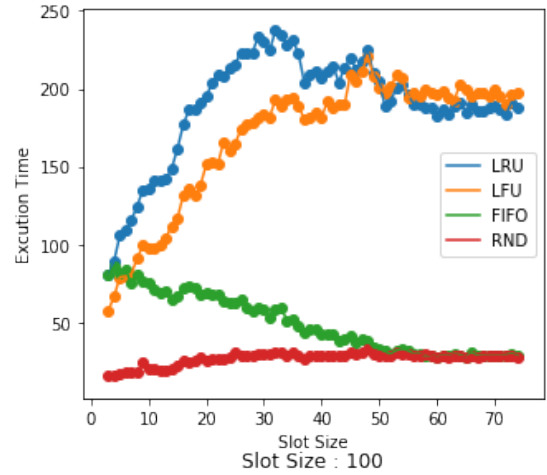
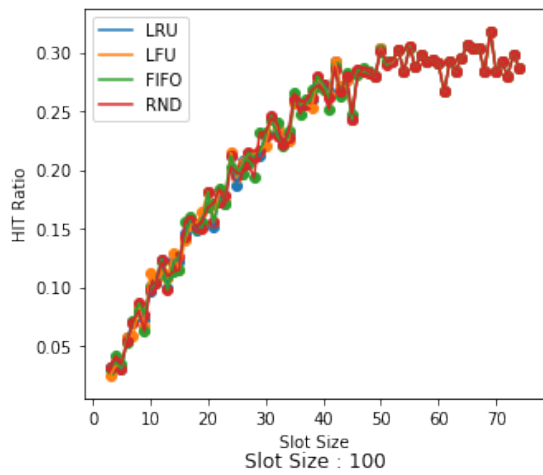
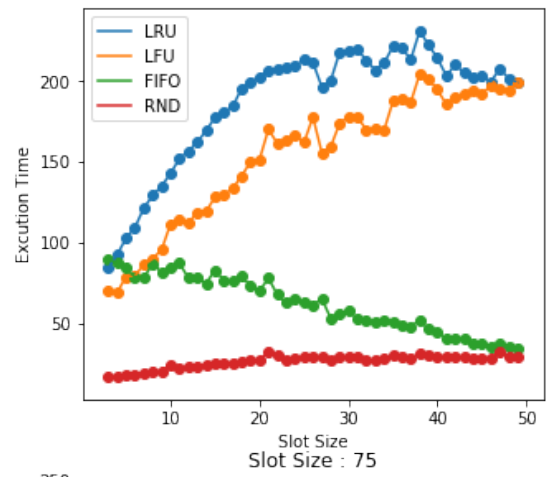
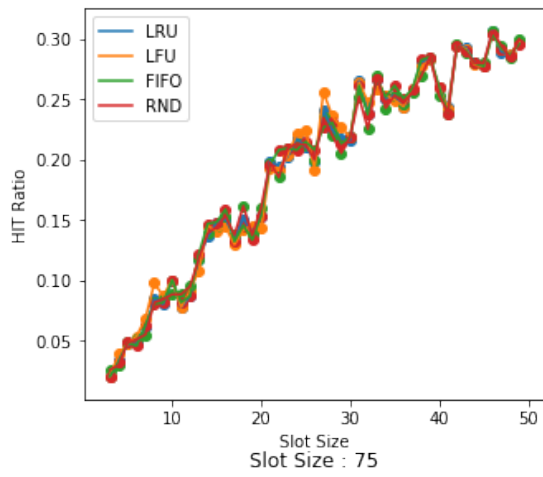
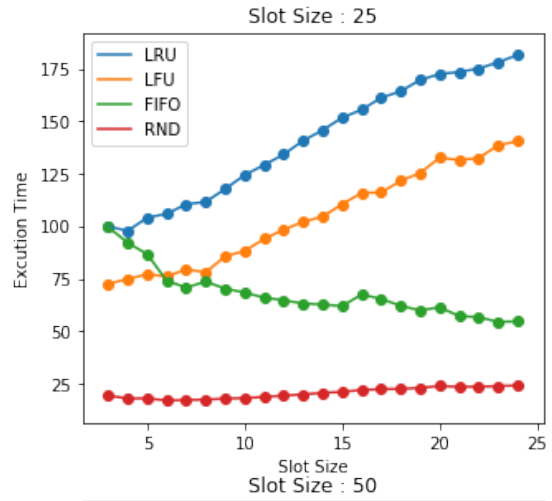
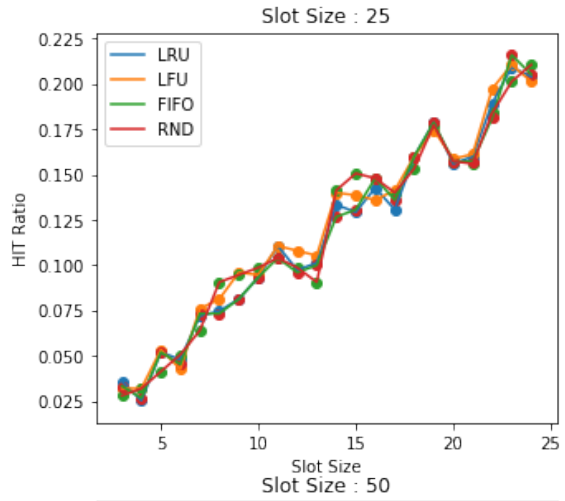
2.3.4 난수의 최대 수가 100일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [113]: visualize(input_size = 75, input_max_size = 100)
```

가능한 입력 수가 75이며 난수의 최대 수가 100일 때

```
Out[113]:
```

	HIT	TIME
LRU / 25	0.12	141.15
LFU / 25	0.12	104.38
RND / 25	0.12	20.52
FIFO / 25	0.12	67.88
LRU / 50	0.19	186.78
LFU / 50	0.19	153.74
RND / 50	0.19	26.56
FIFO / 50	0.19	61.91
LRU / 75	0.22	189.46
LFU / 75	0.22	169.45
RND / 75	0.22	27.55
FIFO / 75	0.22	49.94
LRU / 100	0.24	185.48
LFU / 100	0.24	177.04
RND / 100	0.24	27.52
FIFO / 100	0.24	42.76



2.4 가능한 입력 수가 100일 때

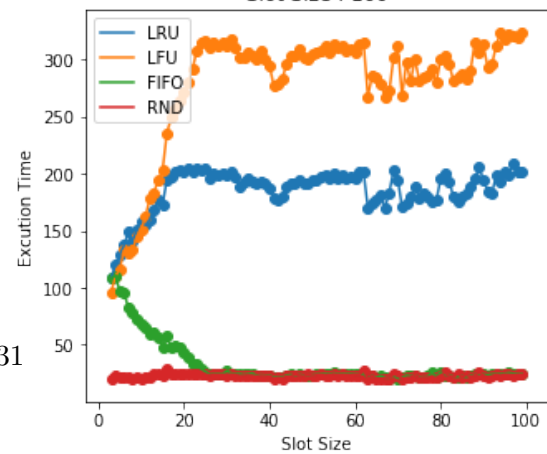
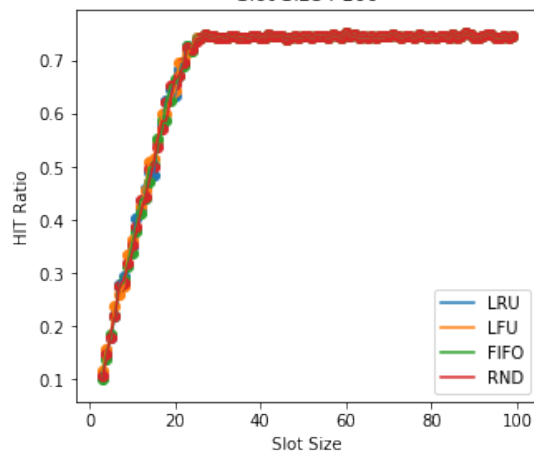
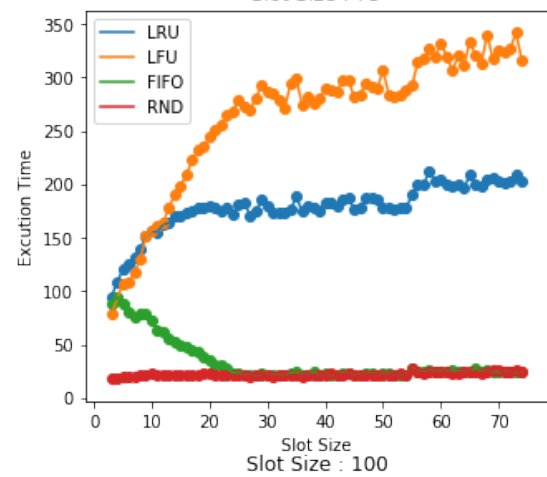
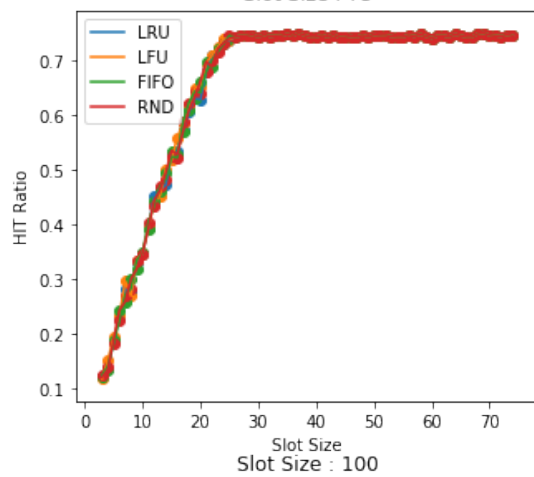
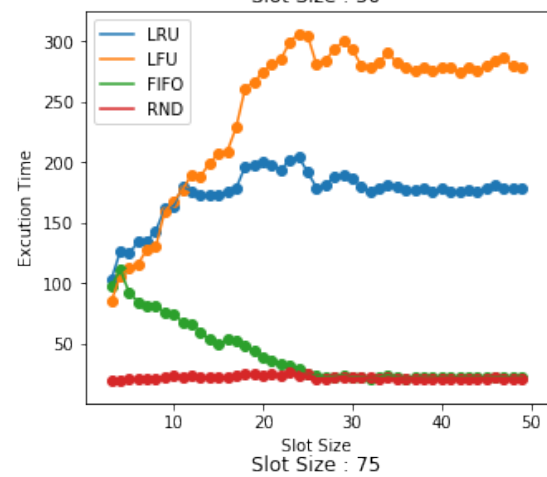
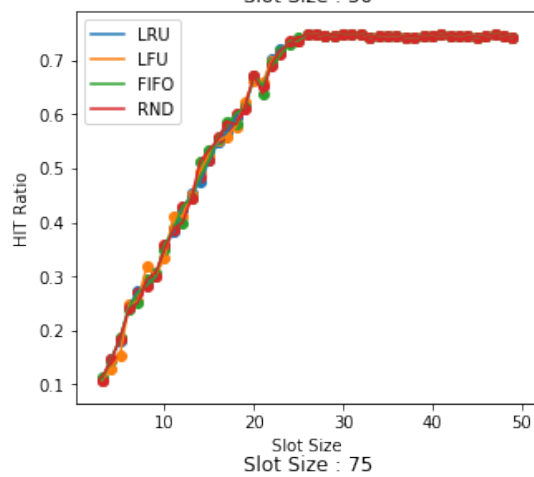
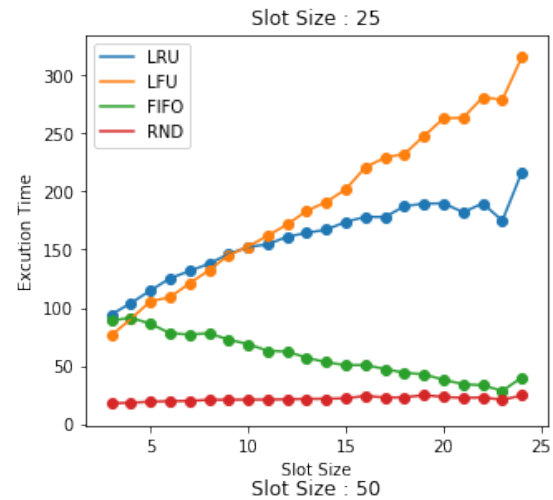
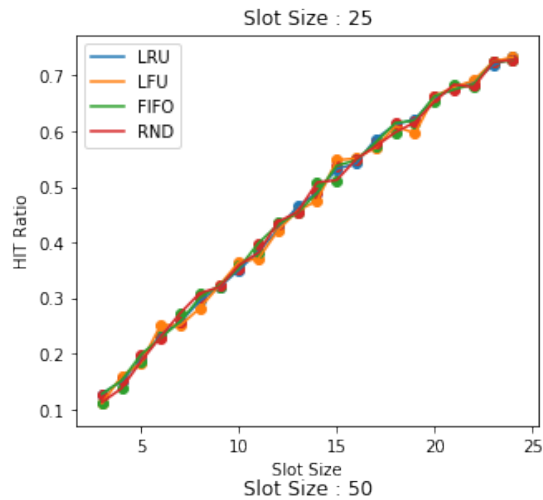
2.4.1 난수의 최대 수가 25일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [114]: visualize(input_size = 100, input_max_size = 25)
```

가능한 입력 수가 100이며 난수의 최대 수가 25일 때

```
Out [114]:
```

	HIT	TIME
LRU / 25	0.46	159.62
LFU / 25	0.46	189.61
RND / 25	0.46	21.55
FIFO / 25	0.46	58.44
LRU / 50	0.61	174.57
LFU / 50	0.61	243.44
RND / 50	0.61	22.13
FIFO / 50	0.61	40.88
LRU / 75	0.66	178.59
LFU / 75	0.66	263.87
RND / 75	0.66	22.32
FIFO / 75	0.66	33.78
LRU / 100	0.68	186.19
LFU / 100	0.68	278.03
RND / 100	0.68	23.03
FIFO / 100	0.68	32.52



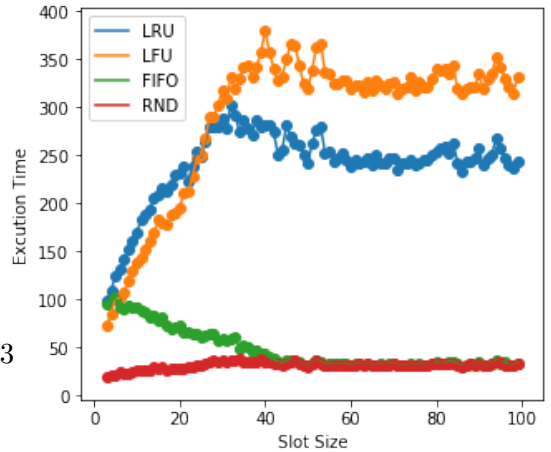
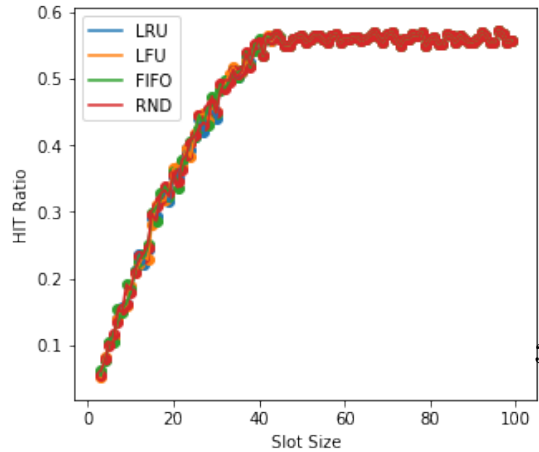
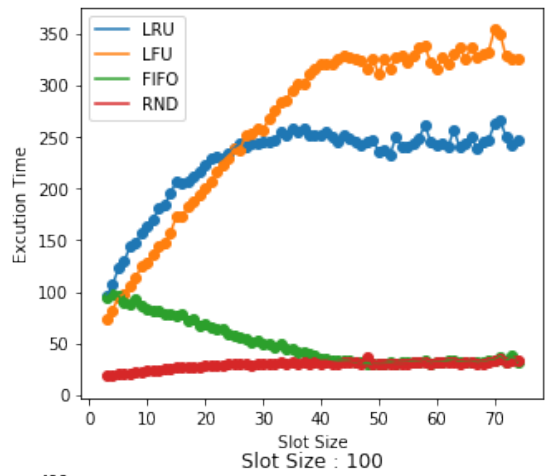
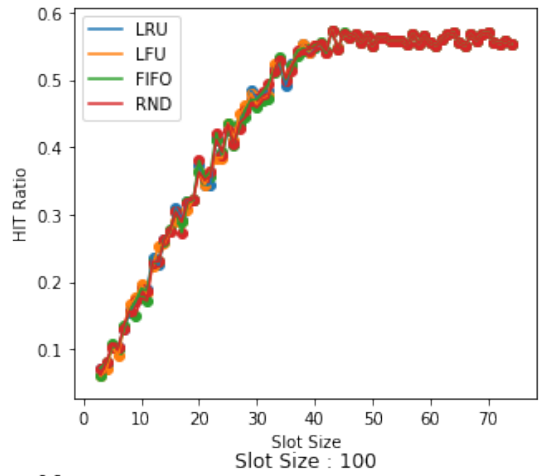
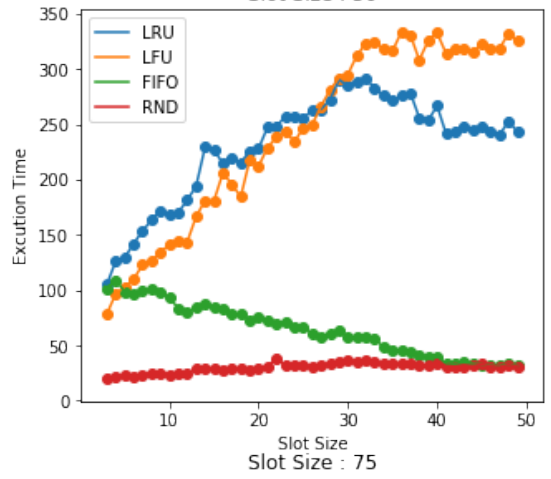
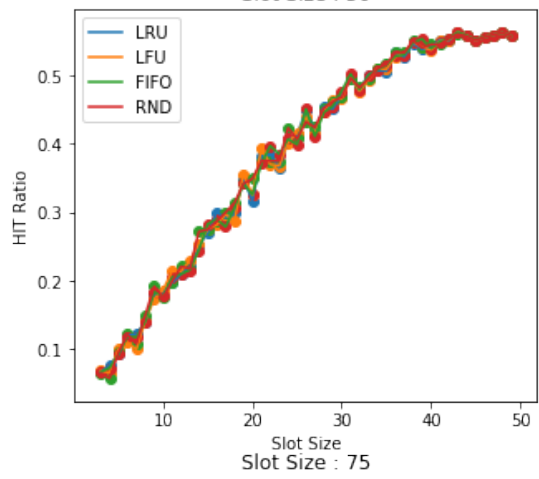
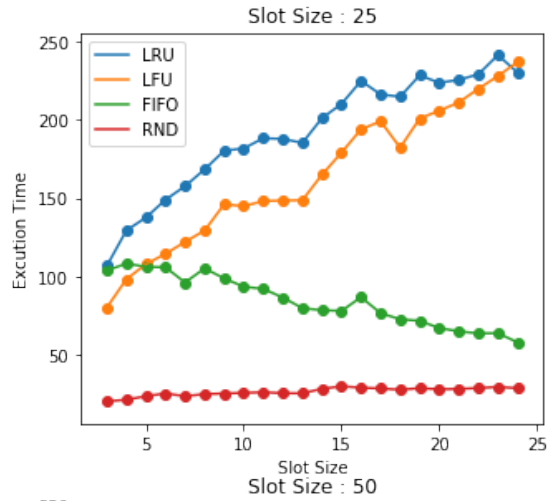
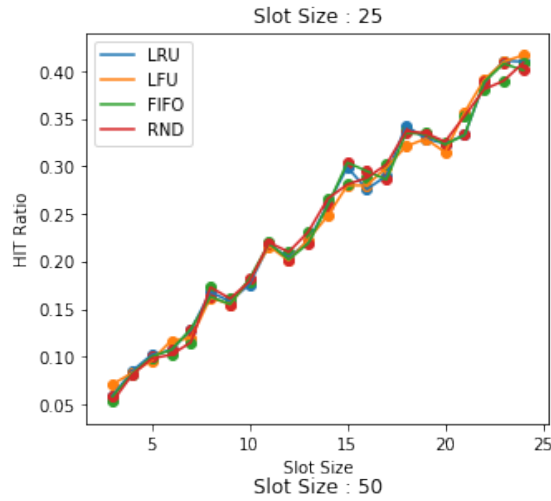
2.4.2 난수의 최대 수가 50일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [115]: visualize(input_size = 100, input_max_size = 50)
```

가능한 입력 수가 100이며 난수의 최대 수가 50일 때

```
Out[115]:
```

	HIT	TIME
LRU / 25	0.24	191.85
LFU / 25	0.24	164.25
RND / 25	0.24	26.57
FIFO / 25	0.24	84.57
LRU / 50	0.38	230.74
LFU / 50	0.38	243.06
RND / 50	0.39	29.86
FIFO / 50	0.38	64.21
LRU / 75	0.45	227.04
LFU / 75	0.45	263.49
RND / 75	0.45	29.30
FIFO / 75	0.45	49.96
LRU / 100	0.48	240.80
LFU / 100	0.48	289.49
RND / 100	0.48	31.03
FIFO / 100	0.48	47.44



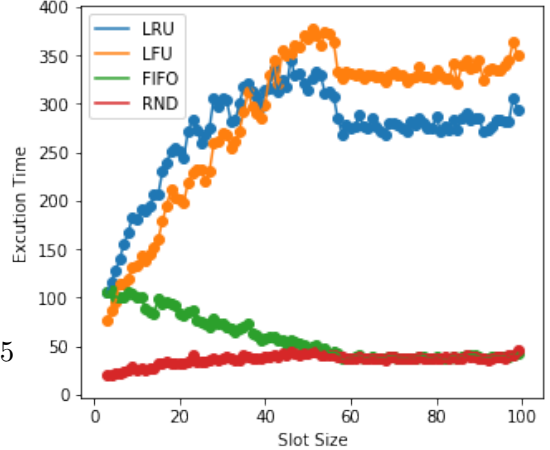
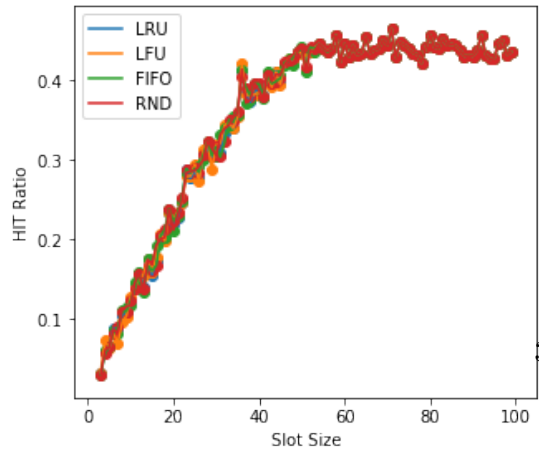
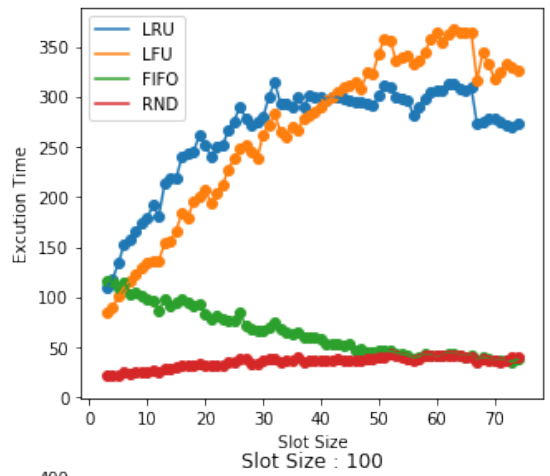
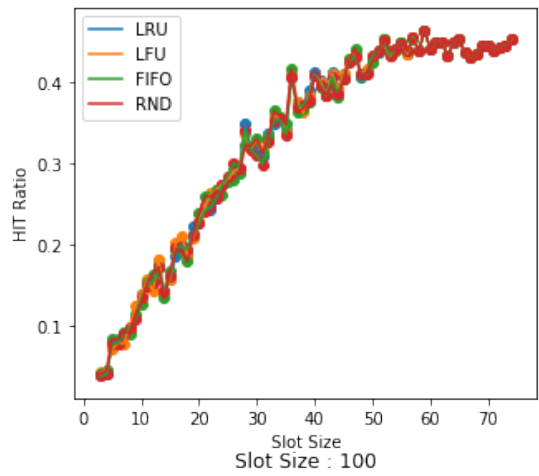
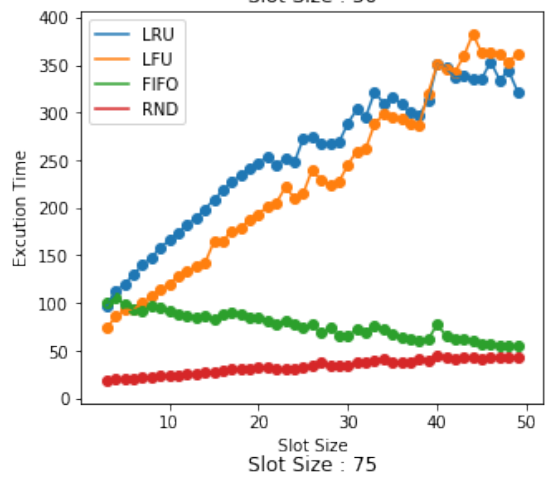
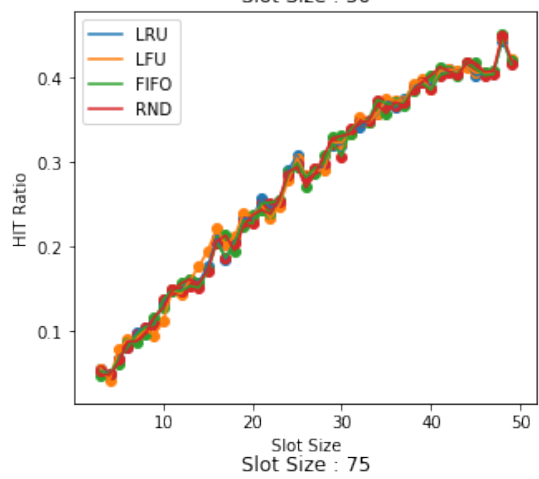
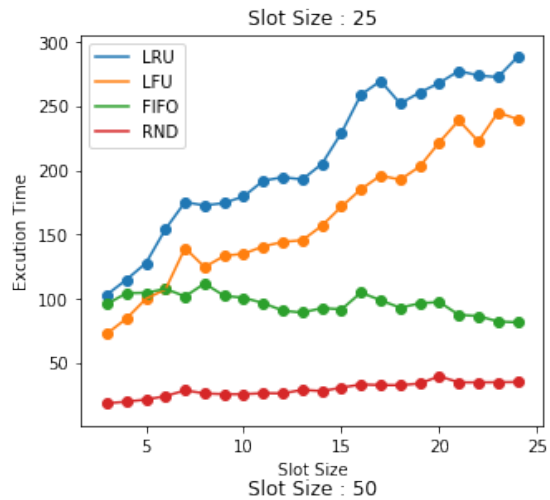
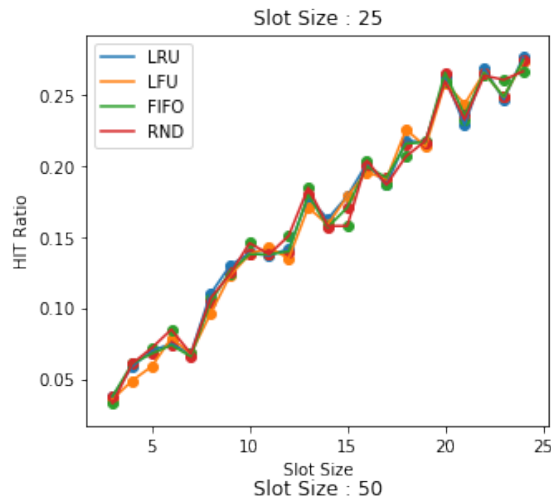
2.4.3 난수의 최대 수가 75일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [116]: visualize(input_size = 100, input_max_size = 75)
```

가능한 입력 수가 100이며 난수의 최대 수가 75일 때

```
Out[116]:
```

	HIT	TIME
LRU / 25	0.16	210.65
LFU / 25	0.16	163.61
RND / 25	0.16	29.27
FIFO / 25	0.16	96.28
LRU / 50	0.27	255.20
LFU / 50	0.27	229.60
RND / 50	0.27	33.18
FIFO / 50	0.27	76.47
LRU / 75	0.33	265.75
LFU / 75	0.33	264.98
RND / 75	0.33	35.37
FIFO / 75	0.33	64.85
LRU / 100	0.36	272.33
LFU / 100	0.36	286.84
RND / 100	0.36	36.28
FIFO / 100	0.36	58.20



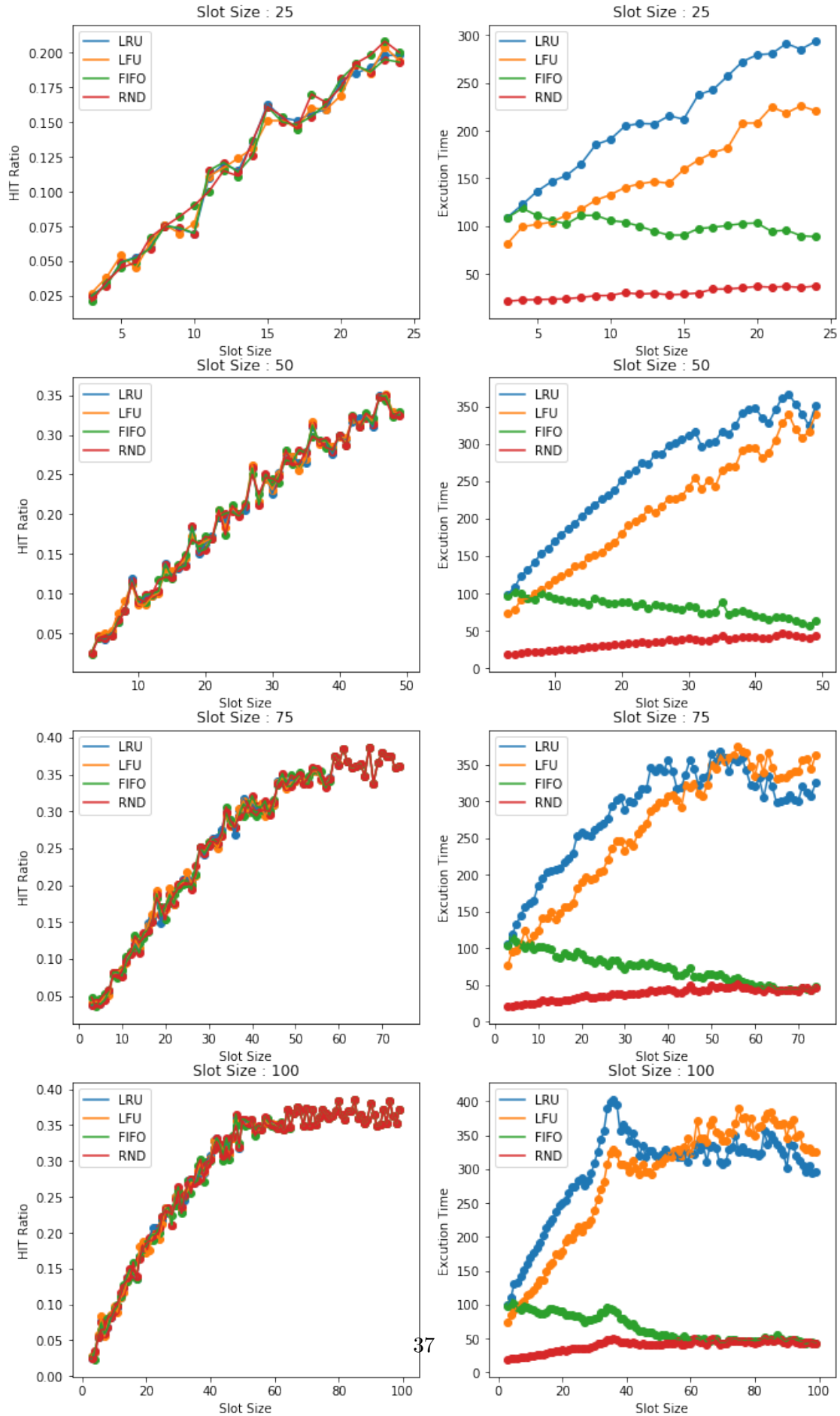
2.4.4 난수의 최대 수가 100일 때 슬롯 사이즈를 25,50,75,100으로 주어 적중률과 수행시간 비교

```
In [117]: visualize(input_size = 100, input_max_size = 100)
```

가능한 입력 수가 100이며 난수의 최대 수가 100일 때

```
Out[117]:
```

	HIT	TIME
LRU / 25	0.12	213.41
LFU / 25	0.12	156.56
RND / 25	0.12	29.66
FIFO / 25	0.12	101.10
LRU / 50	0.21	263.62
LFU / 50	0.21	210.66
RND / 50	0.21	33.99
FIFO / 50	0.21	81.46
LRU / 75	0.26	286.74
LFU / 75	0.26	263.12
RND / 75	0.26	38.38
FIFO / 75	0.26	72.37
LRU / 100	0.29	297.98
LFU / 100	0.29	284.86
RND / 100	0.29	40.10
FIFO / 100	0.29	65.21



```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```