과제

장유선

2023.08.17

1. 문제 정의

1에서 1,000,000까지의 원소를 지닌 집합이 존재한다

이 집합에서 임의로 700,000개의 원소를 지닌 부분 집합을 2개 만들고, 해당 부분 집합에 대하여

* 합집합
* 교집합
* 차집합

의 개수를 출력한다

그리고 해당 작업이 수행되는 시간을 출력한다

★ random.sample() 은 사용하지 말고 만들어 주세요

2. Python Code Hard Copy

|  |
| --- |
| import time  import random  #전체집합  full\_set = list(range(1, 1000001))  #부분집합 생성 함수  def generate\_random\_subset(count, source\_set):      subset = set()      while len(subset) < count:          index = random.randint(0, len(source\_set) - 1)          element = source\_set.pop(index)          subset.add(element)      return subset  # 부분집합  subset1 = generate\_random\_subset(700000, full\_set.copy())  subset2 = generate\_random\_subset(700000, full\_set.copy())  start\_time = time.time()  # 합집합, 교집합, 차집합 계산  union\_set = subset1.union(subset2)  intersection\_set = subset1.intersection(subset2)  difference\_set = subset1.difference(subset2)  end\_time = time.time()  print("합집합 개수:", len(union\_set))  print("교집합 개수:", len(intersection\_set))  print("차집합 개수:", len(difference\_set))  print("실행 시간:", end\_time - start\_time, "초") |

4. Code 설명

import time

import random

# 전체집합

full\_set = list(range(1, 1000001))

# 부분집합 생성 함수 정의

def generate\_random\_subset(count, source\_set):

    subset = set()  # 빈 집합

    while len(subset) < count:  # 원하는 개수만큼 원소가 추가될 때까지 반복

        index = random.randint(0, len(source\_set) - 1)  # 무작위 인덱스를 선택

        element = source\_set.pop(index)  # 선택한 인덱스의 원소를 전체집합에서 뽑는다

        subset.add(element)  # 부분집합에 추가

    return subset

# 부분집합 생성

subset1 = generate\_random\_subset(700000, full\_set.copy())

subset2 = generate\_random\_subset(700000, full\_set.copy())

start\_time = time.time()  # 시작 시간 기록

# 합집합, 교집합, 차집합 계산

union\_set = subset1.union(subset2)  # 두 부분집합의 합집합

intersection\_set = subset1.intersection(subset2)  # 두 부분집합의 교집합

difference\_set = subset1.difference(subset2)  # 첫 번째 부분집합에서 두 번째 부분집합을 뺀 차집합

end\_time = time.time()  # 종료 시간 기록

# 결과 출력

print("합집합 개수:", len(union\_set))

print("교집합 개수:", len(intersection\_set))

print("차집합 개수:", len(difference\_set))

print("실행 시간:", end\_time - start\_time, "초")

4. 결과

|  |
| --- |
| 합집합 개수: 910011  교집합 개수: 489989  차집합 개수: 210011  실행 시간: 0.1256418228149414 초 |

5. 결과 화면

