

기본 안내

- 파일 - 다운로드 - MS Word (.DOCX)로 다운받거나, 파일 - 사본 만들기로 자신의 구글 드라이브에 복사한다.
- 각 문제의 제출 방법 안내대로 과제를 푼 내역을 붙여넣은 후, PDF 파일로 변환하여 이 루리에 제출한다.

(40점) 행렬 곱셈을 수행하는 단순 알고리즘과 쉬트라센 알고리즘을 구현하라

설명

구현해야 할 코드는 다음과 같다

(10점) 단순 알고리즘

- 입력
 - A: 정수형(int) 2차원 배열
 - B: 정수형(int) 2차원 배열
- 출력:
 - A와 B의 행렬곱 AB
- 예:
 - `mat_mult(A: array, B: array): array`

(10점) 쉬트라센 알고리즘

- 입력
 - A: 정수형(int) 2차원 배열
 - B: 정수형(int) 2차원 배열
- 출력:
 - A와 B의 행렬곱 AB
- 예:
 - `mat_strassen(A: array, B: array): array`

(5점) 두 행렬의 같음을 판단하는 함수

- 입력
 - A: 정수형(int) 2차원 배열
 - B: 정수형(int) 2차원 배열
- 출력:
 - A와 B가 같을 시 **True** 아닐시 **False** (bool)
- 예:
 - `is_equal_mat(A: array, B: array): bool`

(15점) 시뮬레이션 및 결과

- 단위 시뮬레이션
 - **-100 이상 100이하의 랜덤한 정수로 구성된 128 x 128 행렬 A, B를 생성한다.**
 - 단순 알고리즘`mat_mult`으로 행렬 A, B를 곱한 결과를 배열 C에 저장하고, 걸린 시간을 측정한다.
 - 슈트라스센 알고리즘`mat_strassen`으로 행렬 A, B를 곱한 결과를 배열 D에 저장하고, 걸린 시간을 측정한다.
 - C와 D가 같은지`is_equal_mat` 확인한다. 같지 않을 경우 구현이 잘못된 것이다.
- 단위 시뮬레이션을 1,000번 반복해서 (1) 단순 알고리즘이 행렬곱을 할 때 **평균적으로 걸린 시간**, (2) 슈트라스센 알고리즘이 행렬곱을 할때 **평균적으로 걸린 시간**, (3) 1,000회 시행 동안 두 알고리즘이 같은 출력을 낸 횟수를 출력하라
 - 예.

```
Simple: 10.223s
Strassen: 3.532s
N. Equals: 1,000 times
```

제출 방법

구현한 코드와 출력 결과를 캡처해서 아래에 붙여넣으시오

(10점) 단순 알고리즘

(10점) 쉬트라센 알고리즘

(5점) 두 행렬의 같음을 판단하는 함수

(15점) 시뮬레이션 및 결과

(30점) 이항 계수를 계산하는 동적 계획법 알고리즘을 구현하라

설명

구현해야 할 코드는 다음과 같다

(10점) 기본 동적 계획법 알고리즘

- 입력
 - n : 정수형(int)
 - k : 정수형(int)
- 출력:
 - $C(n, k)$
- 제약 사항
 - 작은 입력 사례에 대한 결과를 저장하는 2차원 배열 B 를 활용
- 예:
 - `bin_coeff(n: int, k: int): int`

(10점) 개선된 동적 계획법 알고리즘

- 입력
 - n : 정수형(int)
 - k : 정수형(int)
- 출력:
 - $C(n, k)$
- 제약 사항
 - 작은 입력 사례에 대한 결과를 저장하는 1차원 배열 B 를 활용
- 예:
 - `bin_coeff_imp(n: int, k: int): int`

(10점) 시뮬레이션 및 결과

- 단위 시뮬레이션
 - 100이하의 랜덤한 정수 n, k 를 뽑는다. 단, k 는 n 보다 클 수 없다.
 - 기본 동적 계획법 알고리즘 `bin_coeff`로 이항계수를 계산한다.
 - 개선된 동적 계획법 알고리즘 `bin_coeff_imp`로 이항계수를 계산한다.
 - 두 값이 같은지 확인한다. 같지 않을 경우 구현이 잘못된 것이다.

- 단위 시뮬레이션을 1,000번 반복해서 (1) 기본 동적 계획법 알고리즘이 이항계수를 구할 때 **평균적으로 걸린 시간**, (2) 개선된 동적 계획법 알고리즘이 이항계수를 구할 때 **평균적으로 걸린 시간**, (3) **1,000회 시행 동안 두 알고리즘이 같은 출력을 낸 횟수**를 출력하라
 - 예.

```
bin_coeff: 10.223s
bin_coeff_imp: 3.532s
N. Equals: 1,000 times
```

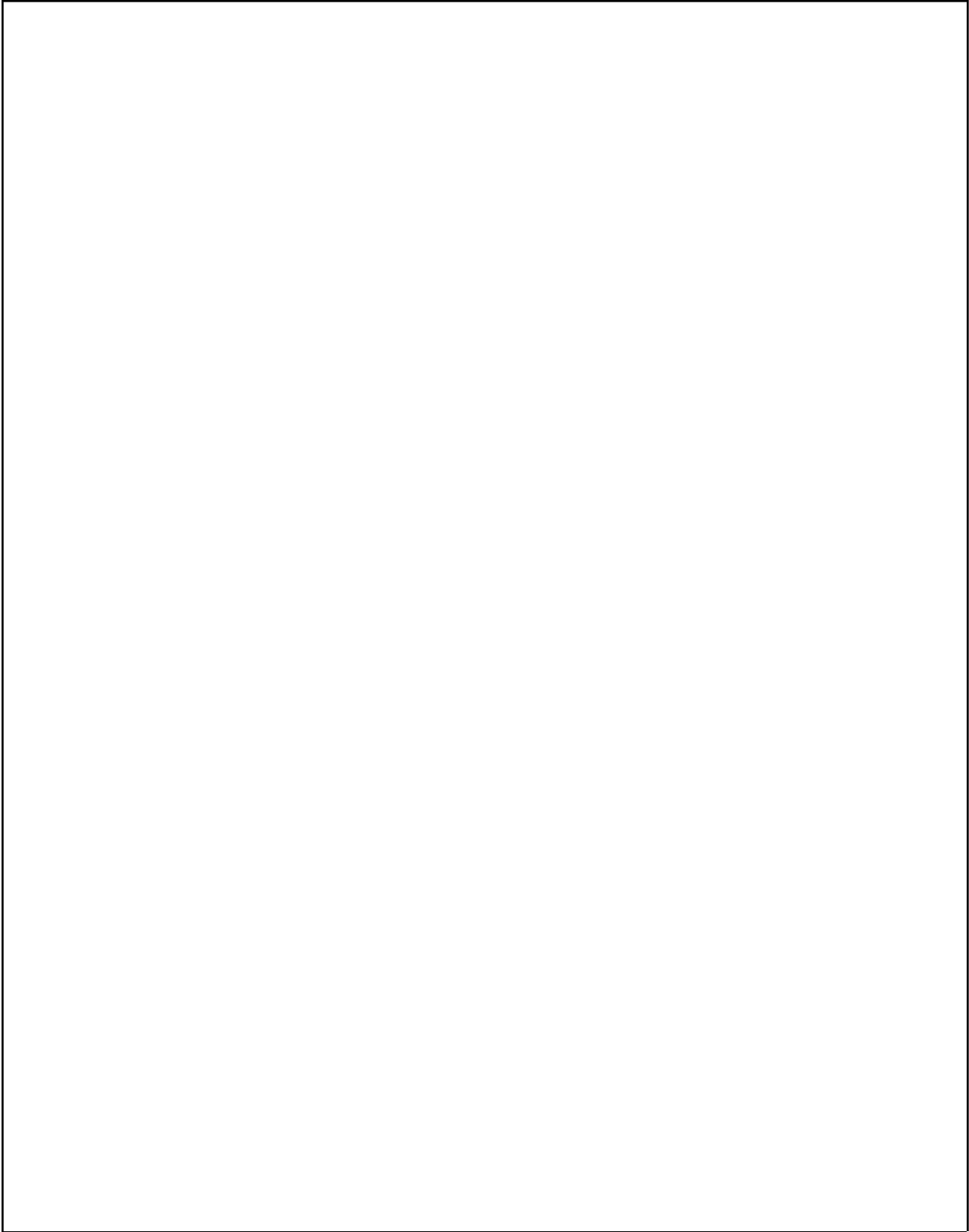
제출 방법

구현한 코드와 출력 결과를 캡처해서 아래에 붙여넣으시오

(10점) 기본 동적 계획법 알고리즘

(10점) 개선된 동적 계획법 알고리즘

(10점) 시뮬레이션 및 결과





(30점) 다음 점화식의 일반해를 구하시오

※ θ 를 구하는 것이 아님에 유의

제출 방법

각 문제를 종이, 수식 에디터(예. 한글, 파워포인트, 기타 온라인 수식 에디터 등) 등을 통해 푼 후 답과 풀이 과정을 화면 캡처해서 붙여넣으시오.

1. (10점) $T(n) = 5T(n - 1) + 2n, T(0) = 0$

답	
풀이과정	

2. (10점) $T(n) = 2T(\frac{n}{3}) + n^2$, $T(1) = 1, n = 3^k$ for $k \geq 0$

답	
풀이과정	

3. (10점) $T(n) = T(n - 1) + T(n - 2)$, $T(0) = 0$, $T(1) = 1$

답	
풀이과정	