2주차 과제1 #3

2022145079 임혜린

본 과제는 Python, VS Code를 사용하였음을 밝힙니다.

공통 조건 (Stokes second problem)

텍스트, 폰트, 편지, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

1-1

문제

텍스트, 폰트, 편지, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

풀이

텍스트, 친필, 폰트, 번호이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

1-2

문제텍스트, 폰트, 편지, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

풀이

텍스트, 친필, 폰트, 번호이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

2-1

텍스트, 폰트, 편지, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

FTCS는 시간 방향 전진(Forward Time), 공간 방향 중앙차분(Central Space) 방식으로 진행하는 근사 방법을 뜻한다. 1-1의 Stokes second problem의 지배 방정식에 FTCS를 적용하면

폰트, 라인, 도표, 텍스트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.



이와 같으며 안정성 조건은 이다.

텍스트, 스크린샷, 멀티미디어 소프트웨어, 소프트웨어이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

주어진 조건에 따라 v=1이며 dy=0.1로 놓았으므로 dt≤0.005이다. 조건을 만족하는 dt=2π/2000만큼 t를 증가해가며 t에서의 속도 리스트 u\_n을 통해 u\_n1[j]를 도출하는 식을 만든 후 해당 t가 그래프를 그리고자 하는 nt에 해당하는 경우 u\_profile 리스트에 추가하는 형식을 사용하였다.

u\_profile[0]~ u\_profile[4]는 FTCS scheme at nt=0, π/2, π, 3π/2, 2π 그래프, u\_profile[5]~ u\_profile[9]는 준정상 상태에서의 속도, 즉 FTCS scheme at quasti-steady state (at nt=10π, 10.5π, 11π, 11.5π, 12π) 그래프를 나타내는 데 사용되었다.

FTCS scheme at nt=0, π/2, π, 3π/2, 2π의 그래프 코드와 결과

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 도표, 라인, 그래프이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

FTCS scheme at quasi-steady state (at nt=10π, 10.5π, 11π, 11.5π, 12π)의 그래프 코드와 결과

텍스트, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.텍스트, 라인, 도표, 그래프이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

FTCS scheme at nt=0, π/2, π, 3π/2, 2π, 10π, 10.5π, 11π, 11.5π, 12π의 그래프 코드와 결과

텍스트, 스크린샷이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 도표, 라인, 그래프이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.(두 그래프 함께 나타내기)

실선이 준정상 상태에서의 그래프이다.

2-2

텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

Crank-Nicolson scheme은 시간 미분을 전 시간 단계와 다음 시간 단계의 중앙 평균으로 근사한다. 시간 방향 전진이었던 FTCS와 달리 n번째와 n+1번째 시점의 함수값을 모두 포함하여 미분값을 구하기 때문에 n\_y\*n\_y 벡터와 np.linalg.solve를 이용하여 코드를 구성하였다.

Stokes second condition의 지배방정식에 적용하면 아래와 같다.

폰트, 라인, 친필, 텍스트이(가) 표시된 사진

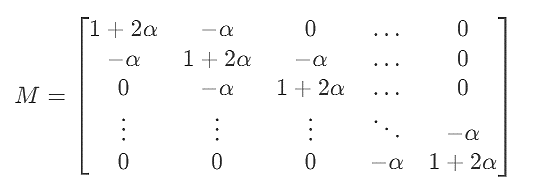
AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 폰트, 친필, 화이트이(가) 표시된 사진

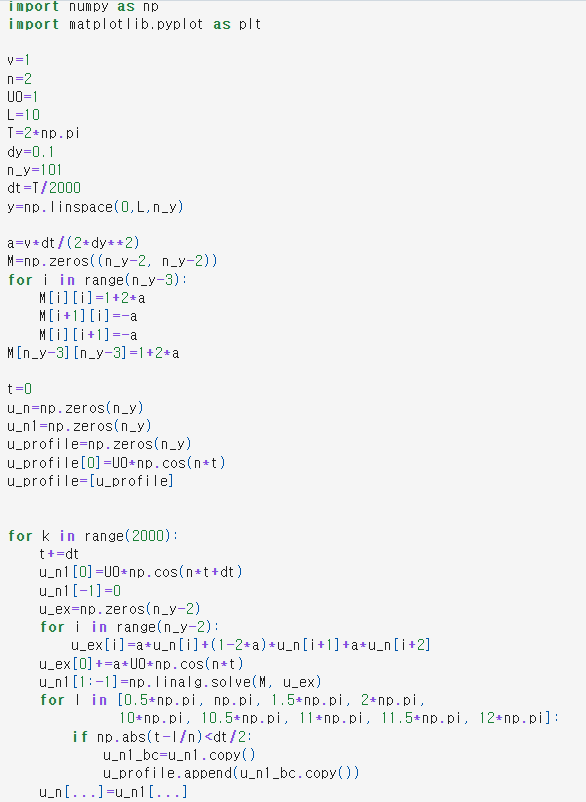
AI 생성 콘텐츠는 정확하지 않을 수 있습니다.텍스트, 폰트, 친필, 화이트이(가) 표시된 사진

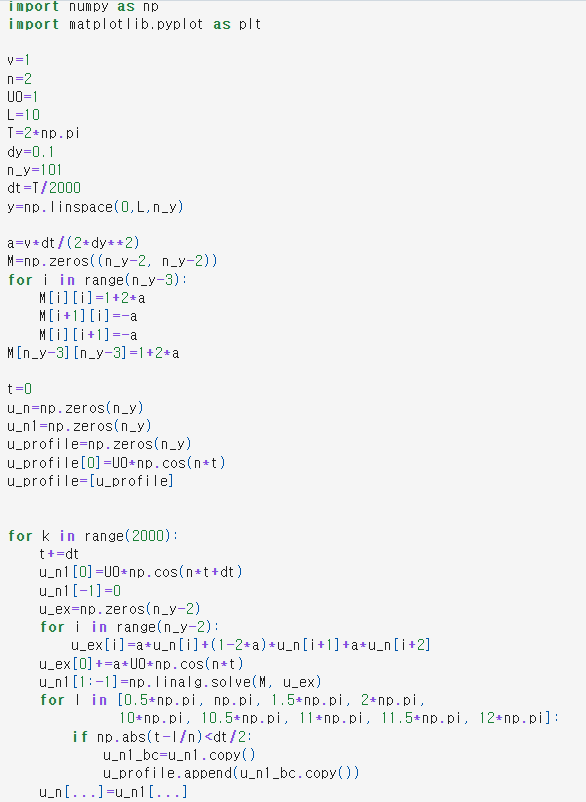
AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

로 잡았을 때, 이므로 이를 만족시키기 위해 n\_y-2\*n\_y-2 벡터 M을 만든 후, 위 식의 우변 값을 y에 따라 저장한 리스트 u\_x와 M을 np.linalg.solve하여 이것을 len이 n\_y인 리스트 u\_n1의 양 끝 값을 제외한 부분에 저장하였다. u\_n1의 양 끝 값은 boundary condition을 만족하기 위해 따로 지정하였다.

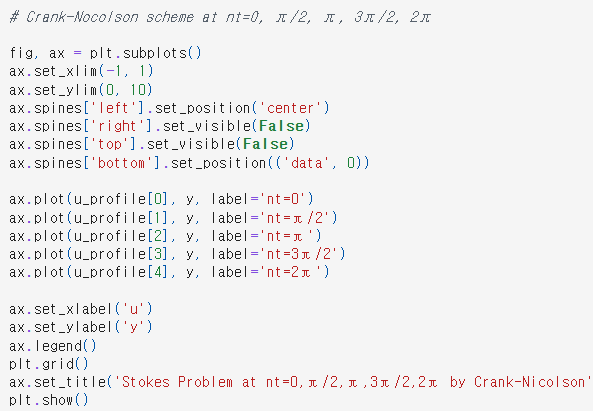


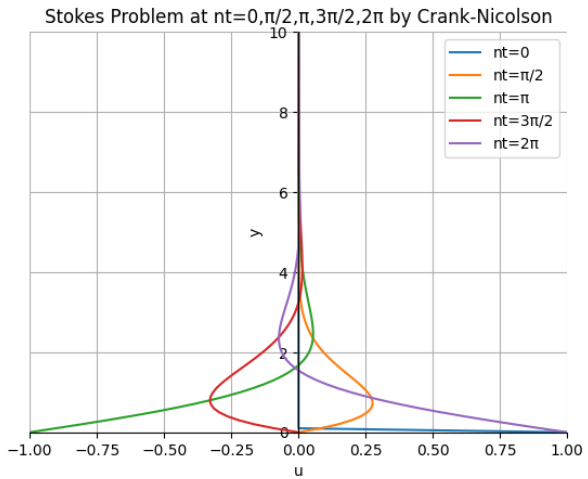
이렇게 저장된 u\_n1은 t가 변함에 따라 리셋되며, 해당 t가 그래프를 그리고자 하는 nt에 해당하는 경우 u\_profile 리스트에 추가되도록 하였다.



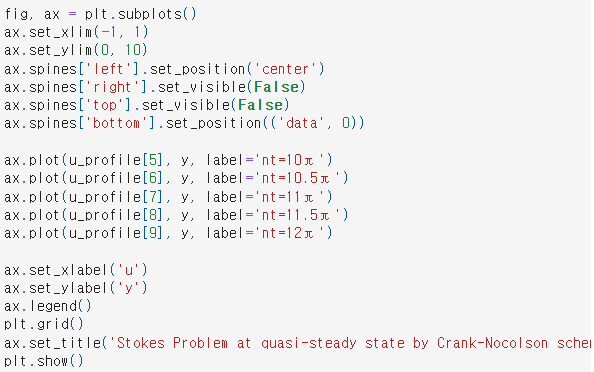


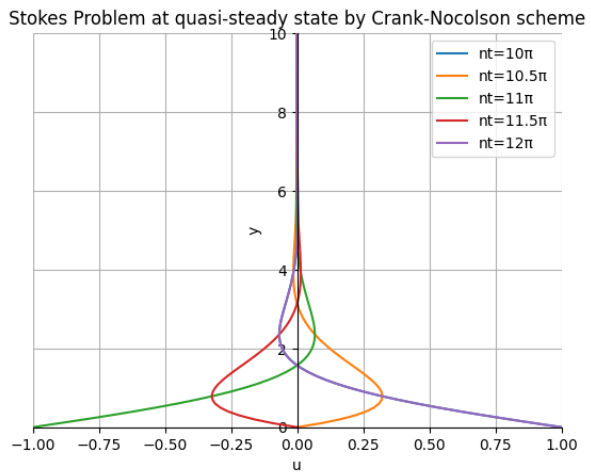
Crank-Nicolson scheme at nt=0, π/2, π, 3π/2, 2π의 그래프 코드와 결과





Crank-Nicolson scheme at quasi-steady state (nt=10π,10.5π,11π,11.5π,12π)의 그래프 코드와 결과

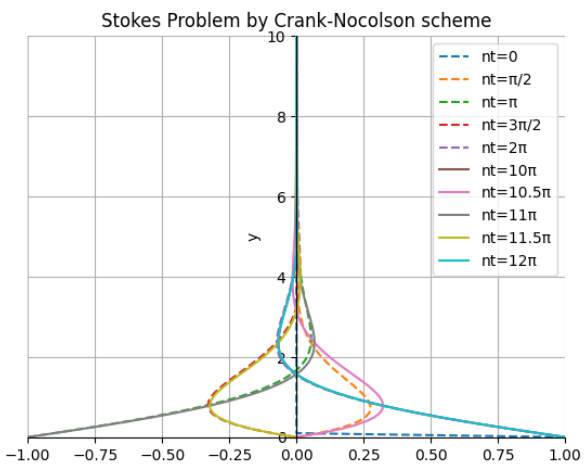




Crank-Nicolson scheme at nt=0, π/2, π, 3π/2, 2π, 10π, 10.5π, 11π, 11.5π, 12π 그래프 코드와 결과





(두 그래프 함께 나타내기)

실선이 준정상 상태에서의 그래프이다.

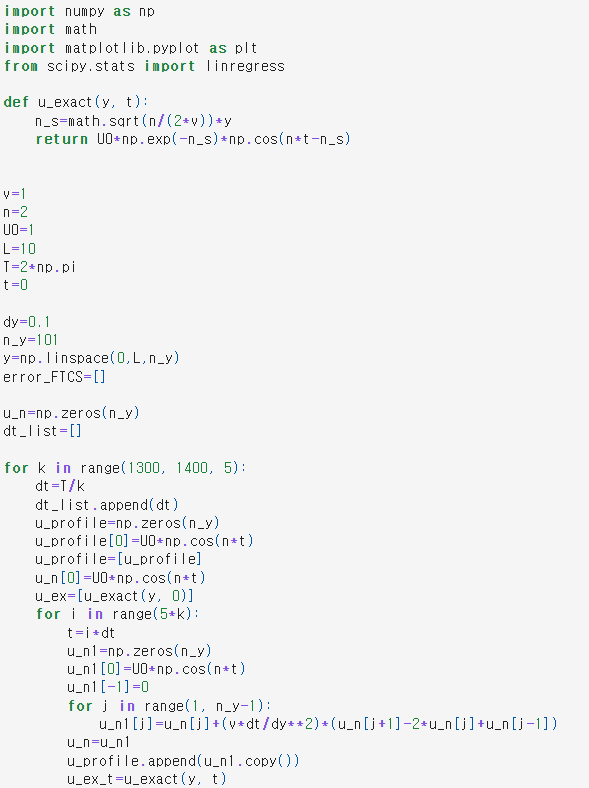
2-3

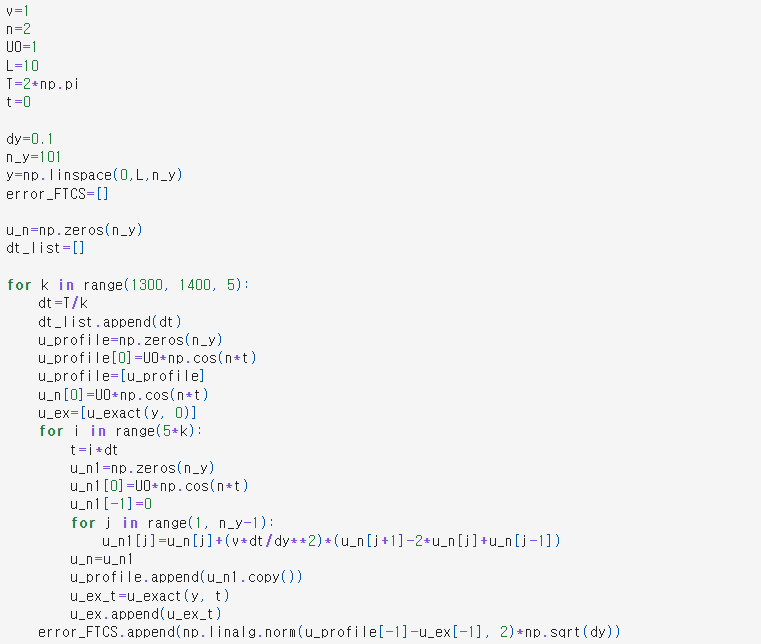
텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

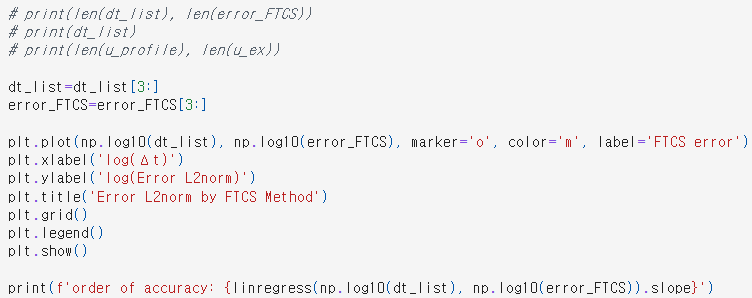
기본 코드는 2-1, 2-2와 같으나 for문을 이용하여 dt를 변화시키며 진행하였다. 동일한 dt번째의 시점을(ex: t=5\*k\*dt) 잡아 그때의 각 y에서의 u를 scheme으로 도출한 값과 exact solution 값의 L2 norm을 구하여 error\_FTCS 또는 error\_CN에 추가하였다. 이렇게 구한 각 dt에서의 error을 모아 log10 그래프를 그린 결과 FTCS는 약 1.56, Crank-Nicolson은 약 1.2의 order of accuracy를 보였다.

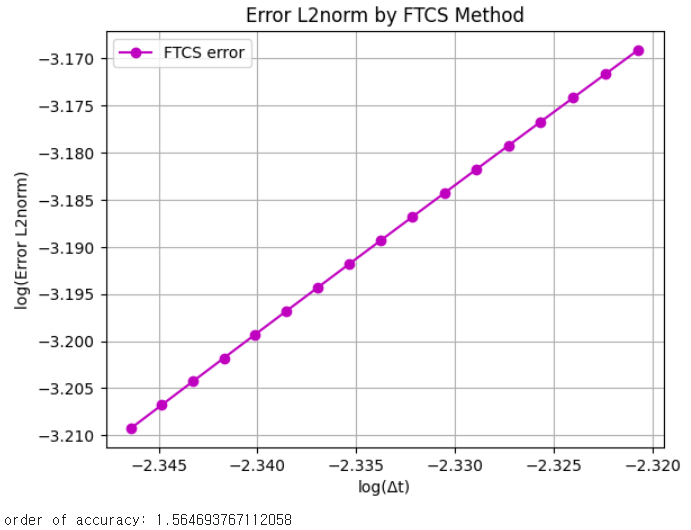
* FTCS Method의 경우



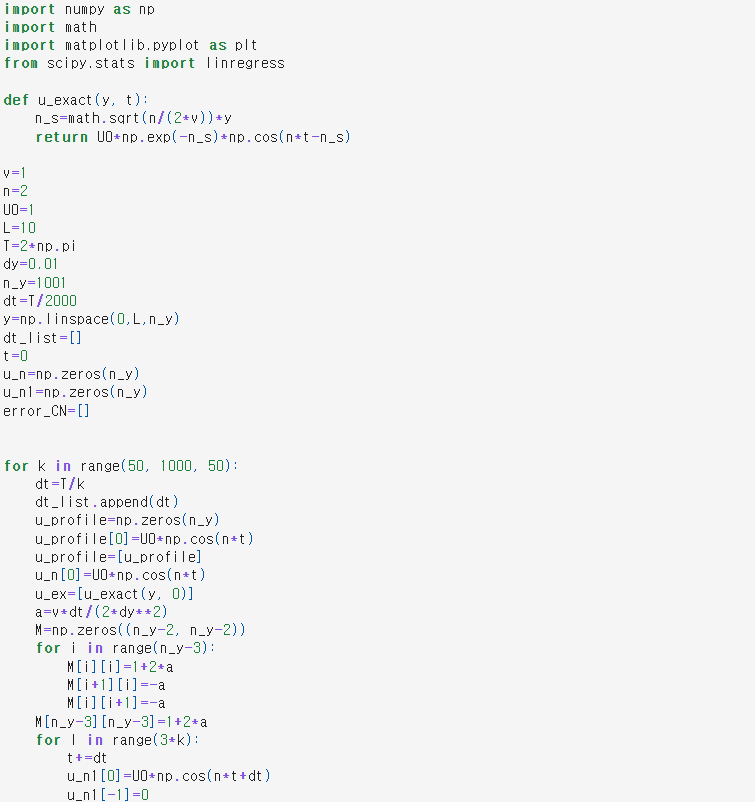


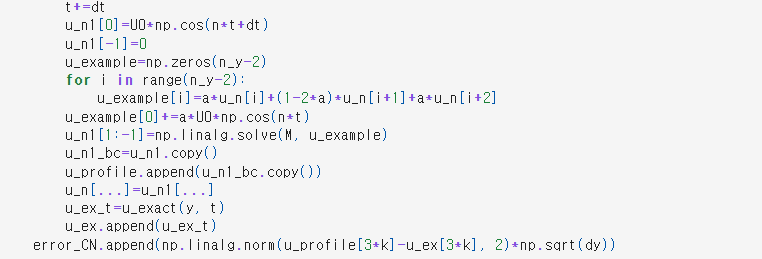
Error L2 norm by FTCS Method의 그래프 코드와 결과



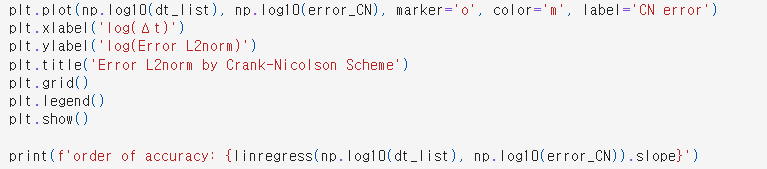
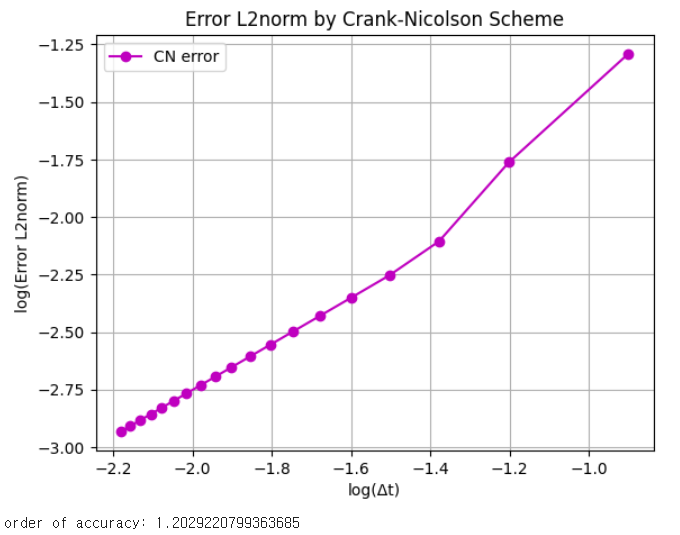
order of accuracy: 1.564693767112058

* Crank-Nicolson scheme의 경우





Error L2 norm by Crank-Nicolson scheme의 그래프 코드와 결과



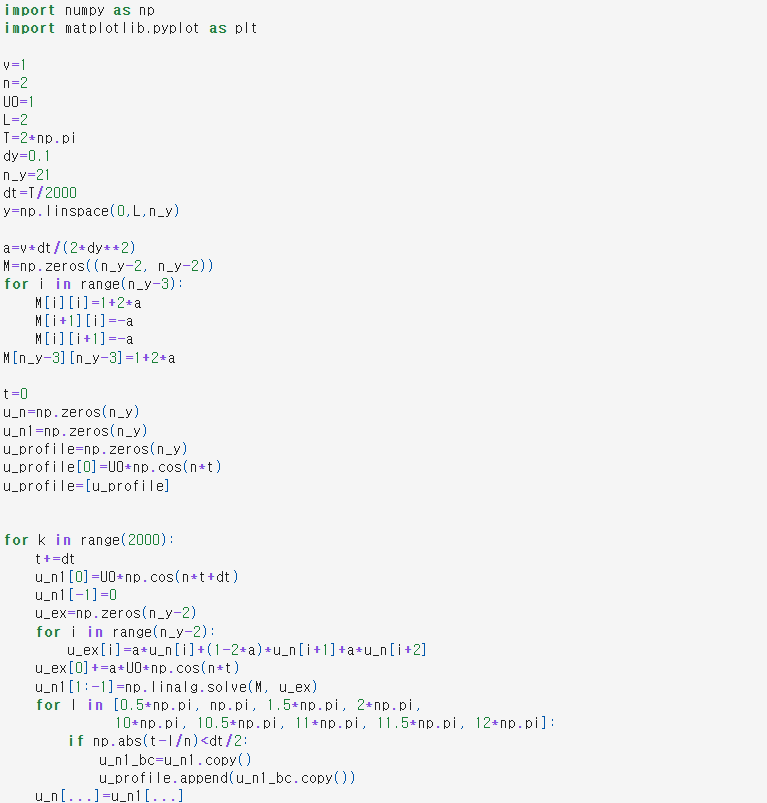
order of accuracy: 1.2029220799363685

2-4

텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

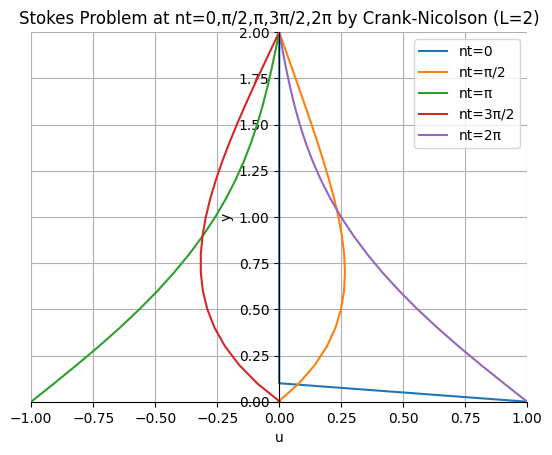
코드는 L=2, n\_y=21으로 변경된 것 외에는 2-2와 동일하다.



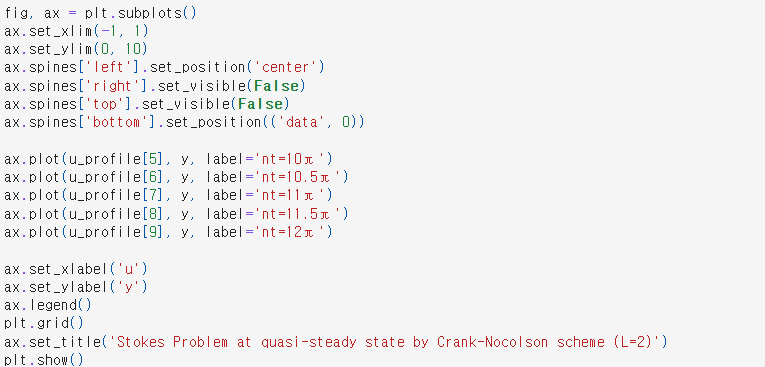
Crank-Nicolson scheme at nt=0, π/2, π, 3π/2, 2π의 그래프 코드와 결과

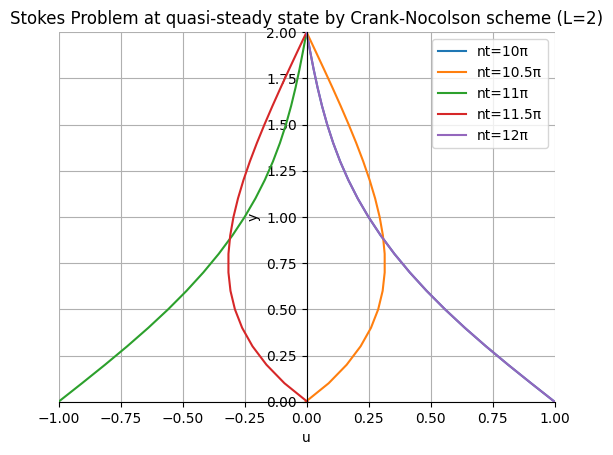






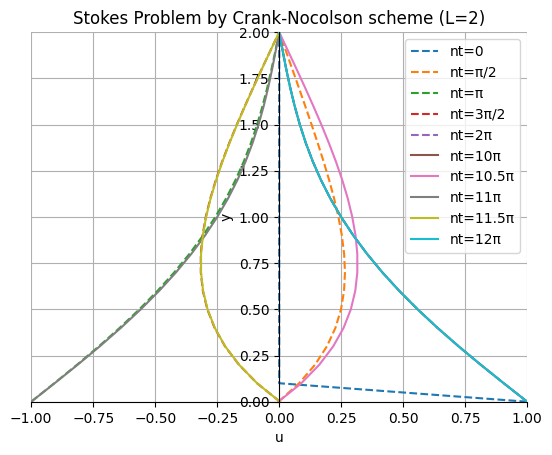
Crank-Nicolson scheme at quasi-steady state (nt=10π,10.5π,11π,11.5π,12π)의 그래프 코드와 결과





Crank-Nicolson scheme at nt=0, π/2, π, 3π/2, 2π, 10π, 10.5π, 11π, 11.5π, 12π 그래프 코드와 결과





코드에는 y를 0부터 10까지 보이는 것으로 써 있으나 더 편하게 보기 위해 0, 2으로 바꾸어 진행하였다.

결과적으로 L=10일 때보다 진동의 영향이 L까지 더욱 효과적으로 전달되는 것을 발견할 수 있다.