|  |  |
| --- | --- |
| **| 문제의도** | * PyTorch에서 텐서 연산을 수행하는 방법을 학습하고, Out-of-place 연산과 In-place 연산의 차이를 이해하기 * 두 개의 3x3 텐서를 생성하여 다양한 연산을 수행하며, In-place 연산이 원본 텐서에 미치는 영향을 확인하기 |
| **| 사전지식** | * PyTorch 라이브러리 설치 및 사용법 * 텐서 연산의 개념과 Out-of-place, In-place 연산의 차이 이해 |
| **| 문제핵심** | * torch.rand()를 사용하여 3x3 랜덤 텐서를 생성하는 과정 * 텐서 간의 덧셈, 뺄셈, 곱셈 연산을 수행하고, 각각의 결과를 확인하는 과정 * In-place 연산을 사용하여 원본 텐서에 직접 영향을 주는 연산을 수행하는 방법 |

|  |  |
| --- | --- |
| **| 문제해설** | |
| **| 해설** | 1. 3x3 랜덤 텐서 생성  3x3 크기의 랜덤 텐서를 생성하기 위해 torch.rand() 함수를 사용합니다. torch.rand() 함수는 0과 1사이의 난수를 포함한 텐서를 생성합니다.  2. 텐서 연산 수행  두 텐서 간의 덧셈, 뺄셈, 곱셈 연산을 수행하기 위해 Out-of-place 방식으로 연산을 합니다. Out-of-place 방식은 결과를 새로운 텐서에 저장하기 위해 사용하며, 이 방식은 원본 텐서가 변경되지 않고 결과가 새로 생성된 텐서에 저장됩니다.  3. In-place 연산 수행  원본 텐서에 직접 연산 결과를 반영하기 위해 add\_() 함수를 사용하여 In-place 방식으로 덧셈 연산을 수행합니다. In-place 연산은 기존 텐서 a에 바로 연산 결과가 반영되므로, 원본 텐서가 변경됩니다. |