|  |  |  |
| --- | --- | --- |
| 1주차 선형시스템 보고서 | | |
| 작성일 : 2024년 03월 11일 | | 작성자 : 이준용 |
| 구분 | 내용 | |
| 학습 범위와 내용 | 1. 1주차 온라인 강의 내용 | |
| 정리 내용 | 1.2.1 IEEE 64-bit floating point number representation  IEEE에서 개발한 컴퓨터에서 부동소수점을 표현하는 가장 널리 쓰이는 표준  실습 숫자 8.5  2진수 변환 하면 1000.1 -> 변환한 2진수를 정규화  1000.1 정규화하면 1.0001 \* 2^3  지수부(exponent)가 n 비트라면 2^(n-1)-1을 bias 상수라고함 -> 1023 + 3 =1026  10진수 1026을 2진수로 변환하여 지수부에 대응  1026 -> 10000000010  가수부는 정규화 한 값의 소수점 아래 부분을 가수부에 순서대로 입력하고 나머지는 0으로 처리  소수점 아래 부분은 0001이므로 0001000000….0000  부호 1bit => 0  지수부 11bit => 10000000010  소수부 52bit => 000100000…0000  1.2.2 Various kinds of computing Errors   * Truncation Error   무한으로 또는 적분을 유한한 개수의 항으로 근사화할 때 발생한다. 정확한 해를 얻기 위해서는 무한한 항이 필요하지만 유한한 항으로만 근사하는 경우 발생하는 오차이다.   * Round-off Error   실수를 컴퓨터의 유한한 메모리에 표현할 때 정밀도에 제한이 있다. 반올림 오차는 실수를 부동 소수점 표현으로 근사화할 때 발생하며, 실제 값과 저장된 값 사이에 차이가 발생한다.   * Overflow/Underflow:   컴퓨터에서는 숫자를 유한한 비트로 표현한다. 오버플로는 계산 결과가 최대 표현 가능 값보다 큰 경우 발생하며, 언더플로는 결과가 가장 작은 표현 가능 값보다 작은 경우 발생한다. 이는 계산에서 오류나 예상치 못한 동작을 초래할 수 있다.   * Negligible Addition   크기가 크게 차이나는 두 숫자를 더할 때, 작은 숫자가 정밀도에서 사실상 "손실"될 수 있다. 이로 인해 작은 숫자가 합에 기여하지 않아 오차가 발생할 수 있다.   * Loss of Significance   거의 같은 두 숫자를 빼는 경우 발생한다. 제한된 정밀도로 인해 결과에 중요한 숫자가 손실되어 계산의 정확도가 감소한다.   * Error Magnification   작은 오차를 포함하는 숫자를 큰/작은 숫자로 나누거나 곱할 때 오류가 크게 증폭될 수 있습니다. 작은/큰 숫자로 나누거나 곱함으로써 작은 오차가 크게 확대될 수 있다.   * Errors depending on the numerical algorithms, step size, and so on   수치 알고리즘의 선택과 수치 방법에서 사용되는 단계 크기는 오류를 도입할 수 있다. 서로 다른 알고리즘은 정확도와 안정성이 다를 수 있으며, 단계 크기의 선택은 해결의 정밀도에 영향을 줄 수 있다.  1.2.3 Absolute/Relative Computing Errors  Absolute Computing Errors: 이는 계산된 값과 실제 값 사이의 차이를 나타냅니다. 즉, 절대적 오차는 실제 값과 계산된 값 사이의 절대적인 차이를 의미.  Relative Computing Errors: 이는 절대적 오차를 실제 값으로 나눈 것으로, 계산된 값이 실제 값에 대해 얼마나 큰지를 나타냅니다. 상대적 오차는 보통 백분율로 표현되며, 절대적 오차를 실제 값으로 나눈 후 100을 곱하여 계산된다.  1.3.2 Vector Operation Versus Loop Iteration  벡터 연산(Vector Operation): 벡터 연산은 배열이나 벡터를 한 번에 처리하는 방법입니다. 이 방식은 많은 경우에 더 효율적이며, 컴퓨터의 병렬 처리를 활용하여 계산을 가속화할 수 있다.  루프 반복(Loop Iteration): 루프 반복은 배열이나 리스트의 각 요소에 순차적으로 접근하고 작업을 반복하는 방법. 각 요소에 대해 연산을 반복하므로 연산이 더 길어질 수 있다. | |
| 질문 내용 | **Q1절대적 오차와 상대적 오차 중에서 어떤 것이 특정 문제를 해결하는 데 더 유용한지 결정하는 데 어떤 요인들을 고려해야 할까요?**  **Q2런타임 오류를 피하기 위해 벡터 연산과 루프 반복 중에서 어떤 접근 방식을 선택해야 하고, 각 방법이 어떻게 런타임 오류를 줄일 수 있는지 설명해주세요.** | |