**고급소프트웨어 실습 보고서 #1**

**20191657 최세은**

난수 생성 방법

1. 선형 합동 생성기(Linear congruential generator, LCG)

C 언어의 경우 일반적으로 사용하는 stblib.h에 포함되어 있는 srand(), rand() 함수 등은 선형 합동법으로 구현되어 있다. 선형 합동 생성기는 다음 재귀 관계로 정의된 순열 Xi을 반환한다.



선형 합동법은 아주 빠르고 적은 메모리를 사용한다. 또한 이해하기가 쉽다. 그러나 비록 최대 주기를 가지도록 인자를 선택했더라도 아주 좋은 품질의 난수열을 생성해 내지 못한다. 예를 들어 선형 합동 생성기가 만드는 연속된 난수들 사이에 상당한 상관 관계가 존재하기 때문에 몬테 카를로 시뮬레이션에 적합하지 않으며, 마지막으로 생성된 난수를 알면 그 뒤에 만들어질 난수를 모두 예측할 수 있기 때문에 암호학적인 목적으로도 사용할 수 없다.

실습에서 선형 합동 생성기로 다음과 같이 난수를 생성했다.

|  |
| --- |
| srand(time(NULL));  a = rand() % 201 - 100;  b = rand() % 201 - 100; |

2. 메르센 트위스터

메르센 트위스터(Mersenne Twister)는 1997년에 마츠모토 마코토와 니시무라 다쿠지가 개발한 유사난수 생성기로 기존 생성기들의 문제점들을 피하면서 매우 질이 좋은 난수를 빠르게 생성할 수 있도록 설계되었다.

메르센 트위스터의 이름은 난수의 반복 주기가 메르센 소수인 데에서 유래했다. 메르센 트위스터는 그 속도와 난수의 품질 때문에 점점 많은 곳에서 채택되고 있으며, 흔히 주기가 (219937-1)인 MT19937을 사용한다. 따라서 주기가 매우 길며 높은 동일분포률을 가진다. 또한 비트 연산만으로 알고리즘의 구현이 가능하기 때문에 매우 빠르다. 시스템에 의존하지만 MT는 파이프라인이나 캐쉬 메모리를 가진 시스템에서 때때로 표준 ANSI-C 라이브러리 보다 빠르게 작동한다고 보고되었다.

|  |
| --- |
| #include <random>  mt19937 engine((unsigned int)time(NULL));  uniform\_int\_distribution<int> distribution(-100, 100);  a = distribution(engine);  b = distribution(engine); |