

과제 #1

4190.414A 멀티코어 컴퓨팅(001)

Due: 2021년 3월 29일(월) 23시 59분

1 문제 1: 컴퓨터에서의 수 표현

수 하나를 `scanf`로 입력받아 첫번째 줄에는 `printf`로 그대로 출력하고, 두번째 줄에는 이진 표현으로 변환 후 출력하는 프로그램을 작성하라. C언어의 `int`, `long`, `float`, `double`, `long double` 타입에 대해 동작하는 `int.c`, `long.c`, `float.c`, `double.c`, `longdouble.c`를 각각 작성하여 제출하면 된다. 아래는 `float.c` 프로그램 실행 예시이다.

```
[jinpyo@login]$ ./float
Enter a number: 103.625
103.625000
01000010110011110100000000000000
[jinpyo@login]$ ./float
Enter a number: -31415950000
-31415949312.000000
11010000111010100001000100101011
```

입력 받은 수와 첫번째 출력이 다른 경우가 있는가? 두번째 출력은 수업에서 배운 것과 일치하는가?

2 문제 2: 부동소수점 연산 성능

부동 소수점 표현으로 저장된 수로 덧셈 또는 곱셈을 각각 충분한 횟수만큼 수행하고 걸린 시간과 FLOPS(floating point operations per second)를 보고하라. 얻은 FLOPS와 사용한 CPU의 theoretical peak FLOPS를 비교하여 예측한 값을 얻었는지, 그렇지 않다면 원인이 무엇인지 고민해보자. `float`와 `double` 타입에 대해 실험을 수행하면 된다.

3 문제 3: Vector Instruction

Vector instruction에 대해 간단히 알아보고, 사용하는 컴파일러(e.g., `gcc`)에서 명시적으로 vector instruction을 사용하는 방법에 대해 조사하라. 문제 2와 같은 실험을 vector instruction을 사용해서 수행하고 결과를 비교해보라. 또한, vector instruction 중 FMA(Fused Multiply-Add)를 사용해보고 곱셈과 덧셈을 따로 한 경우와 비교해 보라.

참고 Intel AVX2, `gcc x86 built-in functions`, `gcc vector extensions`

4 주의 사항

- 실습용 서버가 아닌 다른 서버에서 과제를 진행해도 좋다. 다만 보고서에 실험 환경을 명시해야 한다.
- 성능 측정 시 반드시 실제로 흐른 시간(wall clock time)을 사용한다. 개인의 판단에 따라 유의하지 않은 부분(e.g., 입출력과 메모리 할당)은 측정에서 제외해도 좋다.
- 보고서는 간략하게 필요한 내용만 적는다.
- 보고서에는 반드시 컴파일 및 실행 방법이 명시되어 있어야 한다.

5 제출 방법

- 조교 메일(jinpyo@aces.snu.ac.kr)로 보고서를 포함한 모든 파일을 하나의 파일(e.g., .zip, .tar.gz)로 압축 후 첨부하여 제출한다
- 메일 제목은 [mc2021] 계정이름_HW1으로 한다. (e.g., [mc2021] mc99_HW1)
- 첨부파일명은 계정이름_HW1.확장자로 한다. (e.g., mc99_HW1.zip, mc99_HW1.tar.gz)
- 제출한 메일은 기계적으로 처리되므로 위의 내용을 지키지 않을 시 누락될 수 있으니 잘 지켜주시기 바랍니다.
- Grace day를 사용하고자 하는 경우에는 메일 내용에 이를 반드시 포함한다.