2021 csed211 lab2

20200445 박은하

이번 lab에서 가장 중요한 변수는 sign과 exp와 frac이다. 각각은 float의 bit pattern중 sign부분과 exp부분 frac부분이고 (sign<<31) | (exp << 23) | frac라는 식을 통해 float의 비트패턴을 인코딩할 수 있다.

1. float\_neg

unsigned로 sign, exp, frac 변수를 선언해 비트패턴을 저장한다

NaN인 경우 매개변수에 담긴 값을 그대로 리턴하고, 아닌 경우 sign값만 반대로 해서 리턴한다.

1. float\_i2f

정수를 float의 비트패턴으로 encoding하는 함수이다.

만약 x가 0인 경우 0을 return한다.

여기서 기본적인 아이디어는 음수나 양수나 절댓값이 같은 경우 float로 인코딩할 때 frac부분과 exp부분의 비트패턴이 동일하다는 것이다. 따라서 sign을 통해 음수로 인식한 경우 양수로 바꾸고 exp와 frac을 구한다. 이때 integer의 min값은 대응되는 max가 없으므로 else if를 통해 따로 처리해주었다.

다음은 해당하는 양수가 2의 몇승인지 처리하기 위해 우측 쉬프트를 한 값이 0이될때까지 E라는 변수를 증가시켰다. 따라서 exp=E+bias(127)로 구할 수 있었다.

X를 31-E만큼 left shift하면 32개의 bit가 생성되고 좌측부터 M의 값이 생긴다. 따라서 float의 frac파트는 left shift한 x를 다시 우측으로 8만큼 shift시켜 0x7ffff와 bitwise operator&을 취한 소수점 앞의 1을 생략한 23개의 비트부분이다.

만약 frac 부분을 통해 integer를 다 나타낼 수 없을 때 생략되는 부분을 반올림해주는 과정이 필요하다. 이때 E는 2의 23승보다 크다고 할 수 있고 반올림 방법은 round to even이다. 따라서 32비트 중 생략되는 부분이 least significant byte인데, 이 byte가 0x80이면서 frac의 첫번째 자리가 1일경우 반올림되고 아예 0x80보다 클 경우 또한 반올림된다. 반올림되면서 frac부분이 23비트를 넘을 경우 frac를 0으로하고, exp를 증가시킨다.

1. float\_twice

float를 두 배한 값의 비트 패턴을 리턴하는 함수이다.

매개변수로 들어온 float의 비트패턴 값이 NaN일 경우 값을 그대로 리턴한다.

만약 exp==0일 경우 denormalized value인데 이 때는 uf를 좌측으로 1만큼 shift한다.

만약 exp<=0xfe인 경우 normalized value인데 이때는 exp를 1만큼 증가시킨다. 만약 증가시켜서 exp가 0xff가 될 경우 infinity case이므로 frac을 0으로 set한다.

1. float\_abs

부동 소수점 f의 절댓값 bit pattern을 반환하는 함수이다. 매개변수로 들어온 f의 bitpattern을 통해 NaN이라고 판단 될 때 그대로 매개변수 값을 반환한다.

만약 그 외의 경우에는 0x7fffffff&uf를 통해 절댓값을 반환한다.

1. float\_half

부동 소수점 f를 2로 나눈 값의 bit pattern을 반환하는 함수이다. 이 함수 또한 NaN인 경우 매개변수 값을 그대로 반환한다.

rounding방법으로 rounding to even을 사용하는데 삭제되는 값과 그 앞의 값을 고려하여 frac에 1을 더하거나 더하지 않는 방법으로 표현한다. 2로 나눠지면 1만큼 우측 쉬프트가 이루어지기에 frac&3한값이 3과 같을 경우 반올림된다. 따라서 up변수에 (frac&3>>1)&(uf&1)의 값을 저장하여 반올림하여 frac 파트에 1을 더할 때 이용한다.

만약 denormalized value일 경우 frac을 우측으로 1만큼 shift시키고 frac의 값에 up을 더해서 frac을 갱신한다.

만약 exp가 1인 normalized value인 경우 2로 나누면 denormal이 된다. 이때 exp는 0이되고 frac은 uf&0x7fffffff를 1만큼 우측 shift하고 up을 더해준 값에 bitwise operator &을 통해 0x7fffff를 취해 앞에서부터 23비트만큼 구한 값이다.

나머지 normalized value의 경우 exp의 값을 1만큼 감소시킨다.