

$x, y$  는  $e^x = y$  를 만족하는 Float32 타입의 수,  $i, j$  는 각각  $x, y$  의 각 비트를 보존한 채 Int32 로 인식시킨 수라고 하자.  
 $b \in [0, 1]$  일 때  $\log_2(1 + b)$  를  $b + 0.0573$  으로 근사시킬 수 있으므로

$$\begin{aligned} x &= \ln y \\ &= \frac{\log_2 y}{\log_2 e} \\ &\simeq \frac{1}{\log_2 e} [2^{-23}j - 127 + 0.0573] \end{aligned}$$

이다. 우변을 정리하여

$$2^{23}(x \log_2 e + 127 - 0.0573) = j$$

을 얻을 수 있다. 따라서,

$$y \simeq * (\text{Float32} *) (2^{23}(x \log_2 e + 127 - 0.0573) \text{ as Int32})$$

이다. 이제  $x, y$  가 Float64 타입이라 하면

$$x \simeq \frac{1}{\log_2 e} [2^{-52}j - 1023 + 0.0573]$$

이므로

$$y \simeq * (\text{Float64} *) (2^{52}(x \log_2 e + 1023 - 0.0573) \text{ as Int64})$$

이다. 이러한 방법으로 지수를 계산하는 함수를 다음 페이지에서 작성해 보았다.

```

const mult32:f32 = 12102203.161561485;
const adder32: f32 = 1064872507.1615615;

const mult64:f64 = 6497320848556798.0;
const adder64: f64 = 4606924340207518000.0;

fn fast_exp32(x: f32) -> f32 {

    union U {
        f: f32,
        i: i32,
    }

    unsafe {
        let mut u = U { f: x };
        u.i = (u.f * mult32 + adder32) as i32;
        u.f
    }
}

fn fast_exp64(x: f64) -> f64 {

    union U {
        f: f64,
        i: i64,
    }

    unsafe {
        let mut u = U { f: x };
        u.i = (u.f * mult64 + adder64) as i64;
        u.f
    }
}

```

$x$  를 1 에서 701 까지 0.1 씩 증가시켜서 fast\_exp64 를 계산하여 얻은 오차는 아래와 같다.

```
e^1 : trad = 3.00416602e0, fast = 3.05932909e0, error = 9.81968901e-1
e^101 : trad = 8.07555019e43, fast = 8.02456026e43, error = 1.00635423e0
e^201 : trad = 2.17080249e87, fast = 2.12590311e87, error = 1.02112014e0
e^301 : trad = 5.83537138e130, fast = 5.93651567e130, error = 9.82962348e-1
e^401 : trad = 1.56861618e174, fast = 1.59051345e174, error = 9.86232580e-1
e^501 : trad = 4.21662405e217, fast = 4.14155986e217, error = 1.01812462e0
e^601 : trad = 1.13347794e261, fast = 1.12837100e261, error = 1.00452594e0
e^701 : trad = 3.04692148e304, fast = 3.10776457e304, error = 9.80422233e-1
Deviation : 0.01791137, Min_error : -0.01974522 at 521, Max_error : 0.04048682 at 520
```

$x = 88$  에 대해 32 비트 지수 계산을 100000회,  $x = 400$  에 대해 64 비트 지수 계산을 100000회 반복하는데 소요된 시간은 다음과 같다.

```
test fast_exp32 ... bench:      465,635 ns/iter (+/- 75,791)
test fast_exp64 ... bench:      467,780 ns/iter (+/- 30,638)
test trad_exp32 ... bench:      624,230 ns/iter (+/- 53,545)
test trad_exp64 ... bench:      672,160 ns/iter (+/- 188,869)
```

사용된 시스템 : AMD Ryzen 5 5800X, 32GB DDR4-3200, Windows 11, Rust 1.73.0