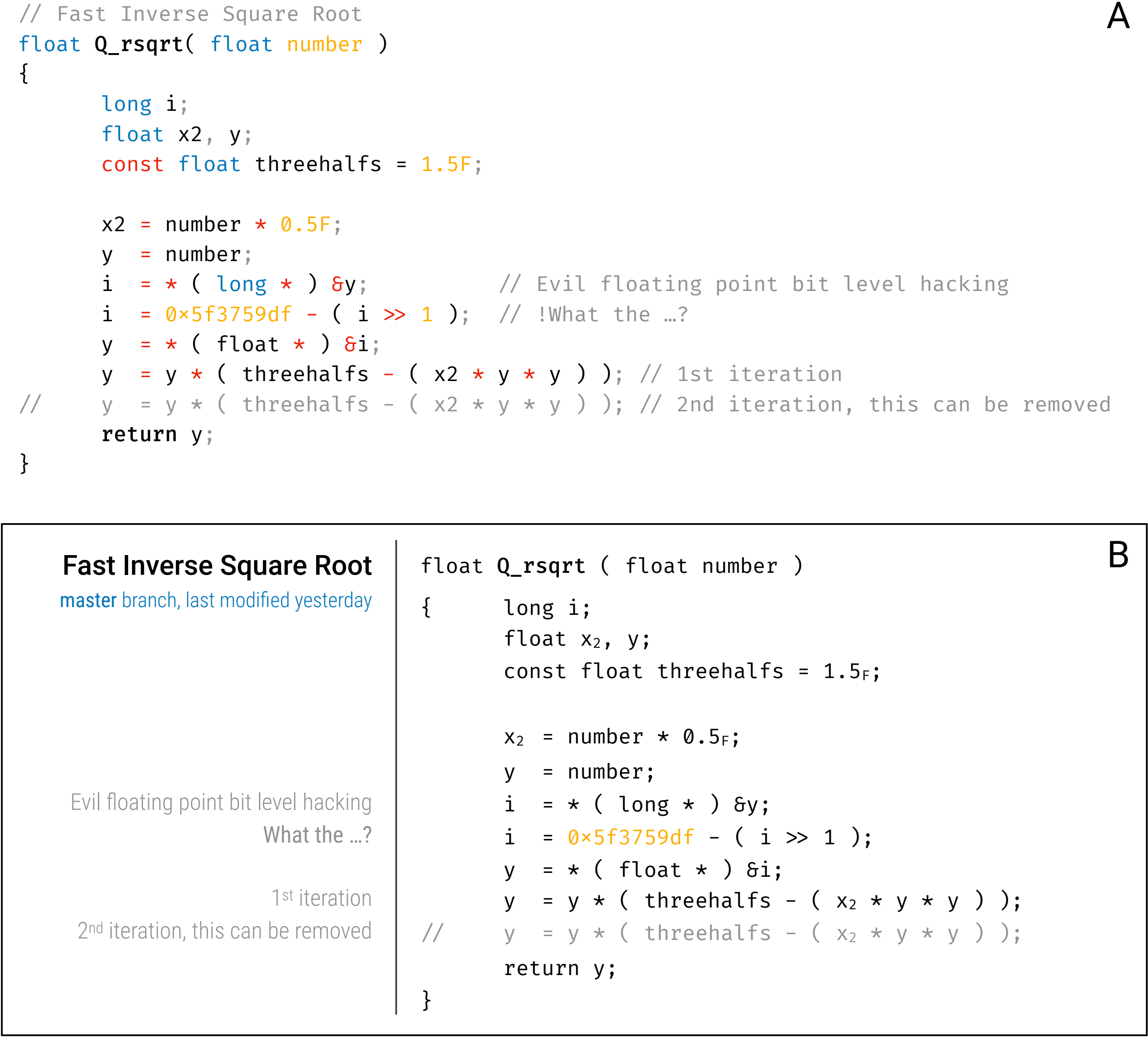

```c
// Fast Inverse Square Root
float Q_rsqrt( float number )
{
	long i;
	float x2, y;
	const float threehalfs = 1.5F;

	x2 = number * 0.5F;
	y  = number;
	i  = * ( long * ) &y;             // Evil floating point bit level hacking
	i  = 0×5f3759df - ( i >> 1 );   // !What the …?
	y  = * ( float * ) &i;
	y  = y * ( threehalfs - ( x2 * y * y ) ); // 1st iteration
//	y  = y * ( threehalfs - ( x2 * y * y ) ); // 2nd iteration, this can be removed
	return y;
}
```

**Fast Inverse Square Root**

master branch, last modified yesterday

Evil floating point bit level hacking

What the …?

1st iteration

2nd iteration, this can be removed

```c
float Q_rsqrt ( float number )
{       long i;
        float x2, y;
        const float threehalfs = 1.5F;

        x2  = number * 0.5F;
        y   = number;
        i   = * ( long * ) &y;
        i   = 0×5f3759df - ( i >> 1 );
        y   = * ( float * ) &i;
        y   = y * ( threehalfs - ( x2 * y * y ) );
//      y   = y * ( threehalfs - ( x2 * y * y ) );
        return y;
}
```