```ts
function fuji<TS extends RuleType, A, B = A>(r1: Rule<TS, A, B>):
Fuji<TS, B>

function fuji<TS extends RuleType, A, B = A, C = B>(
  r1: Rule<TS, A, B>,
  r2: Rule<TS, B, C>
): Fuji<TS, C>

function fuji<TS extends RuleType, A, B = A, C = B, D = C>(
  r1: Rule<TS, A, B>,
  r2: Rule<TS, B, C>,
  r3: Rule<TS, C, D>
): Fuji<TS, D>

function fuji<TS extends RuleType, A, B = A, C = B, D = C, E = D>(
  r1: Rule<TS, A, B>,
  r2: Rule<TS, B, C>,
  r3: Rule<TS, C, D>,
  r4: Rule<TS, D, E>
): Fuji<TS, E>

function fuji<TS extends RuleType, A, B = A, C = B, D = C, E = D, F =
E>(
  r1: Rule<TS, A, B>,
  r2: Rule<TS, B, C>,
  r3: Rule<TS, C, D>,
  r4: Rule<TS, D, E>,
  r5: Rule<TS, E, F>
): Fuji<TS, F>
```

Object.assign

```
assign<T, U>(target: T, source: U): T & U;
assign<T, U, V>(target: T, source1: U, source2: V): T & U & V;
assign<T, U, V, W>(target: T, source1: U, source2: V, source3: W): T & U & V & W
```