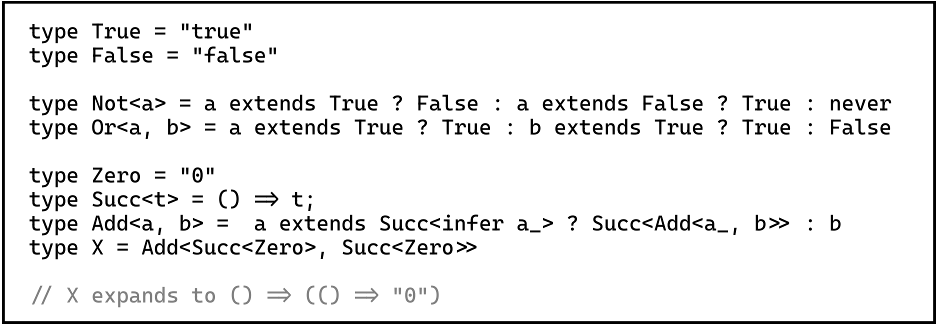
conditional.ts



```typescript
type True = "true"
type False = "false"

type Not<a> = a extends True ? False : a extends False ? True : never
type Or<a, b> = a extends True ? True : b extends True ? True : False

type Zero = "0"
type Succ<t> = () => t;
type Add<a, b> =  a extends Succ<infer a_> ? Succ<Add<a_, b>> : b
type X = Add<Succ<Zero>, Succ<Zero>>

// X expands to () => (() => "0")
```