```
code = | src = "?"
       |
       | getSrc()
       |    src
       |    .replace("?", escape(src))
       |
       |
       |
```

escape(code)

$V(x,y) = p(<x,y>)$ — вычислимая ф-ия от двух переменных

$V(src, x)$
  print(count(x, src))

Преобразуем

$V(x)$ escape
  ┌ src = '?'
  └ src = replace(src, '?', escape(src))

HALT не разрешима

∃ разрешим
  halt(p)
  q(x)
    if(halt(q)):
        while true
    else
        return 1

И не разрешим
  q(x)
    if u(q,x)
        return 0
    else
        return 1

Ⅱ о рекурсии (о рефлексии)
  $V(x,y)$ — вычисл. ф-ия от 2 аргументов
  ∃ программа $r(t)$
  ∀t $V(r, t) = r(t)$

$V(q, x)$
  if halt(q):
  | while true
  else
  | return 1

∀ t. $r(t) = V(r, t)$
  $V(q,t)$ зависаем ⟺ q(ε) ост.
  $r(ε) = V(r, ε)$ зависаем ⟺ r(ε) ост.

───────────────────────────────

Ⅱ (У-Р) ∀ нетр. св-во вычисл. языков не разрешимо
            inA(p)
  ▶   A — нетр. св-во
        L ∈ A   inL(x)
        M ∉ A   inM(x)

        p(x)
          if inA(p):
              return inM(x)
          else
              return inL(x)
```

Вторая $\underline{\underline{T}}$ Гёделя о неполноте

$\forall$ дост. бог. форм. системе $\exists$ ист. недок. утв.

<span style="color:blue">Программа p ост. на входе x</span>

$p(x)$
  s := "p не ост. на входе x"
  for $t \in \Sigma^*$
      если t - док-во s
          return

<span style="color:red">предположили непротиворечивость</span>

s - ложно $\Rightarrow$ $\nexists t$ -док-во s $\Rightarrow$ p не зав. $\Rightarrow$ s ист.

s - ист. $\Rightarrow$ ( s доказуемо $\Rightarrow$ s - ложно)

s - ист., s не док.

$\underline{\underline{T}}1$ Гёделя о неполноте

$\forall$ дост. бог. сист. не может доказать свою непротиворечивость

$\underline{\underline{T}}$ (о неподвижн. точке)

$\forall$ всюду определ. выч. ф-ции f $\exists$ программа p: $\forall t \; p(t) = q(t)$, где $q = f(p)$

<span style="color:blue">$p(x)$
    $q = f(p)$
    return $q(x)$</span>