Mit viel Hilfe von Luca Nicolas Kempkes.

```python

def root(n):
    # {P = n > 0}
    r = 0
    k = 1
    # {INV} = {r == S(k-1) ^ (S(k-1) + S(k-2)) < n}
    while (r+r+k) < n:
        r = r + k
        k = k + 1

    # {INV v ~B} = {()r == S(k-1) ^ (S(k-1) + S(k-2)) < n)}
    # {Q = ((k - 1) ** 2) < n <= (k**2)}

    return k
```

Zu $\{INV \wedge \neg B\}$

Beginnend mit der 1. Bedingung: $(r == S(k-1))$

$r == S(k-1)$

$r == \sum_{i=0}^{k-1} i$ Addiert man auf beiden Seiten k:

$r+k = \sum_{i=0}^{k} i$ Dann folgt daraus, dass $\underline{r+k = S(k)}$

2. Bedingung: $(S(k-1)+S(k-2)) < n$

Wir setzen für $S(k-1)$ r ein, wie oben gezeigt:

$(r+k+r < n)$ Das ist gleich B.

Wenn $\underbrace{(S(k-1)+S(k-2)) < n}_{INV}$ und $\underbrace{(r+k+r < n)}_{B}$ gelten,

dann $\{INV \wedge B\}$. Daraus kann man ableiten,

dass INV ein gültige Invariante für die Schleife ist.