Basic Output and Loops in Lazy-K

```
# Lazy-K output with finite-loop
# commented example code.

K(  # Consume input, produce output
IS  # redundant I, added by the build script to give "K. IS" ...

# Begin section: "K. is"
(SI(K(      # K (75)
  S(K(S(S(KS)K)(S(S(KS)K)I)))(S(S(KS)K)I(S(S(KS)K)(SII(S(S(KS)K)I))))
)))
(K(S(SI(K(  # . (46)
  S(S(KS)K)(S(K(S(S(KS)K)(SII(S(S(KS)K)I))))(S(S(KS)K)I(S(S(KS)K)(S(S(KS)K)I))))
)))
(K(S(SI(K(  # Space (32)
  S(K(S(S(KS)K)I))(S(SII)I(S(S(KS)K)I))
)))
(K(S(SI(K(  # i:
  S(S(KS)K)(S(K(SII(S(S(KS)K)I)))(S(S(KS)K)(S(S(KS)K)I(S(S(KS)K)(SII(S(S(KS)K)I))))))
)))
(K(S(SI(K(  # s:
  S(S(KS)K)(S(K(S(S(KS)K)I))(S(S(KS)K)(S(K(S(S(KS)K)I))(S(S(KS)K)(SII(S(S(KS)K)(S(S(KS)K)I)))))))
)))

# Middle section:
(K(
# Number of repetitions of Noun:
((
  S(K(S(S(KS)K)I))(S(S(KS)K)(SII(S(S(KS)K)I)))  # 10 times
# 49998:
# S(S(KS)K)(S(S(KS)K)I)(S(S(KS)K)(S(S(KS)K)I(S(S(KS)K)(SII(S(S(KS)K)I))))(S(S(KS)K)
# (S(K(S(S(S(KS)K))(SII)(S(S(KS)K)I)))(S(K(S(S(KS)K)I))(S(S(KS)K)(SII(S(S(KS)K)I)))))

# Noun
(S(K(    # Apply (o x y) => S(K x)y
S(K(
S(SI(K(  # cons (join characters into a string) (cons x y) => S(SI(K x))(K y)
  S(K(S(S(KS)K)I))(S(SII)I(S(S(KS)K)I))  # Space (32)
)))))K))
(S(K(S(K(
S(SI(K(
  S(S(KS)K)(S(S(S(KS)K))(SII)(S(S(KS)K)I))  # A (65)
)))))K))
(S(K(S(K(
S(SI(K(
  S(S(KS)K)(S(S(S(KS)K))(SII)(S(S(KS)K)I))
)))))K))
(S(K(S(K(
S(SI(K(
  S(S(KS)K)(S(S(S(KS)K))(SII)(S(S(KS)K)I))
)))))K))
(S(K(S(K(
S(SI(K(
  S(S(KS)K)(S(S(S(KS)K))(SII)(S(S(KS)K)I))
)))))K))
(S(K(
S(SI(K(
  S(S(KS)K)(S(SII)I(S(S(KS)K)I))(S(S(KS)K))(SII(S(S(KS)K)(S(S(KS)K)I)))  # Comma (44)
)))))K))))))))

# End section
(S(SI(K(  # ASCII 08 (backspace) cheating to overwrite the last comma of the repeated section
  S(S(KS)K)(S(S(KS)K)I)(S(S(KS)K)I)
)))
(K
(S(SI(K(  # period . (46)
  S(S(KS)K)(S(K(S(S(KS)K)(SII(S(S(KS)K)I))))(S(S(KS)K)I(S(S(KS)K)(S(S(KS)K)I))))
)))
(K
# End of output (256)
(K(SII(SII(S(S(KS)K)I))))))))))))))))))))
```

K. is AAAA, AAAA, AAAA, AAAA, AAAA, AAAA, AAAA, AAAA, AAAA, AAAA.