Kolejka FIFO na liście cyklicznej 01

-------------------------------------

Rozbuduj program realizujący funkcjonalność kolejki FIFO na strukturze jednokierunkowej listy cyklicznej.

Wejście

Ciąg trzech komend: P, L, A rozdzielonych znakiem nowej linii.

Wyjście

Rezultaty użycia jednej z trzech dostępnych komend:

P – drukuj wszystkie elementy włożone do kolejki rozdzielone ciągiem znaków: "–>". Jeżeli kolejka jest pusta drukuj: "–" (już zaimplementowane).

L – drukuj liczbę elementów listy (już zaimplementowane).

A – powiększ listę dodając nowy, pusty element.

Przykład

Wejście

P

L

A

P

L

A

A

A

A

P

L

Wyjście

-

0

-

1

-

5

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Kolejka FIFO na liście cyklicznej 02

--------------------------------------

Rozbuduj program realizujący funkcjonalność kolejki FIFO na strukturze jednokierunkowej listy cyklicznej.

Wejście

Ciąg trzech komend: P, L, E n rozdzielonych znakiem nowej linii.

Wyjście

Rezultaty użycia jednej z trzech dostępnych komend:

P – drukuj wszystkie elementy włożone do kolejki rozdzielone ciągiem znaków: "–>". Jeżeli kolejka jest pusta drukuj: "–" (już zaimplementowane).

L – drukuj liczbę elementów listy (już zaimplementowane).

E - wstaw element o kluczu n do kolejki, jeżeli nie ma miejsca powiększ listę dodając nowy, pusty element.

Przykład

Wejście

P

L

E 4

P

L

E 7

E -1

P

L

E 0

E 4

P

L

Wyjście

-

0

4

1

4->7->-1

3

4->7->-1->0->4

5

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Kolejka FIFO na liście cyklicznej 03

--------------------------------------

Rozbuduj program realizujący funkcjonalność kolejki FIFO na strukturze jednokierunkowej listy cyklicznej.

Wejście

Ciąg trzech komend: P, L, E n, D rozdzielonych znakiem nowej linii.

Wyjście

Rezultaty użycia jednej z trzech dostępnych komend:

P – drukuj wszystkie elementy włożone do kolejki rozdzielone ciągiem znaków: "–>". Jeżeli kolejka jest pusta drukuj: "–" (już zaimplementowane).

L – drukuj liczbę elementów listy (już zaimplementowane).

E - wstaw element o kluczu n do kolejki, jeżeli nie ma miejsca powiększ listę dodając nowy, pusty element.

D - wyjmij i wydrukuj element z kolejki, jeżeli kolejka jest pusta wydrukuj: '-'. Nie usuwaj elementów listy.

Przykład

Wejście

P

L

D

E 4

P

L

E -1

E 4

P

L

D

P

L

D

P

L

D

P

L

Wyjście

-

0

-

4

1

4->-1->4

3

4

-1->4

3

-1

4

3

4

-

3

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Kolejka FIFO na liście cyklicznej 04

-------------------------------------

Rozbuduj program realizujący funkcjonalność kolejki FIFO na strukturze jednokierunkowej listy cyklicznej.

Wejście

Ciąg trzech komend: P, L, E n, D, G rozdzielonych znakiem nowej linii.

Wyjście

Rezultaty użycia jednej z trzech dostępnych komend:

P – drukuj wszystkie elementy włożone do kolejki rozdzielone ciągiem znaków: "–>". Jeżeli kolejka jest pusta drukuj: "–" (już zaimplementowane).

L – drukuj liczbę elementów listy (już zaimplementowane).

E - wstaw element o kluczu n do kolejki, jeżeli nie ma miejsca powiększ listę dodając nowy, pusty element.

D - wyjmij i wydrukuj element z kolejki, jeżeli kolejka jest pusta wydrukuj: '-'. Nie usuwaj elementów listy.

G - usuń wszystkie puste elementy z listy.

Przykład

Wejście

G

E 4

G

P

L

D

G

P

L

G

P

L

E 4

E -1

E 4

E 0

E 7

P

L

D

D

D

G

P

L

Wyjście

4

1

4

-

0

-

0

4->-1->4->0->7

5

4

-1

4

0->7

2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Kolejka FIFO na liście cyklicznej 05

------------------------------------

Rozbuduj program realizujący funkcjonalność kolejki FIFO na strukturze jednokierunkowej listy cyklicznej.

Wejście

Ciąg trzech komend: P, L, C n rozdzielonych znakiem nowej linii.

Wyjście

Rezultaty użycia jednej z trzech dostępnych komend:

P – drukuj wszystkie elementy włożone do kolejki rozdzielone ciągiem znaków: "–>". Jeżeli kolejka jest pusta drukuj: "–" (już zaimplementowane).

L – drukuj liczbę elementów listy (już zaimplementowane).

C n - powiększ listę dodając n nowych, pustych elementów.

Przykład

Wejście

P

L

C 1

P

L

C 1

C 1000

P

L

Wyjście

-

0

-

1

-

1002

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Kolejka FIFO na liście cyklicznej 06

-------------------------------------

Rozbuduj program realizujący funkcjonalność kolejki FIFO na strukturze jednokierunkowej listy cyklicznej.

Wejście

Ciąg trzech komend: P, L, E n1,n2,...,nk rozdzielonych znakiem nowej linii.

Wyjście

Rezultaty użycia jednej z trzech dostępnych komend:

P – drukuj wszystkie elementy włożone do kolejki rozdzielone ciągiem znaków: "–>". Jeżeli kolejka jest pusta drukuj: "–" (już zaimplementowane).

L – drukuj liczbę elementów listy (już zaimplementowane).

E n1,n2,...,nk - wstaw k elementów o kluczu ni do kolejki, jeżeli nie ma miejsca powiększ listę dodając nowe, puste elementy.

Przykład

Wejście

P

L

E 4 3

P

L

E 7 43 65 55 4 7 5 2 3

E -1 -65 0 0 0 8 54

P

L

E 0 3 5 -8 543

E 4

P

L

Wyjście

-

0

4->3

2

4->3->7->43->65->55->4->7->5->2->3->-1->-65->0->0->0->8->54

18

4->3->7->43->65->55->4->7->5->2->3->-1->-65->0->0->0->8->54->0->3->5->-8->543->4

24

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Kolejka FIFO na liście cyklicznej 07

-------------------------------------

Rozbuduj program realizujący funkcjonalność kolejki FIFO na strukturze jednokierunkowej listy cyklicznej.

Wejście

Ciąg trzech komend: P, L, E n1,n2,...,nk, D n rozdzielonych znakiem nowej linii.

Wyjście

Rezultaty użycia jednej z trzech dostępnych komend:

P – drukuj wszystkie elementy włożone do kolejki rozdzielone ciągiem znaków: "–>". Jeżeli kolejka jest pusta drukuj: "–" (już zaimplementowane).

L – drukuj liczbę elementów listy (już zaimplementowane).

E n1,n2,...,nk - wstaw k elementów o kluczu ni do kolejki, jeżeli nie ma miejsca powiększ listę dodając nowe, puste elementy.

D n - wyjmij i wydrukuj n elementow z kolejki, jeżeli kolejka jest pusta wydrukuj: '-'. Nie usuwaj elementow listy.

Przykład

Wejście

P

L

D 3

E 4 3

P

L

E 7 43 65 55 4 7 5 2 3

E -1 -65 0 0 0 8 54

P

L

D 5

E 0 3 5 -8 543

D 8

E 4

P

D 13

P

L

Wyjście

-

0

- - -

4->3

2

4->3->7->43->65->55->4->7->5->2->3->-1->-65->0->0->0->8->54

18

4 3 7 43 65

55 4 7 5 2 3 -1 -65

0->0->0->8->54->0->3->5->-8->543->4

0 0 0 8 54 0 3 5 -8 543 4 - -

-

18

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Kolejka FIFO na liście cyklicznej 08

-------------------------------------

Rozbuduj program realizujący funkcjonalność kolejki FIFO na strukturze jednokierunkowej listy cyklicznej.

Wejście

Ciąg trzech komend: P, L, E n1,n2,...,nk, R n, rozdzielonych znakiem nowej linii.

Wyjście

Rezultaty użycia jednej z trzech dostępnych komend:

P – drukuj wszystkie elementy włożone do kolejki rozdzielone ciągiem znaków: "–>". Jeżeli kolejka jest pusta drukuj: "–" (już zaimplementowane).

L – drukuj liczbę elementów listy (już zaimplementowane).

E n1,n2,...,nk - wstaw k elementów o kluczu ni do kolejki, jeżeli nie ma miejsca powiększ listę dodając nowe, puste elementy.

R n - usuń element listy o zadanym kluczu, jeżeli nie istnieje wydrukuj: '-'.

Przykład

Wejście

P

L

R 5

P

L

E 7

P

L

R 7

P

L

E 4 -1 0 4

P

L

R 4

P

L

R -1

P

L

R 4

P

L

R 0

P

L

R 7

P

L

Wyjście

-

0

-

0

7

1

-

0

4->-1->0->4

4

-1->0->4

3

0->4

2

0

1

-

0

-

0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_