**Homework 1: Stack – Queue**

**A – Theory part:**

A1. Match each of the following application with an appropriate choice from the list of available data structures, including stack, queue and linked list.

|  |  |
| --- | --- |
| Applications | Data structures |
| Arithmetic expression evaluation | Stack |
| When a resource is shared among multiple consumers. | Linked list |
| Managing nested function calls | Stack |
| A grocery store decided that customers who come first will be served first | Queue |
| Print jobs submitted by users of the system | Queue |
| Reverse a string | Stack |
| Manage a sorted list of integers | Linked list |
| A grocery list ordered by the occurrence of the items in the store | Linked list |
| Represent a polynomial by storing its coefficients and exponents | Linked list |
| Perform arithmetic operations on long integers | Linked list |

A2. For each pseudo-code segments below, what output is displayed after it executes?

A picture containing text, font, line, number

Description automatically generated

Output:

9

8

7

6

5

4

3

2

1

A picture containing text, font, line, white

Description automatically generated

Output:

9

9

9

.... // Run forever

A3. Trace through the state of the stack S / queue Q in the following pseudo-code fragments.

|  |  |  |  |
| --- | --- | --- | --- |
| a. | Stack S; // A stack of string  push(S, "happy");  push(S, "sad");  string st = top(S);  push(S, "numb");  push(S, st+"dle");  pop(S);  st = top(S);  pop(S);  push(S, st); | S = {}  S = {happy}  S = {happy, sad}  S = {happy, sad}, st = “sad”  S = {happy, sad, numb}  S = {happy, sad, numb, sadle}  S = {happy, sad, numb}  S = {happy, sad, numb}, st = “numb”  S = {happy, sad}  S = {happy, sad, numb} | |
| b. | Queue Q; // A queue of integers  enqueue(Q, 5);  enqueue(Q, 7);  enqueue(Q, 13);  deQueue(Q);  int t = front(Q);  enqueue(Q, 12+t);  deQueue(Q);  enQueue(Q, front(Q));  deQueue(Q); | | Q = {}  Q = {5}  Q = {5, 7}  Q = {5, 7, 13}  Q = {7, 13}  Q = {7, 13}, t = 7  Q = {7, 13, 19}  Q = {13, 19}  Q = {13, 19, 13}  Q = {19, 13} |

A4. Consider the following sequences of letters.

(i) E A S \* Y \* Q U E \* \* \* S T \* \* \* I O \* N \* \* \*

(ii) L A \* S T I \* N \* F I R \* S T \* \* O U \* T \* \* \* \* \* \*

a. A letter means push and an asterisk means pop in the sequence, which is performed on an initially empty LIFO stack. Show the sequence of popped-out letters.

b. A letter means enqueue and an asterisk means dequeue in the following sequence, which is performed on an initially empty FIFO queue. Show the sequence of dequeued letters.

a.

(i) Popped: S, Y, E, U, Q, T, S, A, O, N, I, E

(ii) Popped: A, I, N, R, T, S, U, T, O, I, F, T, S

b.

(i) Popped: E, A, S, Y, Q, U, E, S, T, I, O, N

(ii) Popped: L, A, S, T, I, N, F, I, R, S, T, O, U

A5. Suppose that a client performs an intermixed sequence of enqueue and dequeue operations. The enqueue operations put the integers 0 through 9 in order onto the queue; the dequeue operations remove an element from the queue.

Following are sequences of integers that might involve in the removal. Which of those sequence(s) could not occur? Give your reasons.

a) 0123456789

b) 4687532901

c) 2567489310

d) 4321056789

a) 0123456789

- Queue:

Enqueue: 0, 1, 2, 3, 4 - [0, 1, 2, 3, 4]

Dequeue: 0, 1, 2, 3, 4 - []

Enqueue: 5, 6, 7, 8, 9 - [5, 6, 7, 8, 9]

Dequeue: 5, 6, 7, 8, 9 - []

🡪 This sequence could occur.

b) 4687532901

- Queue:

Enqueue: 4, 6, 8 - [4, 6, 8]

Dequeue: 4, 6, 8 - []

Enqueue: 7 - [7]

🡪 This sequence could not occur because the enqueue operations put the integers 0 through 9 in order onto the queue 7 < 8.

c) 2567489310

- Queue:

Enqueue: 2, 5, 6, 7 - [2, 5, 6, 7]

Dequeue: 2, 5, 6, 7 - []

Enqueue: 4 - [4]

🡪 Like b), this sequence could not occur because the enqueue operations put the integers 0 through 9 in order onto the queue 4 < 7.

d) 4321056789

- Queue:

Enqueue: 4 - [4]

Dequeue: 4 - []

Enqueue: 3 - [3]

🡪 Like b) and c), this sequence could not occur because the enqueue operations put the integers 0 through 9 in order onto the queue 3 < 4.

🡪 Sequences could not occur: b), c) and d).

A6. Suppose that a client performs an intermixed sequence of push and pop operations. The push operations put the integers 0 through 9 in order onto the stack; the pop operations remove an element from the stack.

The following are sequences of integers that might involve in the removal. Which of those sequence(s) could not occur? Give your reasons.

a) 4321098765 b) 4687532901 c) 2567489310 d) 4321056789

e) 1234569870 f) 0465381729 g) 1479865302 h) 2143658790

a) 4321098765

- Stack:

Push: 0, 1, 2, 3, 4 - [0, 1, 2, 3, 4]

Pop: 4, 3, 2, 1, 0 - []

Push: 5, 6, 7, 8, 9 - [5, 6, 7, 8, 9]

Pop: 9, 8, 7, 6, 5 - []

🡪 This sequence could occur

b) 4687532901

- Stack:

Push: 0, 1, 2, 3, 4 - [0, 1, 2, 3, 4]

Pop: 4 - [0, 1, 2, 3]

Push: 5, 6 - [0, 1, 2, 3, 5, 6]

Pop: 6 - [0, 1, 2, 3, 5]

Push: 7, 8 - [0, 1, 2, 3, 5, 7, 8]

Pop: 8, 7, 5, 3, 2 - [0, 1]

Push: 9 - [0, 1, 9]

Pop: 9 - [0, 1]

🡪 Can't pop 1 before 0 so this sequence couldn’t occur.

c) 2567489310

- Stack:

Push: 0, 1, 2 - [0, 1, 2]

Pop: 2 - [0, 1]

Push: 3, 4, 5 - [0, 1, 3, 4, 5]

Pop: 5 - [0, 1, 3, 4]

Push: 6 - [0, 1, 3, 4, 6]

Pop: 6 - [0, 1, 3, 4]

Push: 7 - [0, 1, 3, 4, 7]

Pop: 7, 4 - [0, 1, 3]

Push: 8 - [0, 1, 3, 8]

Pop: 8 - [0, 1, 3]

Push: 9 - [0, 1, 3, 9]

Pop: 9, 3, 1, 0 - []

🡪 This sequence could occur.

d) 4321056789

- Stack:

Push: 0, 1, 2, 3, 4 - [0, 1, 2, 3, 4]

Pop: 4, 3, 2, 1, 0 - []

Push: 5 - [5]

Pop: 5 - []

Push: 6 - [6]

Pop: 6 - []

Push: 7 - [7]

Pop: 7 - []

Push: 8 - [8]

Pop: 8 - []

Push: 9 - [9]

Pop: 9 - []

🡪 This sequence could occur.

e) 1234569870

- Stack:

Push: 0, 1 - [0, 1]

Pop: 1 - [0]

Push: 2 - [0, 2]

Pop: 2 - [0]

Push: 3 - [0, 3]

Pop: 3 - [0]

Push: 4 - [0, 4]

Pop: 4 - [0]

Push: 5 - [0, 5]

Pop: 5 - [0]

Push: 6 - [0, 6]

Pop: 6 - [0]

Push: 7, 8, 9 - [0, 7, 8, 9]

Pop: 9, 8, 7, 0 - []

🡪 This sequence could occur.

f) 0465381729

- Stack:

Push: 0 - [0]

Pop: 0 - []

Push: 1, 2, 3, 4 - [1, 2, 3, 4]

Pop: 4 - [1, 2, 3]

Push: 5, 6 - [1, 2, 3, 5, 6]

Pop: 6, 5, 3 - [1, 2]

Push: 7, 8 - [1, 2, 7, 8]

Pop: 8 - [1, 2, 7]

🡪 Can't pop 1 before 7 so this sequence couldn’t occur.

g) 1479865302

- Stack:

Push: 0, 1 - [0, 1]

Pop: 1 - [0]

Push: 2, 3, 4 - [0, 2, 3, 4]

Pop: 4 - [0, 2, 3]

Push: 5, 6, 7 - [0, 2, 3, 5, 6, 7]

Pop: 7 - [0, 2, 3, 5, 6]

Push: 8, 9 - [0, 2, 3, 5, 6, 8, 9]

Pop: 9, 8, 6, 5, 3 - [0, 2]

🡪 Can't pop 0 before 2 so this sequence couldn’t occur.

h) 2143658790

- Stack:

Push: 0, 1, 2 - [0, 1, 2]

Pop: 2, 1 - [0]

Push: 3, 4 - [0, 3, 4]

Pop: 4, 3 - [0]

Push: 5, 6 - [0, 5, 6]

Pop: 6, 5 - [0]

Push: 7, 8 - [0, 7, 8]

Pop: 8, 7 - [0]

Push: 9 - [0, 9]

Pop: 9, 0 - []

🡪 This sequence could occur.

🡪 Sequences couldn’t occur: b), f) and g)