Q1 A.

Start with an empty stack of integers. You will attempt to do a sequence of pushes and pops so that the sequence of pops will be a specified permutation of 1, 2, 3 ,4 , 5, 6. You will be able to do exactly 6 push operations and 6 pop operations. The first push pushes 1 onto the stack; the next pushes 2; and so forth. The sixth push pushes 6 onto the stack.

For this exercise, we will let S denote a push operation and X a pop operation. Example:

The sequence SSSSSSXXXXXX outputs 654321.

1. Describe a sequence of pushes and pops that would produce output 325641 (or explain why it is not possible)

**Ans:**

The output sequence 325641 cannot be generated using the given push (S) and pop (X) operations due to the Last In, First Out (LIFO) nature of a stack. For 3 to be removed before 2 and 5, 6 would need to be pushed earlier, which conflicts with the required output order, making the sequence invalid.

1. Describe a sequence of pushes and pops that would produce output 154623 (or explain why it is not possible)

**Ans:**

| **Step** | **Operation** | **Stack (Top → Bottom)** | **Output** |
| --- | --- | --- | --- |
| 1 | Push 1(S) | 1 |  |
| 2 | Push 2(S) | 2 1 |  |
| 3 | Push 3(S) | 3 2 1 |  |
| 4 | Pop(X) | 2 1 | 3 |
| 5 | Push 4(S) | 4 2 1 |  |
| 6 | Push 5(S) | 5 4 2 1 |  |
| 7 | Pop(X) | 4 2 1 | 5 |
| 8 | Pop(X) | 2 1 | 4 |
| 9 | Push 6(S) | 6 2 1 |  |
| 10 | Pop(X) | 2 1 | 6 |
| 11 | Pop(X) | 1 | 2 |
| 12 | Pop(X) |  | 1 |

**Operation Sequence**: S S S X S S X X S X X X

**Final Output**: 1 5 4 6 2 3

Q1 B.

Suppose we store n keys in a hash table of size m = n^2 using a hash function h randomly chosen from a Universal class H of hash functions. Assume that X is a random variable that counts the number of collisions. Show that the Expected number of Collisions is < 1/2.

Given:

* + m=n2 (size of the hash table).
  + Universal hash function h.

Let X be the random variable representing collisions. The expected number of collisions is calculated as follows:

E[X]= n(n−1)/2m

Substituting m=n2:

E[X]=n(n−1)/2n2  = n−1/2n

For n≥1, E[X] < 1/2. Hence, the expected number of Ecollisions is less than 1/2.

Q2.

For each integer n = 1,2,3,…, 7, determine whether there exists a red-black tree having exactly n nodes, with *all of them black.* Fill out the chart below to tabulate the results:

|  |  |
| --- | --- |
| Num Nodes n | If there exists a red-black tree with all nodes black |
| 1 | Yes |
| 2 | No |
| 3 | No |
| 4 | Yes |
| 5 | No |
| 6 | No |
| 7 | No |

Q3. For each integer n = 1,2,3,…, 7, determine whether there exists a red-black tree having exactly n nodes and exactly one red node*.* Fill out the chart below to tabulate the results:

|  |  |
| --- | --- |
| Num Nodes n | If there exists a red-black tree with exactly one red node |
| 1 | No |
| 2 | Yes |
| 3 | No |
| 4 | Yes |
| 5 | Yes |
| 6 | No |
| 7 | Yes |

Q4

Show the red black tree that results after each of the integer keys 21,32,64,75 and 15 inserted in that c into an initially empty red black tree. Clearly show the tree that results after each insertion (indicating the col each node), and make clear any rotations that must be performed

To solve Question 4, we will simulate the step-by-step insertion of the keys into an initially empty red-black tree while ensuring the tree adheres to red-black tree properties:

1. Every node is either red or black.
2. The root is always black.
3. Red nodes cannot have red children (no two consecutive red nodes).
4. Every path from a node to its descendant NULL nodes has the same number of black nodes.
5. When a violation occurs, rotations and/or recoloring are applied.
6. Insert 21, which becomes the root which is always black 21(Black)
7. Insert 32, to the right of 21. New nodes are red by default. No property violations.

21(Black)

32(Red)

1. Insert 64 to the right of 32, new nodes are red by default. It violates the property check because of two consecutive red nodes(32 and 64). Performing a resolution, and rotation: Left rotation at 21, recolor 32 to black.

32(Black)

21(Red) 64(Red)

1. Insert 75, added to the right of 64. New nodes are red. Violation because of two consecutive reds. Recolor 64 and 21 to black. Recolor 32 to red (if necessary, but in this case, the root remains black to maintain balance).

32(Black)

21

(

Black

**)**

64

(

Black

**)**

75(Red)

1. Insert 15, its by default red, and you insert it to the left of 21/

32(Black)

21

(

Black

**)**

64

(

Black

**)**

15(Red) 75(Red)