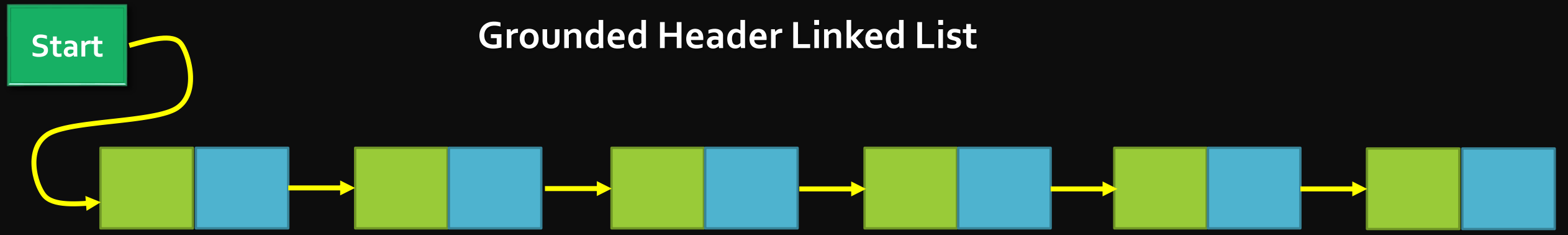
Data Structure (CS-303)

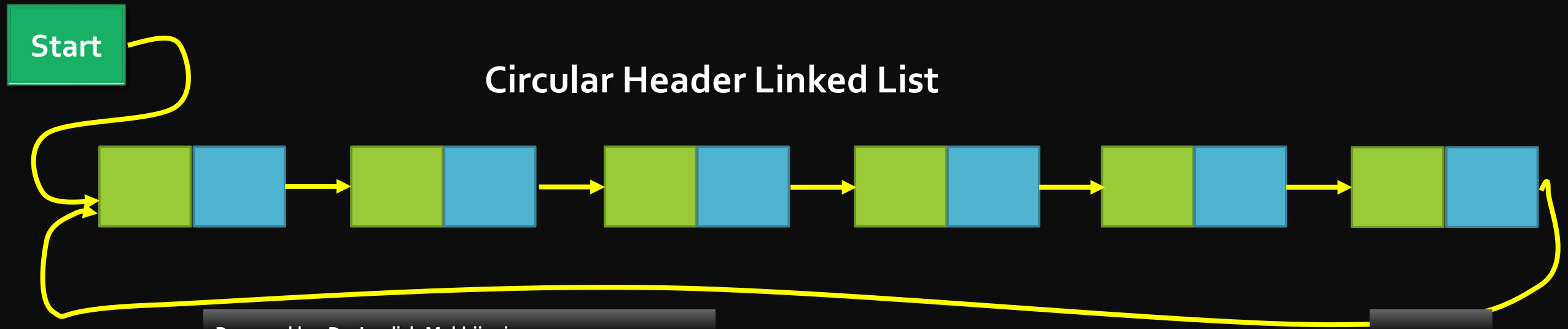
Circular Linked List

CIRCULAR LINKED LIST

Grounded Header Linked List



Circular Header Linked List



CIRCULAR LINKED LIST

A header linked list is a linked list which always contains a special node, called the header node, at the beginning of the list.

1. A grounded header list a header list where the last node contains the null pointer.
2. A circular header list is a header list where the last node points back to the header node.

CIRCULAR LINKED LIST TRAVERSAL

TRAVERSE(START)

1. Set PTR := LINK[START] [Initialized the pointer PTR]
2. Repeat Steps 3 and 4 while PTR ≠ START:
3. Apply PROCESS to INFO[PTR]
4. Set PTR := LINK [PTR] [PTR now points to the next node]

   [End of Step 2 loop]
5. Exit

CIRCULAR LINKED LIST SEARCH

SEARCHCHLL(INFO, LINK, START, ITEM, LOC)

1. Set PTR := LINK[START]
2. Repeat while INFO[PTR] ≠ ITEM and PTR ≠ START:
   Set PTR := LINK[PTR] [PTR now points to the next node]
   [End of loop]
3. If INFO[PTR] = ITEM, then:
   Set LOC := PTR
   Else:
   Set LOC:= NULL
   [End of If structure]
4. Exit

CIRCULAR LINKED LIST INSERTION AT BEGINNING

```
1.      IF AVAIL = NULL
                Write OVERFLOW
                Go to Step 11
        [END OF IF]
2.      SET NEW_NODE= AVAIL
3.      SET AVAIL = AVAIL→ NEXT
4.      SET NEW_NODE → DATA = VAL
5.      SET PTR = START
6.      Repeat Step 7 while PTR → NEXT != START
7.              PTR = PTR → NEXT
        [END OF LOOP]
8.      SET NEW_NODE → NEXT = START
9.      SET PTR → NEXT = NEW_NODE
10.     SET START = NEW_NODE
11.     EXIT
```

CIRCULAR LINKED LIST INSERTION AT END

```
1.      IF AVAIL = NULL
                Write OVERFLOW
                Go to Step 10
        [END OF IF]
2.      SET NEW_NODE= AVAIL
3.      SET AVAIL = AVAIL→ NEXT
4.      SET NEW_NODE → DATA = VAL
5.      SET NEW_NODE → NEXT = START
6.      SET PTR = START
7.      Repeat Step 8 while PTR → NEXT != START
8.              PTR = PTR → NEXT
        [END OF LOOP]
9.      SET PTR → NEXT = NEW_NODE
10.     EXIT
```

CIRCULAR LINKED LIST DELETE FIRST NODE

1. IF START = NULL
   Write UNDERFLOW
   Go to Step 8
   [END OF IF]
2. SET PTR = START
3. Repeat Step 4 while PTR → NEXT != START
4. SET PTR = PTR → NEXT
   [END OF LOOP]
5. SET PTR → NEXT = START → NEXT
6. FREE START
7. SET START = PTR → NEXT
8. EXIT

CIRCULAR LINKED LIST
DELETE THE LAST NODE

1. IF START = NULL
   Write UNDERFLOW
   Go to Step 8
   [END OF IF]
2. SET PTR = START
3. Repeat Steps 4 and 5 while PTR → NEXT != START
4. SET PREPTR = PTR
5. SET PTR = PTR → NEXT
   [END OF LOOP]
6. SET PREPTR → NEXT = START
7. FREE PTR
8. EXIT