Nama : TIARA YEMELDA

NPM : G1F021008

1. Write the algorithm of queue mechanism using
   * Single linked list
   * Array alternative 1
   * Array alternative 2
   * Array alternative 3
2. Use the same infotype as before
3. Each member is to write 1 mechanism

Jawab :

* Single linked list Algoritma:
  + Simpan 2 reference: front → … → … →back
  + enqueue(Benda x):
    - Buat sebuah node baru N yang datanya x
    - if queue sebelumnya empty, maka front = back = N
    - else tambahkan N di akhir (dan update back)
  + dequeue():
    - Hapus elemen pertama: front = front.next
* Array alternative 1 Algoritma :

Add(P,3)

Add(P,4)

Add(P,2)

Del(P)

Del(P) Add(P,5)

Del(P)

Del(P)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 3 | 4 | 2 |  |  |

Head = 1

Tail = 3

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 2 |  |  |  |  |

Head = 1

Tail = 0

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 5 | 2 |  |  |  |

Head = 1

Tail = 2

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
|  |  |  |  |  |

Head = 0

Tail = 0

Is empty = False

* Array Alternative 2 Algoritma:

Add(P,3)

Add(P,4)

Add(P,2)

Del(P)

Del(P) Add(P,5)

Del(P) Add(P,6)

Add(P,7)

Del(P)

Del(P)

Del(P)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 3 | 4 | 2 |  |  |

Head = 1

Tail = 3

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 2 |  |  |  |  |

Head = 1

Tail = 0

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 5 | 2 |  |  |  |

Head = 1

Tail = 2

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 2 |  |  |  |  |

Head = 1

Tail = 0

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 7 | 6 | 2 |  |  |

Head = 1

Tail = 3

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |

Head = 0

Tail = 0

Is empty = False

* Array alternative 3 Algoritma:

Add(P,3)

Add(P,4)

Add(P,2)

Del(P)

Del(P) Add(P,5)

Del(P) Add(P,6)

Add(P,7)

Add(P,8)

Del(P)

Del(P)

Del(P)

Del(P)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 3 | 4 | 2 |  |  |

Head = 1

Tail = 3

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 2 |  |  |  |  |

Head = 1

Tail = 0

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 5 | 2 |  |  |  |

Head = 1

Tail = 2

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 2 |  |  |  |  |

Head = 1

Tail = 0

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 8 | 7 | 6 | 2 |  |

Head = 1

Tail = 3

Is empty = True

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
|  |  |  |  |  |

Head = 0

Tail = 0

Is empty = False