NAMA   : Muhammad Rafi' Abdurrahman
NIM   : 18224004

Latihan Soal Queue dengan Struktur Berkait

```c
1    #include "prioqueuelist.h"
2    #include <stdio.h>
3
4    Address newNode(ElType x, int pr){
5    Address new = malloc(sizeof(Node));
6    if(new == NULL) return Nil;
7    INFO(new) = x;
8    PRIO(new) = pr;
9    NEXT(new) = Nil;
10   return new;
11   }
12
13   void delNode(Address *P){
14   free(*P);
15   }
16
17   boolean isEmpty(PrioQueue q){
18   return q.addrHead == Nil; //Cuma ada head
19   }
20
21   int length(PrioQueue q){
22   int count = 0;
23   Address traversal = ADDR_HEAD(q);
24   while(traversal != Nil){
25   count++;
26   traversal = NEXT(traversal);
27   }
28   return count;
29   }
30
31   void CreateQueue(PrioQueue *q){
32   ADDR_HEAD(*q) = Nil;
33   }
34
35   void enqueue(PrioQueue *q, ElType x, int pr){
36   Address new = newNode(x,pr);
37   if(new == Nil) return;
38
39   if(isEmpty(*q)){
40   ADDR_HEAD(*q) = new;
41   }else{
42   Address traversal = ADDR_HEAD(*q);
43   Address prev = Nil;
44   while(PRIO(new) >= PRIO(traversal) && NEXT(traversal) != Nil){
45   printf("traversal %d\n", INFO(traversal));
46   prev = traversal;
```

```
47    traversal = NEXT(traversal);
48    }
49

50    if (PRIO(new) >= PRIO(traversal) && NEXT(traversal) == Nil){ //New
node prio terbesar
51    NEXT(traversal) = new;
52    }
53    else if(prev == Nil){ //Q punya satu elemen saja / new Node terkecil
54    NEXT(new) = traversal;
55    ADDR_HEAD(*q) = new;
56    }else{ //New node di tengah
57    NEXT(prev) = new;
58    NEXT(new) = traversal;
59    }
60
61    }
62    }
63
64    void dequeue(PrioQueue *q, ElType *x, int *pr){
65    Address temp = ADDR_HEAD(*q);
66    if(NEXT(temp) == Nil){
67    ADDR_HEAD(*q) = Nil;
68    }else{
69    ADDR_HEAD(*q) = NEXT(temp);
70    }
71
72    *x = INFO(temp);
73    *pr = PRIO(temp);
74    delNode(&temp);
75    }
76
```