Type Pasien:<Nama, klinik:string, usia: integer>
Type infotype: Pasien
Type adr: pointer to elem
Type elem:<info: infotype, next, prev: adr>
Type queue:<Head, Tail: adr>

2A Procedure Dequeue(in/out Q:queue, out P: adr)

Kamus
function isEmpty(queue) -> boolean

Algoritma
P = Q.Head
if isEmpty(Q) then
output ("Queue kosong")
else if Q.Head == Q.Tail then
Q.Head = NIL
Q.Tail = NIL
else
Q.Head = P -> next
P -> next = NIL
Q.Head -> prev = NIL

```
  endif
endprocedure
```

2.C

```
Procedure pindahTestCovid (in/out QReg, QCovid:
queue)
Kamus
  function isEmpty(queue) -> boolean
  P: adr
  procedure Enqueue(queue, adr)
  procedure Dequeue(queue, adr)
  function size(queue) -> integer
Algoritma
  if isEmpty(QCovid) then
    while not isEmpty(QReg) and size(QCovid)
    != 3 do
      Dnqueue(QReg, P)
      Enqueue(QCovid, P)
    endwhile
  else
    output("pemindahan belum bisa dilakukan")
  endif
endprocedure
```