(duenes 1. Linear Quene: # include < stdio. h> # define quene [N]: De l'+ jordison int front =- 1, rear: -1; void enquenelinta) if (rear = = N-1) Printf ("overflow"); ehe { quenelrear]: n; void degne () if (front == -1 && rear == -1) prints ("Vonderflow"): 3/2/se [x+4]: 00 priorité ("Y. J." quenéfront); front + +) void display() I Printf ("Underflow");

1 prints ("Quene Contains"): for lint i= front; ix rear +1; i++) doubtes about my # 1 prints ("/.", quene (i)); But grane [5] . void enginene (total) or front = - 1 kt rech = int main () grane grante or enquene (10)3 enque ne (20); enquene (30)) grant (" Breene is full"); display Quene(1; int degned = dequenels; prints ("Degred item: 1.d", degred); display anenets: que ve [rest]: x. return o Output: grene cordains. month Curdifley 20 30 Queve contains front rear - 19: 30 print[(".(.d., 9[frost]); out - front + 1) " Wil.

08/01/2021 2. Circular Quenes: Powers Contonne # include (statio.h) # define N=5 int grene[5]; int front=-1, rear=-1; void enquene (intr) for (front = = -1 && rear = = -1) { front = rear = 0; quene (rear) = x; E(a) enouspone clse if ((reax +1)4.N=-front) prints (" Brene is full"); Josephy Grene (1) int degreed = degreenel); real = (real + 1)=/oN; by (on the burger) thing display arone (); I void dequeue () 2 prints ("underflow"); 01 Queux contoriors { front=rear = -18; printf(".r.d", q[front]); foront= (foront +1) 1/2 N;

3

void display () ? prints ("underflow"); print ["Quene Contains");

int i= front;

while (i!= rear) else { 2 Prints (" t.d", quere(i)); i= (i+1)-/0N; prints ("+.d", quene(rear)); int main() ¿ enqueve(10); enquene(30); display Quenel); int dequed=dequene(); prints (" Dequed item; "/, d; dequed);

display Quesel); eneque (40); eneque (50); display Brane: return o', Cutbrit. Quene Contains Quercontains. 30 Quene contains 30