D O conceito TAD (tipo Abstrato de Dada) é um conjunto de volores de funções Na linguagem C é feits crionds um orquis in e un c ende me in fica e cobeçalhe das dunções e estruturas e ma . a comtem a implementação das tunções que estos declarados no aquivo to.

· Vontagens:

- Encopsulamento: o encopsulamento ajuda a ocultar a complexidade interna e protege es dados centro ocesa más antirizados e dambem e Dom pora rentiliza funçãos e estruturas em outros orquires

- Monutenção: conseguimos obteror o TDA sem obteror as optimos que

o utilizom

O TAD é realizado em orquiros diderentes, pou tica facil para a monutanco, organização o cádigo fico mais limpo e facil de entender. Abstração entrando os dodos o máximo passível, e rentilização do cádigo

(2) FO: 5,7,12,4,0,4,6,8,67,34,23,5,0,44,33,22,6,0

: epibes a ciga

F1 : } 12, 4, 0, 4, 6, 8, 34, 0, 44, 22, 6, 0}

Fz: 15,7,67,23,5,33 }

i) ex = create - pilha (), po = create - pilho(), temp = create_pilho(), preencher-piha (PX); preencher-pilha (PO); (1) enguento iros empty-pilha (PX) tozer i(xq)qrq = x

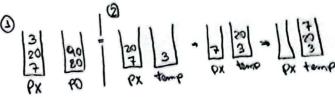
push (x, temp),

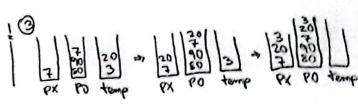
fim-tozer enquanto mão empty-pilho (temp) fozer X = bob (famb);

push (x, po); push (x, px),

1: m-toser

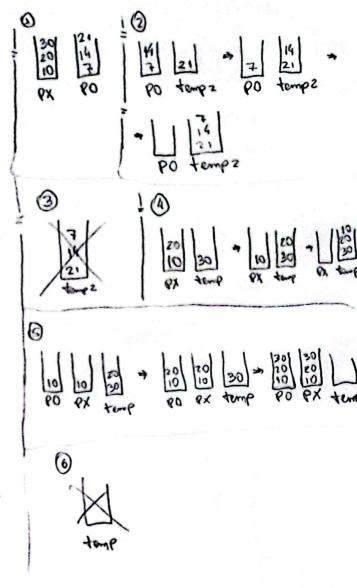
Destruir (temp), (1)







Die) PX = crede - Pilha (); Po = Create - pilhal), temp = create_pilho(), bisorcher - billy (bx); } @ preencher-pilho (PO). temp 2 = create - pilhou) enquanto não empty-piha (10) forer X= POP (PD), } @ fim-fozer gestemi (temp2), 3 enquanto más emply pilha (Px) fozer x = pop(Px), push (x, temp), enquanto não empty-pilho (temp) tozer Bur(x, ex) push (X, PO) fim-fozor destruir (temp))



combinar Lista (L1, L2) = L3 = NULL ; cudrongo (17 = MATT 6 TSIMOT) foses ectus (escop < 13 > epopor 17) 33 imserir (13, L1 = dods), [] = [] = prexime; Romas Inserir (L3, L2 + dado), TS = TS - bioximo; fim-se Lion-fozer enquanto (IJI=NULL) fozer Imserir (L3, L1 - dods), [7=[7 -> broxims, tim-fozer endroups (TS |= NATT) tosa imzrir (L3, L2 = dode), [5 = Fs = broximo] lim-tozer (1) combinantista (41/+12), + 13 = NULL , + temp: enquento (L1 vozio e L21 vozis) foxor 00 (L1 - 000 - L2 - 000) entos (epop-77 (F3) 1, 2000) goub = 17 = bigime; free (L1), L1 = temp 1 insenir (L3, L2 = dods) temp=12 - proxima i free (L2). 12 = temp, Lim se fim lover

LI: 100 10 10 10 10 LI: [M [] - [D [] - [W] 0 L3: [] (7:10日)10日。1000 ② 12:回母,回母。回 L3 回叶口 ③ 山:四田一四日十四 心:自中国于 回 13:回子回中口 L2: [2] 17 (3: [II] (II) @ 11: OB MOD " 四面 回回 :11 1、面面面面面面面面。[1 ⑥山: 回野国界台 मिण्या सिकाः (3:回时)国中日 a continuocão enguento (LI I=NULL) fozur Inserir (L3, L1 - dods) temp = L1 - proximo fire (Ls); LJ = temp, film tozer enquanto (LZI=NUL) toeco Inserir (L3, L2-dods). temp = L2 - praximo, free(LZ). L2 = temp

fim dozer

$$\begin{cases} X = P \rightarrow seq \\ Q = seq = x \\ Q \rightarrow ont = P \\ P \rightarrow seq = q \\ X \rightarrow ont = q \end{cases}$$

1