

$$\begin{array}{r} 58387 \quad \overline{15} \\ 11677 \end{array}$$

2

$$\begin{array}{r} 11677 \quad \overline{15} \\ 2337 \end{array}$$

2

$$\begin{array}{r} 2335 \quad \overline{15} \\ 0 \quad 467 \end{array}$$

$$\begin{array}{r} 467 \quad \overline{15} \\ 2 \quad 93 \end{array}$$

$$\begin{array}{r} 93 \quad \overline{15} \\ 3 \quad 18 \end{array}$$

$$\begin{array}{r} 18 \quad \overline{15} \\ 3 \quad 3 \end{array}$$

$$\begin{array}{r} 3 \quad \overline{15} \\ 3 \quad 0 \end{array}$$

$$\begin{aligned} x &= 3 \\ x &= 3 \times 10 + 3 & (33) \\ x &= 33 \times 10 + 3 & (333) \\ x &= 333 \times 10 + 2 & (3332) \\ x &= 3332 \times 10 + 0 & (33320) \\ x &= 33320 \times 10 + 2 & (333202) \\ x &= 333202 \times 10 + 2 & (3332022) \end{aligned}$$

$$\begin{array}{r} 3 \times \\ \hline 3 \times \\ \hline 3 \times \\ \hline 2 \times \\ \hline 0 \times \\ \hline 2 \times \\ \hline 2 \times \end{array}$$



EINF MA CID

Disciplina: Estruturas de Dados e Algoritmos I-2023/2024

Prova: teste 1 (15-04-2024)

Esta prova tem a duração de 2 horas e é sem consulta. Identifique TODAS as folhas de teste.

1. Considere definida em C, a função:

```

31 unsigned long q1(int n, int b) {
32     Stack s = CreateStack(100);
33     while (n != 0) {
34         Push(n % b, s);
35         n = n / b;
36     }
37     unsigned long x = Pop(s);
38     while (!IsEmptyStack(s)) {
39         x = x * 10 + Pop(s);
40     }
41     return x;
42 }

```

- (a) Qual o retorno de $q1(n, 5)$, para n o seu número de aluno?
 $q1(n, 5) = 3332022$
- (b) Desenhe a stack utilizada na função correspondente à linha 37. Indique claramente qual o topo da stack.
 $\boxed{3} \boxed{3} \boxed{3} \boxed{2} \boxed{0} \boxed{2} \boxed{2}$ ← Topo
- (c) Qual a complexidade da função $q1$? $O(n)$
- (d) Qual o retorno de $q1(10, 1)$? Justifique. Não terá retorno pois constitui um ciclo infinito.

2. Considere definida em C, a função:

```

46 void q2(int n, Queue Q) {
47     Enqueue(1, Q);
48     for (int i = 0; i < n; i++) {
49         Enqueue(Front(Q) * 10 + 0, Q);
50         Enqueue(Front(Q) * 10 + 1, Q);
51         printf("%d\n", Dequeue(Q));
52     }
53 }
54 }
55 }
56 }
57 }

```

- (a) Seja Q , uma Queue de inteiros inicialmente vazia. Qual o output produzido pela chamada $q2(5, Q)$?
 $10, 11, 100, 101$
- (b) Qual o conteúdo da Queue Q após a chamada realizada na alínea 2a? Indique o início e o fim da fila. $Q = [110, 111, 1000, 1001, 1010, 1011]$
- (c) Qual a complexidade da função $q2$?
 $O(n)$
- (d) Discuta a complexidade espacial da função $q2$.
 $O(n)$ derivado do 'for'

$1, 10, 11, 100, 101, 110, 111, 1000, 1001, 1010, 1011$

$1, 10, 11, 100, 101$

```

1  int q3(List L, int k, int m)
2  Position p=First(L);
3  List R=CreateList(NULL);
4  int kma;
5  while (p!=NULL && k>0) { while (P!=NULL
6      p=Advance(P); P=Advance(P)
7      k--; k--;
8  }
9
10 Position p=First(R);
11 for (int i=a; i<=b; i++, Pr=Advance(Pr)) { for (int i=a; i<=b; i++, *)
12     if (P==NULL) { if (P==NULL) {
13         return R; return R;
14         *) P=Advance(P), Pr=Advance(Pr);
15     }
16     else else
17         Insert(Retrieve(P), Pr, R); Insert(Retrieve(P), R, R);
18 }
19 return R; return R;
20
21 }
22 }

```

Seja L a lista L={1;2;3;4;5;6;7;8;9;10}.

- Considere a chamada $q3(L, 3, 8)$, qual a lista retornada pela função?
- Qual o retorno de $q3(L, 7, 20)$?
- Qual o retorno de $q3(q3(L, 2, 6), 0, 3)$?
- Qual a complexidade da função $q3$? $O(n)$

1. Uma *Deque* é uma estrutura de dados que consiste numa lista de itens, e permite as seguintes operações:

- $push(x, D)$: que insere o item x, no início da deque D
- $pop(D)$: que retira da deque o elemento que está no início e devolve-o
- $inject(x, D)$: que insere o item x, no fim da deque D
- $eject(D)$: que retira da deque o elemento que está no fim e devolve-o

(a) Por forma a implementar as deque de forma eficiente vamos usar listas duplamente ligadas e circulares, para ser possível aceder ao fim da deque em tempo constante. Como é já habitual uma *Deque* é um apontador para um nó duplo, sendo a definição destes o habitual (Element, Next e Prev).

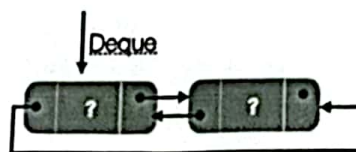


Figure 1: Deque vazia, com nós sentinelas

- À semelhança do que foi realizado nas aulas práticas apresente a implementação da função `Deque CreateDeque(void)`, que retorna uma deque vazia, construída como na figura 1, com um nó sentinelas no início e outro no fim. A referência `prev` do nó do início aponta para o nó do fim
- Implemente uma das funções descritas para a Deque (1,2 3 ou 4), consoante o seu resultado de $f(x) = (x\%4) + 1$ e x é o seu número de aluno.