

UENF

Universidade Estadual do Norte Fluminense Darcy Ribeiro

Curso: Ciência de Computação

Data: 15/04/2021

Atividade: Questões Módulo 2 – AM2

Período: 3º

Disciplina: Estrutura de Dados I

Professor: Fermín Alfredo Tang

Turno: Diurno

Nome do aluno:

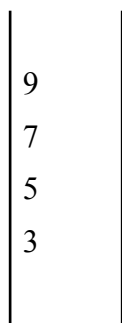
Matrícula:

Questões para AM2

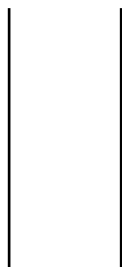
Considerando os conceitos sobre Pilhas:

1. Considere que temos duas pilhas de inteiros vazias, s1 e s2, mostre em uma figura a evolução de cada pilha após as seguintes operações:

```
1 pushstack (s1, 3)
2 pushstack (s1, 5)
3 pushstack (s1, 7)
4 pushstack (s1, 9)
5 pushstack (s1, 11)
6 pushstack (s1, 13)
7 loop not emptystack (s1)
  1 popstack (s1, x)
  2 pushstack (s2, x)
8 end loop
```



S1



S2

2. Considere que temos duas pilhas de inteiros, s1 e s2, mostre em uma figura a evolução de cada pilha após as seguintes operações:

```

1 pushStack (s1, 3)
2 pushStack (s1, 5)
3 pushStack (s1, 7)
4 pushStack (s1, 9)
5 pushStack (s1, 11)
6 pushStack (s1, 13)
7 loop not emptyStack (s1)
  1 popStack (s1, x)
  2 popStack (s1, x)
  3 pushStack (s2, x)
8 end loop

```

3. Use a **transformação manual** para transformar as seguintes expressões infixas nas suas respectivas formas posfixas e prefixas:

- $D - B + C$
- $A * B + C * D$
- $(A + B) * C - D * F + C$
- $(A - 2 * (B + C) - D * E) * F$

4. Mude as expressões infixas da questão anterior em expressões posfixas usando o **método algorítmico** com base em uma pilha. Ilustre passo a evolução da pilha.

C. $AB+C*DF*-C+$ Posfixo

5. Pense em um método algoritmo para transformar a expressão em prefixa.

C. $+-*+ABC*DFC$ Prefixa

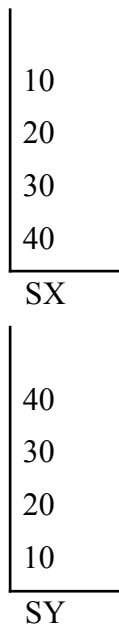
6. Determine o valor das seguintes expressões posfixas, considerando que as variáveis possuem os seguintes valores: $A=2$, $B=3$, $C=4$ e $D=5$. Ilustre o procedimento usando uma pilha para armazenar os valores intermediários.

- $AB * C - D +$
- $ABC + * D -$
- $ABCD * - +$
- $DC * BA + -$

7. Desenvolva um algoritmo para inverter o conteúdo de uma pilha. *Dica.* Use pilhas temporárias para armazenar o conteúdo da pilha original e fazer a inversão. Ilustre com exemplos.

40
30
20
10

S1



2. Desenvolva um algoritmo para testar se duas pilhas P1 e P2 são iguais. Duas pilhas são iguais se possuem os mesmos elementos, exatamente na mesma ordem. Considere que as pilhas P1 e P2 já existem e são passadas como parâmetro. Considere que as pilhas possuem elementos do tipo inteiro. Caso seja necessário, criar pilhas auxiliares. Ilustre com exemplos.
3. Desenvolva uma função chamada *copystack()* que copie o conteúdo de uma pilha em outra. A função deve receber duas pilhas como parâmetros: P1, a pilha origem e P2, a pilha destino. A pilha destino terá seu conteúdo substituído. A ordem dos dados na pilha origem deve ser preservada. *Dica.* Use uma pilha auxiliar para armazenar o conteúdo da pilha origem. Ilustre com exemplos.
4. Desenvolva uma função chamada *catstack()* que concatene (anexar) o conteúdo de uma pilha no topo da outra. A função deve receber duas pilhas como parâmetros: P1, a pilha origem e P2, a pilha destino. A ordem dos dados na pilha origem deve ser preservada. *Dica.* Use uma pilha auxiliar para armazenar o conteúdo da pilha origem. Ilustre com exemplos.

Considerando os conceitos sobre Filas:

5. Imagine que temos uma pilha de inteiros, S, e uma fila de inteiros, Q. Desenhe uma figura de S e Q após as seguintes operações:

```

1 pushStack (S, 3)
2 pushStack (S, 12)
3 enqueue (Q, 5)
4 enqueue (Q, 8)
5 popStack (S, x)
6 pushStack (S, 2)
7 enqueue (Q, x)
8 dequeue (Q, y)
9 pushStack (S, x)
10 pushStack (S, y)

```

6. Qual seria o conteúdo da fila Q, após o seguinte código ser executado:

```

1 Q = createqueue
2 loop (not end of file)
  1 read number
  2 if (number not 0)
    1 enqueue (Q, number)
  3 else
    1 x = queuerear (Q)
    2 enqueue (Q, x)
  4 end if
3 end loop

```

Considerando que a seguinte sequência de dados é ingressada:

5, 7, 12, 4, 0, 4, 6, 8, 67, 34, 23, 5, 0, 44, 33, 22, 6, 0.

7. Qual seria o conteúdo das filas Q1 e Q2 após o seguinte código ser executado:

```

1 Q1 = createqueue
2 Q2 = createqueue
3 loop (not end of file)
  1 read number
  2 enqueue (Q1, number)
  3 enqueue (Q2, number)
  4 loop (not empty Q1)
    1 dequeue (Q1, x)
    2 enqueue (Q2, x)
  5 end loop
4 end loop

```

Considerando que a seguinte sequência de dados é ingressada:

5, 7, 12, 4, 0, 4, 6.

8. Qual seria o conteúdo da fila Q1 após o seguinte código ser executado:

```

1 Q1 = createqueue
2 S1 = createstack
3 loop (not end of file)
  1 read number
  2 if (number not 0)
    1 pushstack (S1, number)
  3 else
    1 popstack (S1, x)
    2 popstack (S1, x)
    3 loop (not empty S1)
      1 popstack (S1, x)
      2 enqueue (Q1, x)
    4 end loop
  4 end if
4 end loop

```

Considerando que, a seguinte sequência de dados é ingressada:

5, 7, 12, 4, 0, 4, 6, 8, 67, 34, 23, 5, 0, 44, 33, 22, 6, 0.

9. Imagine que o conteúdo das filas Q1 e Q2 é o seguinte:

Q1: 42 30 41 31 19 20 25 14 10 11 12 15

Q2: 4 5 4 10 13

onde, o início da fila encontra-se no extremo esquerdo e o fim da fila encontra-se no extremo direito. Qual seria o conteúdo da fila Q3 após a execução do seguinte código.

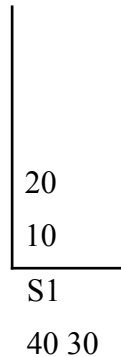
```

1 Q3 = createqueue
2 count = 0
3 loop (not empty Q1 and not empty Q2)
  1 count = count + 1
  2 dequeue (Q1, x)
  3 dequeue (Q2, y)
  4 if (y equal count)
    1 enqueue (Q3, x)
  5 end if
1 end loop

```

10. Dada uma fila de inteiros, escreva um algoritmo que, usando somente as funções do TAD Fila (Queue ADT), calcule e imprima o soma e a média dos inteiros na fila e preserve o conteúdo da fila.
- 10 20 30 40 50
11. Usando somente as funções do TAD Fila (Queue ADT), escreva o algoritmo de uma função que compare o conteúdo de duas filas e retorne True, caso as filas sejam idênticas; e False caso contrário.
12. Usando somente as funções do TAD Fila (Queue ADT), escreva o algoritmo de uma função chamada copyQueue() para copiar o conteúdo de uma fila origem em outra destino. Considere as seguintes situações: i) a fila origem não precisa ser preservada após a cópia; ii) a fila origem precisa ser preservada após a cópia.

13. Escreva um algoritmo para uma função chamada *stackToQueue()* que crie uma fila destino a partir de uma pilha origem. Após criar a fila, o topo da pilha deverá corresponder ao início da fila e a base da pilha deverá corresponder ao fim da fila. A pilha deverá ficar vazia após a execução da função.



14. Dada uma fila de inteiros, escreva um algoritmo que, usando somente as funções do TAD Fila (Queue ADT), elimine todos os inteiros negativos presentes na fila e preserve o conteúdo restante da fila sem mudar a ordem dos elementos.

40 50 -5 30 -7 80 10 20

Considerando os conceitos sobre Listas:

15. Imagine que temos as duas listas, como mostradas na Figura 1.

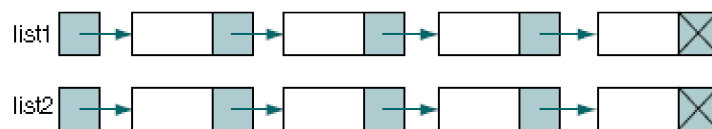


Figura 1.

Responda que acontece se aplicamos as seguinte sentença:

```
list1 = list2
```

16. Imagine que temos a lista mostrada na Figura 2, implementada como uma lista encadeada.

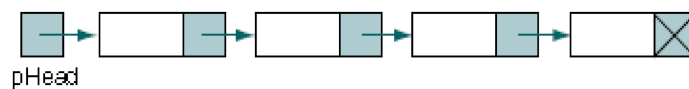


Figura 2.

Mostre que aconteceria se usássemos a seguinte sentença em uma busca percorrendo a lista?.

```
pHead = pHead->link
```

Isso, justifica o uso de dois ponteiros *pPre* e *PLoc* como descritos em aula?

17. Imagine que temos uma lista como a indicada na Figura 3.

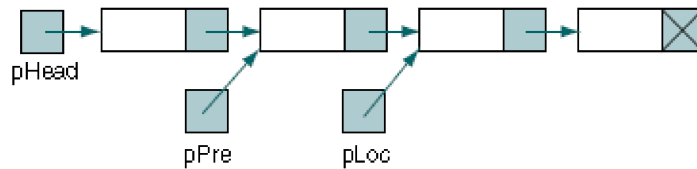


Figura 3.

Onde, os ponteiros pPre e pLoc são usados para percorrer a lista durante uma busca e apontam para o nó predecessor e o nó atual da busca. Ao percorrer a lista devemos avançar atualizando pPre e pLoc. O seguinte código pretende definir a atualização mas tem um erro. Qual é esse erro e como pode ser corrigido?.

```
pLoc = pLoc->link
pPre = pPre->link
```

```
Ppre=NULL
Ploc = pHead
Repete
```

```
if ploc <> NULL
    ploc = ploc ->link
    ppre = ploc
```

```
Ate ploc = NULL
```

18. Escreva um algoritmo que misture duas listas ordenadas em uma única lista usando o TAD Lista.

2 4 6 8 10

1 3 5 7 9

1 2 3

19. Escreva um algoritmo que crie uma matriz bidimensional usando listas encadeadas. Os nós na primeira coluna contém apenas ponteiros. O ponteiro esquerdo aponta para próxima linha. O ponteiro direito aponta para os dados de uma determinada fila. Observe a figura.

