



Disciplina: Estruturas de dados

Professor: Taniro Rodrigues

Aluno: _____

Matrícula: _____

Data: 27 de março de 2018

Nota: _____

Primeira Avaliação

OBSERVAÇÕES:

a) Assine a prova e as folhas de rascunho.

b) A avaliação é individual e sem consulta.

c) Serão aceitas respostas de caneta azul ou preta.

d) Desligue o telefone celular antes da prova. Não manuseie o telefone celular durante a prova (mesmo que desligado).

QUESTÃO 1: (ENADE 2014, Adaptada) Uma estrutura de dados do tipo pilha pode ser usada em um algoritmo que permite imprimir uma palavra de forma invertida. Por exemplo, ABACAXI deve ser impresso IXACABA.

Utilizando as variáveis declaradas abaixo como variáveis globais,

```
#DEFINE TAM 10
```

```
char pilha[TAM];
```

```
int topo;
```

```
char palavra[TAM];
```

Faça o que se pede:

- i) Desenvolva a função push que inclui um elemento na pilha. (valor 0,75)
- ii) Desenvolva a função pop que retira um elemento da pilha. (valor 0,75)
- iii) Desenvolva a função inverte que leia a palavra da entrada padrão (scanf) e, utilizando as funções push e pop da pilha, imprima a palavra de forma invertida (valor 1,0)

Dica: utilize as seguintes assinaturas de funções:

```
void push(char dado);
```

```
char pop ();
```

```
void inverte ();
```

QUESTÃO 2: (ENADE 2017, Adaptada) Uma empresa trabalha na produção de concreto e terceiriza o serviço de transporte do produto. Os caminhoneiros telefonam para a empresa e registram seu interesse pelo trabalho. Todas as manhãs, os caminhoneiros estacionam no pátio da empresa e aguardam sua vez. O atendimento segue o critério de ordem de chegada. Esse processo é, atualmente, controlado pela secretária, que utiliza sua agenda para gerenciar os motoristas diariamente. A empresa, que carrega, no máximo, 10 caminhões por dia, pretende informatizar esse processo. Para a solução do problema, apresenta-se, a seguir, um código em linguagem C que utiliza o conceito de fila, mantendo os elementos sempre nas primeiras posições do vetor.

```
#define TAM 10

int caminhoneiro_id[TAM];
int total;

void inicializa(){
    total = 0;
    return;
}

int empty(){
    if(total == 0){
        return 1;
    }else{
        return 0;
    }
}

int full(){
    if(total >= TAM){
        return 1;
    }else{
        return 0;
    }
}

void enqueue(int id){
    if(full()){
        printf("Fila cheia");
        return;
    }

    caminhoneiro_id[total] = id;
    total = total+1;

    return;
}
```

- i. Implemente a função dequeue que deve remover e retornar um elemento representado por um caminhoneiro (id) ou a mensagem "Fila vazia." Se não houver elementos. (valor 1,25)
- ii. Implemente o procedimento print_fila que deve apresentar a lista dos elementos, ou seja, os ids dos caminhoneiros que estão na fila. (valor 1,25)

QUESTÃO 3. Thiago adora programar tomando um cafezinho. Como estava próximo da prova de estrutura de dados Thiago estava estudando filas circulares e fazendo seu algoritmo no caderno. Infelizmente aconteceu um acidente e Thiago derramou café no seu material de estudos, assim parte do código ficou ilegível. Ajude Thiago a implementar novamente os a fila circular indicando o código correto nos trechos manchados no código abaixo.

i. Procedimento print_filas. (valor 1,25)

```
void print_filas (struct fila* f){
    if (empty(f)){
        [REDACTED]
        [REDACTED]
    }
    int i;
    if(f->inicio [REDACTED] f->fim){
        for (i = f->inicio; [REDACTED] ){
            printf("%d ", f->dados[i]);
        }
    }else{
        for ([REDACTED] i < TAM ; i++){
            printf([REDACTED]);
        }
        for ([REDACTED]){
            printf("%d ", f->dados[i]);
        }
    }
    printf [REDACTED]
    return;
}
```

ii. Procedimento enqueue. (valor 1,25)

```
void enqueue(struct fila* f, int dado){
    f->dados[f->fim] = [REDACTED];
    int estado = f->fim;
    if ([REDACTED] == TAM-1){
        f->fim = [REDACTED]
    }else{
        f->fim = [REDACTED]
    }
    if(f->inicio [REDACTED] f->fim){
        printf("Estouro de fila. Insercao cancelada.\n");
        f->fim = estado;
        return;
    }
    return;
}
```

QUESTÃO 4. Listas são conjuntos de dados ordenados como pilhas e filas. O grande diferencial de uma lista é que seus elementos podem ser acessados de maneira dinâmica, ou seja, início, local específico ou fim. As listas podem ser consideradas estruturas de dados de tamanho dinâmico, pois cada elemento da lista possui um elo (link) para o próximo elemento da lista. Sobre listas:

- i. Defina, em linguagem de programação C, uma estrutura para representar o nó de uma lista. (valor 0,5)
- ii. Implemente um procedimento para inserir um elemento no início de uma lista. (valor 0,75)
- iii. Implemente uma função para remover um elemento no início de uma lista e retornar esse elemento. (valor 0,75)
- iv. Seria possível implementar uma fila dinâmica utilizando apenas os procedimentos dos itens ii e iii? Justifique. (valor 0,5)