



UNIVERSIDADE  
FEDERAL DO CEARÁ

**Avaliação 1**  
**Estrutura de Dados**  
Professora Tatiane Fernandes



**Nome:**

**Matrícula:**

- 1 – A interpretação também faz parte da avaliação.
- 2 – Você poderá utilizar caneta (preta ou azul) ou escrever a lápis.
- 3 – Pedidos de revisão de questões feitas à lápis serão apenas aceitas no dia e horário de entrega e correção da avaliação.
- 4 – Em caso de detecção de utilização de qualquer meio fraudulento na concepção das respostas da avaliação, os envolvidos terão suas notas zeradas sem possibilidade de realização de segunda-chamada.

**#VemmmHexa!** Com a grande repercussão do uso de tecnologias para melhorar a performance de várias áreas, dentre elas esportes em geral, a delegação brasileira resolveu investir uma \$\$\$\$ pesada em recursos tecnológicos que possam auxiliar a comissão técnica e jogadores a ganhar o tão sonhado HEXACAMPEONATO MUNDIAL!!

Para tal, ela contratou um estagiário do campus da UFC de Russas para criar um software que gerencia os dados de cada jogador! O problema é que o projeto começou a ficar lento demais e para garantir que o software fosse entregue a tempo, eles decidiram contratar mais um estagiário: que pode ser VOCÊ! :D Para te contratarem você precisa compreender os códigos que já estão feitos e são apresentados a seguir.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 typedef struct no{
5     char *jogador;
6     char *funcao;
7     int cansaco_mental;
8     int cansaco_fisico;
9     struct no *prox;
10    struct no *ant;
11 }NO;
12
13 NO *inicio = NULL;
14 NO *fim = NULL;
15 int tam = 0;
16
17 void adicionar(char *jog, char* func, int mental, int fisico){
18
19     NO *novo = malloc (sizeof (NO));
20     novo->jogador = jog;
21     novo->funcao = func;
22     novo->cansaco_mental = mental;
23     novo->cansaco_fisico = fisico;
24     novo->prox = NULL;
25
26     if(inicio == NULL){
27         inicio = novo;
28         fim = novo;
29         tam++;
30     }else{
31         fim->prox = novo;
32         novo->ant = fim;
33         fim = novo;
34         tam++;
35     }
36 }
```

```
38 char *remover(char *funcao){
39     char * jogador = "Sem opções!";
40     if(inicio != NULL){
41         NO * lixo = NULL;
42         float nivel_stress;
43         if(inicio->funcao == funcao){
44             nivel_stress = (inicio->cansaco_mental + inicio->cansaco_fisico)/2;
45             if(nivel_stress < 5){
46                 lixo = inicio;
47                 jogador = inicio->jogador;
48                 inicio = inicio->prox;
49                 inicio->ant = NULL;
50                 free(lixo);
51                 tam--;
52             }
53         }
54         if(lixo == NULL && fim->funcao == funcao){
55             //apresente o codigo e explique OU
56             //desenhe e explique esta parte do codigo
57         }
58         if(lixo == NULL && inicio->prox != NULL){
59             NO * aux = inicio->prox;
60             while(aux != fim){
61                 //apresente o codigo e explique OU
62                 //desenhe e explique esta parte do codigo
63                 aux = aux->prox;
64             }
65         }
66     }
67     return jogador;
68 }
69
70 void imprimir(NO *aux){
71     //apresente o codigo e explique OU
72     //desenhe e explique esta parte do codigo
73 }
```

```

101 int main(){
102     adicionar("Alisson - Liverpool", "Goleiro", 9, 8);
103     adicionar("Danilo - Juventus", "Lateral", 4, 0);
104     adicionar("Militão - Real Madrid", "Zagueiro", 5, 3);
105     adicionar("Fabinho - Liverpool", "Goleiro", 2, 1);
106     adicionar("Neymar - PSG", "Atacante", 8, 9);
107     printf("Jogador removido: %s.\n", remover("Atacante"));
108     printf("Jogadores disponíveis:\n");
109     imprimir(inicio);
110
111     return 0;
112 }

```

A ideia central do software é receber da comissão técnica uma avaliação dos jogadores contendo seu nível de cansaço físico e mental (uma escala de 0 a 10, onde quanto maior o valor, maior o nível de cansaço físico e mental do jogador).

*Assim, para ser contratado, você deve responder às seguintes questões:*

1 – (2,5). Explique e desenhe ou codifique e explique como deve ficar a parte do código ausente na função **remover**. Note que a remoção deve ser feita considerando um função (Goleiro, Zagueiro, etc.) e deve ser removido um jogador que tenha um nível de stress abaixo de 5 pontos.

2 – (2,5). Explique e desenhe ou codifique e explique como deve ficar a parte do código ausente da função **imprimir**. A função imprimir DEVE **apresentar o nome e função apenas os jogadores com que tenham um nível de cansaço mental abaixo de 5 E um nível de cansaço físico abaixo de 5. ESTA FUNÇÃO DEVE SER 100% RECURSIVA.**

3 - (2,5) (a) Após a inserção dos códigos ausentes e execução do mesmo, apresente como ficaria o resultado "printado" no terminal. (b) Suponha que o software irá sofrer uma atualização, considerando agora a quantidade de gols feita por cada jogador durante a copa do mundo. Além disso, o software também deverá armazenar o time que o jogador trabalha de forma separada. Apresente como ficaria o struct no com os novos dados.

4 – (2,5). Considerando a análise de complexidade simplificada vista em sala de aula (**linear ou constante**). Apresente a complexidade de cada um dos **casos das funções de adicionar e remover** do código acima, justifique de forma curta cada uma delas.