

Nome: _____ **GABARITO** _____*Obs.: A prova pode ser feita a lápis!*

1. O que será impresso na tela? Mostre o valor que cada variável assume durante a execução e também circule a resposta; não é para explicar o que cada linha faz.

a) (2pts) Escolhe a alternativa e "desenhe" como chegou a resposta

```
#include <stdio.h>
#include <stdlib.h>

typedef struct no {
    int dado;
    struct no *prox;
} No;

main() {
    No *L, *i;
    int n=6;
    if (n==0) L=NULL;
    else {
        L = (No *) malloc(sizeof(No));
        L->dado = n--;
        L->prox = NULL;
        for (;n>0;) {
            i = (No *) malloc(sizeof(No));
            i->dado = n--;
            i->prox = L;
            L = i;
        }
    }

    while (L!=NULL) {
        printf("%d ", L->dado);
        L = L->prox;
    }
}
```

1 2 3 4 5 6

Alternativas:

a) 0 1 2 3 4 5 6

b) 1 2 3 4 5 6

c) 6 5 4 3 2 1

d) 6 5 4 3 2 1 0

e) 1 2 3 4 5 6

b) (1,5pt)

```
void f1(int x, int *p) {
    *p = *p * 2;
    x = x + 2;
}
```

```
void f2(int *p) {
    *p = *p * 2;
}
```

```
main() {
```

```

int x=2,y=4,*p=&y;
f1(x,&x);
f2(p);
printf("%d",x+y);
}

```

12

c) (1,5pt)

```

void f1(int x, int *p) {
    *p = *p * 2;
    x = x + 2;
}

int f2(int *p) {
    *p = *p * 2;
    return *p;
}

main() {
    int x=1,y=3,*p=&y;
    f1(x,&y);
    x=f2(p);
    printf("%d",x+y);
}

```

24

2. Tendo o código:

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

typedef struct estruturaAluno {
    int matricula;
    char nome[80];
    float cr; // 0 a 10.0
} Aluno;

typedef struct lista {
    Aluno aluno;
    struct lista *prox;
} Lista;

int main(){
    Lista *inicio=NULL;

    inicio = arquivoParaLista("alunosRendimento.txt", inicio);
    imprimeLista(inicio);
}

```

a. (4pts) Desenvolva a função arquivoParaLista, a qual abre um arquivo em modo texto que contém em cada linha os três campos da estrutura Aluno. A cada aluno lido do arquivo, a função deve chamar uma função específica para inserir um novo nó no início de uma lista encadeada. Essa função deve ter o seguinte protótipo:

Lista *adicionarAlunoInicio(Lista *inicio, int matricula, char *nome, float cr);

As duas funções devem ser escritas, sendo que ambas retornam o início da lista.

Por simplificação, assuma que o nome do aluno é somente de uma palavra.

b. (1pt) Desenvolva a função imprimeLista, a qual imprime na tela os três campos de todos os alunos da lista encadeada.

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <locale.h>

```

```
typedef struct estruturaAluno {
    int matricula;
    char nome[80];
    float cr; // 0 a 10.0
} Aluno;
```

```
typedef struct lista {
    Aluno aluno;
    struct lista *prox;
} Lista;
```

```
Lista *adicionarAlunoFim(Lista *inicio, int matricula, char *nome, float cr){
    Lista *novo;
    novo=(Lista*)malloc(sizeof(Lista));
    novo->prox = NULL;

    if (inicio==NULL) // se for o primeiro
        inicio=novo;
    else {
        Lista *aux = inicio;
        while (aux->prox!=NULL)
            aux=aux->prox;
        aux->prox = novo;
    }

    novo->aluno.matricula=matricula;
    strcpy(novo->aluno.nome, nome);
    novo->aluno.cr=cr;

    return inicio;
}
```

```
Lista *adicionarAlunoInicio(Lista *inicio, int matricula, char *nome, float cr){
    Lista *novo;
    novo=(Lista*)malloc(sizeof(Lista));
    novo->prox = inicio;

    novo->aluno.matricula=matricula;
    strcpy(novo->aluno.nome, nome);
    novo->aluno.cr=cr;

    return novo;
}
```

```
Lista *arquivoParaLista(char *nomeArquivo, Lista *inicio) {
    FILE *arq;
    int matricula; char nome[80]; float cr;
    arq=fopen(nomeArquivo, "r");
    while(!feof(arq)){
        fscanf(arq, "%d", &matricula);
        fgetc(arq);
        fscanf(arq, "%81[^\n]", nome);
        fgetc(arq);
        fscanf(arq, "%f", &cr);
        fgetc(arq);
```

```
        inicio=adicionarAlunoInicio(inicio, matricula, nome, cr);
    }
    fclose(arq);
    return inicio;
}

void imprimeLista(Lista *inicio) {
    Lista *aux = inicio;
    for (;aux!=NULL; aux = aux->prox)
        printf("\n%d, %s, cr=%.1f", aux->aluno.matricula, aux->aluno.nome, aux->aluno.cr);
}

int main(){
    //  setlocale(LC_ALL, "");
    Lista *inicio=NULL;

    inicio = arquivoParaLista("alunosRendimento.txt", inicio);
    imprimeLista(inicio);
}
```