

Escreva a função recursiva `conta_elementos_rec()`, que recebe uma lista de alunos e retorna o número de alunos nessa lista.

Escreva a função recursiva `conta_sexo_rec()`, que recebe uma lista de alunos e um sexo, e retorna o número de pessoas com esse sexo na lista.

A seguir, algumas definições e implementações para você testar as funções desenvolvidas por você.

```
struct aluno
{
    int mat;
    char nome[21];
    float nota;
    char sexo;
    struct aluno *prox;
};
typedef struct aluno Aluno;

/* função de criação: retorna uma lista vazia */
Aluno* lst_cria (void)
{
    return NULL;
}

void lst_libera (Aluno* lst)
{
    Aluno *p = lst, *t;
    while (p != NULL)
    { t = p->prox; /* guarda referência p/ próx. Aluno */
      free(p); /* libera a memória apontada por p */
      p = t; /* faz p apontar para o próximo */
    }
}

/* inserção no início: retorna a lista atualizada */
Aluno* lst_insere (Aluno* lst, int mat, char *nome, float nota, char sexo)
{
    Aluno* novo = (Aluno*) malloc(sizeof(Aluno));
    if (novo == NULL)
        exit(1);
    novo->mat = mat;
    strcpy(novo->nome, nome);
    novo->nota = nota;
    novo->sexo = sexo;
    novo->prox = lst;
    return novo;
}

int main (void)
{
    Aluno *lal;
    lal = lst_cria();
    lal = lst_insere(lal, 33, "Caio", 3.3, 'M');
    lal = lst_insere(lal, 11, "Ana", 1.1, 'F');
    lal = lst_insere(lal, 44, "Edu", 4.4, 'M');
    lal = lst_insere(lal, 66, "Luiz", 6.6, 'M');
    lal = lst_insere(lal, 22, "Bia", 2.2, 'F');
    lal = lst_insere(lal, 55, "Jane", 5.5, 'F');
    /* inserir aqui a chamada para as suas funções e testes */
    lst_libera(lal);
    return 0;
}
```