



UNIVERSIDADE FEDERAL DO ABC
CENTRO DE MATEMÁTICA, COMPUTAÇÃO E COGNIÇÃO

disciplinasOrdenadas

disciplinasOrdenadas.[c | cpp | java | cs]

Considere a seguinte estrutura para disciplina (o campo `requisito` recebe NULL quando a disciplina não possui requisito):

```
typedef struct tDisciplina {  
    int cod;  
    char *nome;  
    int creditos;  
    struct tDisciplina *requisito;  
} Disciplina;
```

a) Implemente a função abaixo para criar e devolver um ponteiro para uma estrutura `Disciplina` com os valores passados por parâmetro, e com o ponteiro de `requisito` inicializado com NULL. A função deve seguir o seguinte protótipo:

```
Disciplina * criaDisciplina (int cod, char *nome, int creditos);
```

Considere a seguinte estrutura para curso (o campo `ultima_disciplina` recebe NULL quando a lista de disciplinas é vazia):

```
typedef struct tCurso {  
    int cod_curso;  
    char *nome_curso;  
    Disciplina *ultima_disciplina;  
} Curso;
```

b) Implemente a função abaixo para criar e devolver um ponteiro para uma estrutura `Curso` com os valores passados por parâmetro, e com o ponteiro de `ultima_disciplina` inicializado com NULL. A função deve seguir o seguinte protótipo:

```
Curso * criaCurso (int cod_curso, char *nome_curso);
```

Em um curso, os nomes das disciplinas requisito sempre estão ordenados lexicograficamente, ou seja, estão em ordem alfabética. Por exemplo: "Cálculo 1" é requisito para "Cálculo 2"; "Algoritmos" é requisito para "Programacao Paralela", que por sua vez é requisito para "Sistemas Operacionais".

A `ultima_disciplina` é a última disciplina do curso, que deve ser cursada após todas as demais (que seriam seus requisitos).

c) Implemente a função `insereDisciplina` que recebe um curso e uma disciplina nova. A disciplina nova é uma que deve ser inserida na sequência de pré-requisitos. Essa inserção deve considerar que as disciplinas requisito estão ordenadas lexicograficamente pelo nome. Observe que, dependendo do nome da disciplina nova, ela será a última a ser cursada, pois seu nome pode vir depois de todas as demais no curso. A função deve seguir o seguinte protótipo:

```
void insereDisciplina (Curso *curso, Disciplina *nova_disciplina);
```

Importante: para comparar duas strings, use a função `strcmp(str1, str2)` da biblioteca `<string.h>`. A função devolve:

- um valor `< 0` se `str1 < str2`,
- um valor `> 0` se `str1 > str2`,
- e `0` se `str1 == str2`.

Você deve usar a seguinte função `imprime()` e a seguinte função `main()`:

```
void imprime (Curso *c) {
    printf ("Disciplinas do curso: cod=%d nome=%s\n",
           c->cod_curso, c->nome_curso);
    Disciplina *d = c->ultima_disciplina;
    while (d != NULL) {
        printf ("cod=%d nome=%s cred=%d\n", d->cod, d->nome, d->creditos);
        d = d->requisito;
    }
}
```

```

int main () {
    int cod_curso;
    char nome_curso[100];
    scanf ("%d", &cod_curso);
    scanf ("%s", nome_curso);
    Curso *c = criaCurso (cod_curso, nome_curso);

    int cod, creditos;
    char *nome = NULL;
    while (scanf ("%d", &cod) != EOF) {
        nome = malloc (100);
        scanf ("%s %d", nome, &creditos);
        Disciplina *d = criaDisciplina (cod, nome, creditos);
        insereDisciplina (c, d);
    }
    imprime (c);
    return 0;
}

```

Entrada

A primeira linha contém os dados de um curso: um inteiro com o código do curso e uma string com o nome do curso.

A seguir, cada linha contém os dados de uma disciplina: um inteiro representando o código da disciplina, uma string representando o nome, e um inteiro representando os créditos.

O tamanho da entrada é indeterminada e termina com EOF.

Saída

Seu programa deve imprimir as informações das disciplinas do curso, encadeadas por uma LDE, iniciando do ponteiro `ultima_disciplina` e seguindo o ponteiro de `requisito` até o final da lista, conforme a função `imprime()`.

Exemplos

Entrada

```
111 Computacao
1 programacao_paralela 2
2 sistemas_operacionais 2
3 programacao_orientada_a_objetos 2
4 teoria_dos_grafos 2
5 programacao_estruturada 2
6 calculo_2 2
7 calculo_1 2
8 programacao_web 2
```

Saída

```
Disciplinas do curso: cod=111 nome=Computacao
cod=4 nome=teoria_dos_grafos cred=2
cod=2 nome=sistemas_operacionais cred=2
cod=8 nome=programacao_web cred=2
cod=1 nome=programacao_paralela cred=2
cod=3 nome=programacao_orientada_a_objetos cred=2
cod=5 nome=programacao_estruturada cred=2
cod=6 nome=calculo_2 cred=2
cod=7 nome=calculo_1 cred=2
```

Entrada

```
222 Estatistica
1 inferencia_estadistica 4
7 calculo_1 4
2 calculo_4 4
3 probabilidade 4
4 algebra 4
6 calculo_2 4
5 calculo_3 4
```

Saída

```
Disciplinas do curso: cod=222 nome=Estatistica
cod=3 nome=probabilidade cred=4
cod=1 nome=inferencia_estadistica cred=4
cod=2 nome=calculo_4 cred=4
cod=5 nome=calculo_3 cred=4
cod=6 nome=calculo_2 cred=4
cod=7 nome=calculo_1 cred=4
cod=4 nome=algebra cred=4
```