

CONTEXTO: SISTEMA DE GERENCIAMENTO DE RECURSOS DE UMA INSTITUIÇÃO (de ensino)

O objetivo deste trabalho é implementar um programa que gerencie o cadastro/uso de recursos de uma instituição de ensino, com o perfil já do seu conhecimento. Ou seja, os recursos associados a um Departamento de Ciência da Computação.

Assim, os **ambientes** seriam:

- Salas de aula (teóricas/práticas);
- Laboratórios específicos;
- Auditórios;
- Salas de reuniões;
- Salas dos professores; e
- Secretarias.

Considerando cada **ambiente**, um conjunto de **recursos** poderão estar associados:

- Computadores;
- Projetores;
- Mesas;
- Cadeiras;
- etc (aqui, você deverá apresentar recursos adicionais...).

IMPORTANTE! Você deverá fazer associações de recursos, considerando também situações onde é fundamental o agendamento conjunto. Exemplo: projetor + tela de projeção.

Portanto, a solução final deverá permitir:

- o cadastro/atualização de ambientes/recursos;
- a associação de um protocolo, que associa cada recurso com um número de patrimônio ESPECÍFICO;
- a identificação, de forma ORGANIZADA, dos ambientes (permitindo ainda recursos de reorganização).

Tipo *evento* - uma estrutura que contém os campos a seguir:

- cadastro/atualização de ambientes/recursos alocados (pontualmente);

O programa deve oferecer um MENU, com texto suficiente para apresentar as informações:

1. Cadastro de um novo ambiente/recurso;
2. Apresentação de todos os ambientes/recursos;
3. Apresentação de ambientes/recursos específicos;
4. Apresentação de todos os eventos, considerando uma **data específica**;
5. Apresentação de todos os eventos, considerando um texto que tenha relação com a descrição do evento (*string* exata);
6. Remoção de evento, considerando a data e horário de início;
 - o seu programa deverá identificar qual índice do vetor deve ser removido;
 - se existirem dados "na sequência", os mesmos devem ser deslocados;
 - atualizar a quantidade de eventos, bem como a realocação do vetor (espaço ocupado)

Requisitos do programa

- Validação da entrada (data e hora com valores válidos);
- Cuidado com a interface de usuário: coloque mensagens apropriadas para situações excepcionais, tais como erro na validação da entrada ou que o evento já existe (opção 1), lista de eventos vazia (opções 2, 3, 4), ou que o evento não foi encontrado (opção 5);
- Uso apropriado da realocação, para utilizar a memória de maneira otimizada;
- Não se esqueça de desalocar toda a estrutura ao final do programa! Bem como situações pontuais, durante o uso do sistema.

>>> Entrega: arquivo .c;

>>> Entrega: exemplos;

>>> Entrega: documentação/descrição.