

## Pontifícia Universidade Católica de Minas Gerais

<b>Disciplinas:</b>	Fundamentos de Engenharia de Software Algoritmos e Estruturas de Dados I Laboratório de Computação I
<b>Curso:</b>	Engenharia de Software
<b>Profs.:</b>	Maria Augusta Nelson, Roberto Felipe Rocha e Ivre Marjorie R. Machado
<b>Entrega:</b>	03/06/2018
<b>Valor:</b>	10 pontos (FES) – 7 pontos (AEDI) – 7 pontos (LCI)

### Observações:

- O trabalho poderá ser feito em **grupos de até 5 alunos**.
- Cópias de trabalho receberão a nota **ZERO**
- O trabalho será avaliado em **07 pontos**
- O programa deve ser feito na linguagem de programação C
- O programa deverá ser entregue pelo SGA até o dia **03/06/2018** as **23:59 horas**
- Deverá ser entregue o **projeto completo** do programa e a documentação
- Em caso de dúvida, envie email para seu professor

### Salão de Festas Patati Patata

O salão de festas Patati Patata é um salão de festas infantil que foi inaugurado no final de 2017 e, por isso, os donos ainda não haviam se preocupado com a implantação de sistemas para realizar o controle e gestão do negócio. No entanto, alguns problemas já começaram a aparecer, como por exemplo, a marcação de duas festas para um mesmo dia em horários que coincidem. Além disso, alguns dados de clientes e fornecedores, que deveriam ser armazenados e de fácil recuperação, não são feitos. Diante dos problemas vividos pelo salão, os donos resolveram contratar uma empresa desenvolvedora de sistemas (vocês). Sendo assim, é necessário compreender a real necessidade do salão e desenvolver um software específico. A seguir foi descrito como deverá ser o sistema, bem como suas restrições.

### O sistema

Deseja-se cadastrar os clientes, os funcionários, os fornecedores e as festas. As informações que devem ser cadastradas são:

- CLIENTE = código, nome, endereço, telefone, data de nascimento
- FUNCIONARIO = código, nome, telefone, função, salário, tipo (temporário ou fixo)
- FORNECEDOR = código, nome, telefone, produto fornecido
- FESTA = código festa, quantidade de convidados, data, dia da semana, horário (início e fim), tema, código cliente,
- CONTRATO = número contrato, valor total, desconto, valor final, forma de pagamento, status, código festa

Considere as seguintes **restrições**: para cadastrar uma festa, primeiro é necessário que o cliente esteja cadastrado. As festas devem ser cadastradas apenas em datas/horários que não tenham outras

## Pontifícia Universidade Católica de Minas Gerais

festas que coincidem. Considere que apenas no sábado o horário da festa é definido (fixo), pois acontecem duas festas, uma festa das 12 às 16 horas e outra das 18 às 22 horas. Já nos outros dias o cliente pode escolher o horário de início e fim que desejar, lembrando que a duração das festas é de 4 horas.

**\*\* Não esqueça de sempre validar essas restrições**

1. Implemente uma função para cadastrar um cliente. Esta função deve garantir que não haverá mais de um cliente com o mesmo código. Se quiser pode gerar o código automaticamente.
2. Implemente uma função para cadastrar um funcionário. Esta função deve garantir que não haverá mais de um funcionário com o mesmo código. Se quiser pode gerar o código automaticamente.
  - a. Lembre-se de cadastrar se é um funcionário fixo ou temporário
3. Implemente uma função para cadastrar um fornecedor. Esta função deve garantir que não haverá mais de um fornecedor com o mesmo código. Se quiser pode gerar o código automaticamente.
4. Implemente uma função que cadastre uma festa. Para cadastrar uma festa, o sistema deve receber do usuário o nome do cliente que deseja realizar a festa, a quantidade de convidados, a data da festa (lembre-se de validar se já existe festa), horário e tema.
5. Implemente uma função para calcular o valor total a ser pago, baseado na **tabela 1** e o valor final a ser pago, que deverá ser calculado baseado na forma de pagamento, para isso, deverá ser usada a **tabela 2**. Lembre-se de atualizar o status do contrato sempre inicialmente para “a pagar”.

**Tabela 1**

Qtde de convidados	Dia da semana	Valor
30	Segunda a quinta	R\$ 1899,00
30	Sexta a domingo	R\$ 2099,00
50	Segunda a quinta	R\$ 2199,00
50	Sexta a domingo	R\$ 2299,00
80	Segunda a quinta	R\$ 3199,00
80	Sexta a domingo	R\$ 3499,00
100	Segunda a quinta	R\$ 3799,00
100	Sexta a domingo	R\$ 3999,00

**Tabela 2**

Forma de pagamento	Desconto
A vista	10%
Duas vezes	5%
Três vezes	2%
Quatro ou mais vezes	Sem desconto

## Pontifícia Universidade Católica de Minas Gerais

6. Implemente uma função que permita atualizar o status do contrato do cliente para pago ou cancelado.
7. Implemente funções para realizar pesquisa no sistema (pelo nome do cliente):
  - a. informações de clientes
  - b. informações de funcionários
  - c. informações de fornecedores
8. Implemente uma função (relatório) que mostre na tela todas as festas de um determinado cliente.
9. Implemente uma função (relatório) que mostre na tela as informações de uma determinada festa a partir de uma data. Mostre inclusive as informações referente ao contrato da festa com valor total e final.

Para fazer este programa pode ser necessário criar mais funções do que as que estão descritas. Finalmente, faça uma função **main()** que teste o sistema acima. A função **main()** deve exibir um menu na tela, com as opções de cadastrar um cliente, um funcionário, um fornecedor e uma festa. Além disso, permitir realizar as pesquisas. Este menu deve ficar em loop até o usuário selecionar a opção **SAIR**. Além disso, todas as informações deverão ser armazenadas em arquivo(s) texto, portanto, deverá ser feita leitura e escrita em arquivos.

### Metodologia

Este é um trabalho interdisciplinar onde você vai planejar, analisar, projetar e implementar uma solução de software para o problema de controle do salão de festas utilizando o Scrum para gerenciar o seu progresso.

Inicialmente organize o seu *backlog* de produto contendo as funções básicas do sistema. Cada uma das funções será responsabilidade de um membro do grupo e será feita em *sprints* de 3-4 dias. Seguem algumas sugestões de atividades a serem feitas nas *sprints*:

- 1- Definir a assinatura da(s) função(ões). Reflita sobre os parâmetros de entrada e saída da função e comunique aos seus colegas de projeto.
- 2- Documente os parâmetros da sua função. O nome da função deve ser escolhido sob o ponto de vista de quem usa a função ou de quem vai chamar a função e deve refletir o que a função faz.
- 3- Implemente o caso de sucesso da função.
- 4- Selecione casos de testes para verificar o funcionamento da sua função. Um caso de teste deve conter os valores de entrada para a função, a saída esperada.
- 5- Execute os casos de testes, criando um relatório de execução de testes que contem os casos de testes, a saída retornada durante a execução e uma indicação se o teste passou ou não na função. Isso é feito comparando a saída esperada documentada no caso de testes com a saída retornada durante a execução (esperado x real).
- 6- Implemente os casos especiais, exceções que possam existir na sua função. Em seguida execute

## Pontifícia Universidade Católica de Minas Gerais

os casos de testes anteriores para garantir que as mudanças não quebraram o código anterior que já funcionava. Pense também nos novos casos de testes necessários para a nova versão da função.

### O que deve ser entregue para os professores no SGA

- 1- A evolução do *backlog* de produto a cada semana. Indique quais tarefas encontram-se no *backlog* do produto, as que foram alocadas nas *sprints* (indique em qual *sprint* cada tarefa foi alocada).
- 2- A documentação das funcionalidades do software
- 3- A documentação de casos de testes com o relatório de execução dos testes.
- 4- O código em C das funções e do programa principal, juntamente com o **projeto completo** do programa.