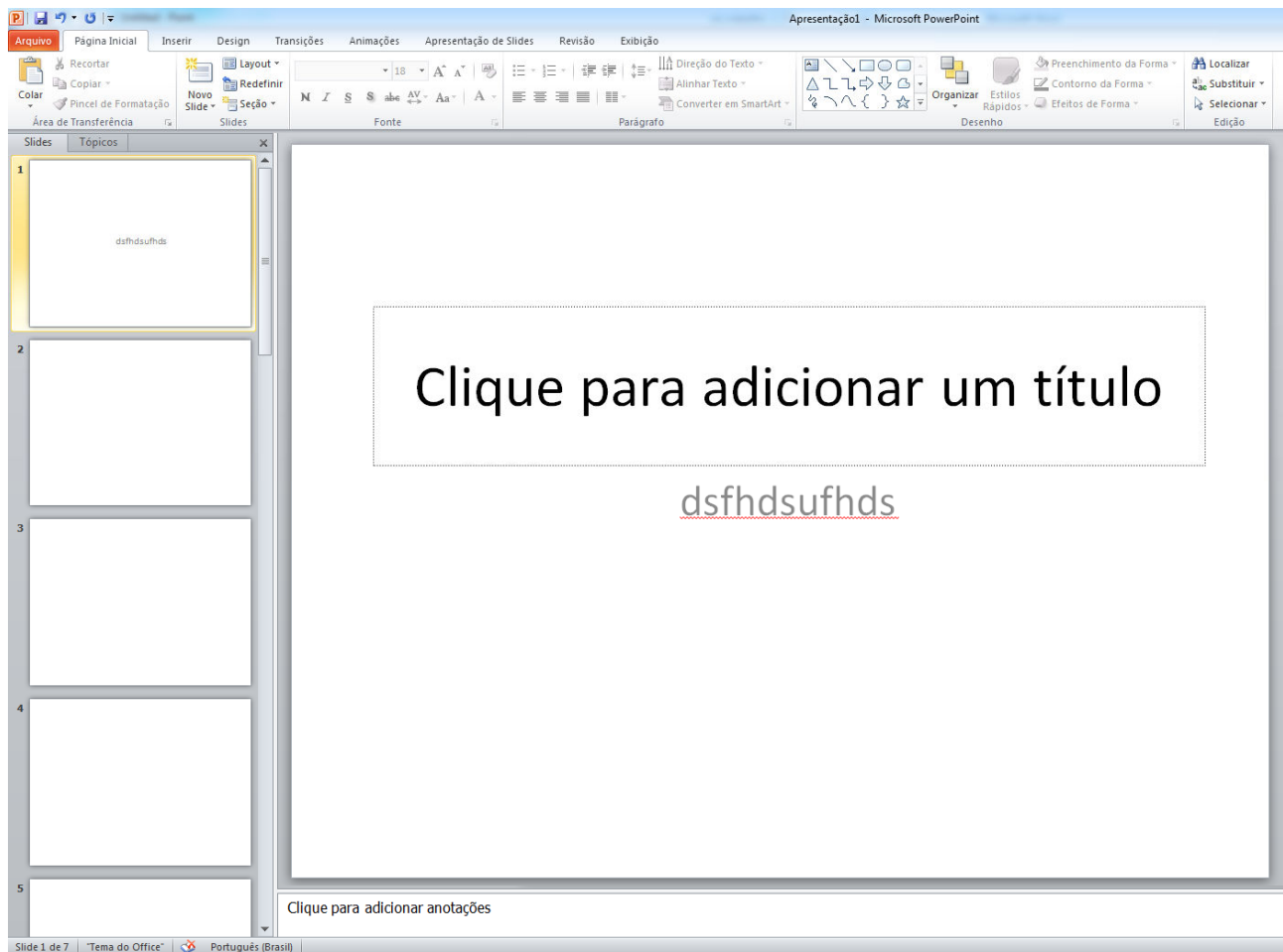


Universidade Federal de Santa Maria
Curso de Ciência da Computação
Disciplina: Estruturas de Dados
Primeiro Semestre de 2011
Prof. Cesar Tadeu Pozzer
Data: 15/09/2011

Trabalho 3 – Apresentação de Slides

Implemente em C ou em C++, utilizando o **MPC** (Multi-Platform Console, versão 4.6.2), um programa que visualize slides.

Deve-se utilizar os trabalhos T1 e T2 para a composição da interface. Além dos componentes Aba de Botões e Barra Lateral, deve-se implementar a área de exibição de slides. Cada slide deve ter um conteúdo único.



Observe que o gerenciamento do conteúdo é relativamente simples, porem a navegação e interação podem ser mais complicados.

Para mais dicas de bônus, vejam os recursos que os programas de apresentação de slides dispõem. A nota deste trabalho por ir além de 10 dependendo do que for implementado.

Requisitos para implementação:

1. Navegar nos slides
2. Editar slides
3. Salvar apresentação
4. Abrir apresentação salva
5. A altura do slide deve ter no mínimo 40 linhas e a largura deverá ter no mínimo 90 caracteres.
6. Imagem (bônus)

Para salvar uma apresentação de slides deverá ser seguida esta formatação (XML-Like):

```
<slides>3</slides>
<slide>1</slide>
<content>
    <textbox> duhau uhaudhauuah uhasudhuash uasudaus duauda</textbox>
    <image img="imagem1.bmp" lin="50" col="2">
    <image img="imagem2.bmp" lin="100" col="2">
</content>
<slide>2</slide>
<content>
</content>
<slide>3</slide>
<content>
</content>
```

Não devem existir espaços em branco dentro das tags, como em < slides >. Um slide terá **uma** caixa de texto que cobrirá todo slide e poderá ter N imagens. A caixa de texto terá o mesmo tamanho do slide e será armazenada no formato de um vetor de char de tamanho LINHAS * COLUNAS.

MPC:

Estudando o arquivo “**mpc.h**” pode se ter uma noção de quais funções o MPC disponibiliza para criação elementos na janela criada pelo próprio MPC. Qualquer dúvida sobre o MPC e/ou suas funções falar com o monitor.

Dicas:

Procure fazer um código padronizado e bem legível. Procure não sair fazendo código diretamente, procure pensar em como exatamente será toda a estrutura do programa e em como será sua organização (considere sempre os seus prazos) para criação de algo **genérico** e **reutilizável**. Para este trabalho é fundamental que seja feita uma estruturação antes de se iniciar a implementação. Faça desenhos em papel de estruturas de dados, formas de interação, modularização de código, etc.

Data e Formato de Entrega:

O trabalho deve ser entregue até o dia 28/11/2011.

No email e no cabeçalho do arquivo fonte, deve conter o nome completo e matrícula do aluno. O programa deve ser enviado em um arquivo compactado fulano.rar (fulano = login da inf). Dentro deste arquivo deve haver um diretório com o mesmo nome do arquivo e dentro deste diretório os arquivos do trabalho. O arquivo deve ser enviado para **pozzer3@gmail.com** e **fredericoal@gmail.com** com o assunto “ED – T3”.

Deve-se enviar somente:

- Arquivos fontes (**somente os arquivos necessários**)
- Imagens que forem utilizadas (limitado a 30KB por imagem) e no máximo 20 imagens
- Projeto Codeblocks para windows
- **Não** deve ser enviado a lib do MPC. Será usada para compilação a lib que está no site.

Critério de Avaliação:

- **Identificação:** Deixar explicado no cabeçalho do arquivo principal (main) os níveis implementados e os extras (se implementar algo fora da lista).
- **Documentação:** descrever no cabeçalho de cada arquivo a idéia geral do código e detalhes específicos de partes que mereçam uma explicação – não comente por exemplo o que faz b++.
- **Pontualidade:** **Trabalhos não entregues na data não serão avaliados e receberão nota zero.**
- **Legibilidade:** nome de variáveis, estruturação do código. O código digital a ser entregue deve ter *3 ou 4 espaços de indentação*.
- **Clareza:** facilidade de compreensão – evite códigos complexos e desnecessários. Adote a solução mais simples possível.
- **Funcionalidade:** o programa deve satisfazer todos os requisitos. **Programas que não compilarem ou que não atenderem nenhum requisito receberão nota 0** (zero).
- **Estruturação:** Como é um programa grande, deve-se elaborar estruturas bem organizadas e modularizadas. A modularização neste trabalho pode ser utilizada em trabalhos futuros relacionados, além de disponibilizar um treinamento para se ter um código mais padrão e compreensivo.

Para maior garantia de funcionamento e compilação, procure compilar o código em **Windows e em Linux.**

Você pode discutir estratégias e ajudar o colega na implementação, porém evite passar código-fonte. Programas semelhantes terão a nota 0 (zero).