

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
DEPARTAMENTO DE INFORMÁTICA APLICADA
INF01120 - TÉCNICAS DE CONSTRUÇÃO DE PROGRAMAS
PROFESSOR: Marcelo Soares Pimenta (mpimenta@inf.ufrgs.br)
Semestre 2020/2

Enunciado do Trabalho Prático – Fase 1

Introdução

Este é um trabalho em grupo, e a lista dos membros deve ser postada no fórum criado para isto no moodle da disciplina. LEIAM ATÉ O FIM este enunciado !!!!!

Descrição:

O objetivo do trabalho é a definição, a IMPLEMENTAÇÃO, teste e depuração de um GERADOR DE MÚSICA A PARTIR DE TEXTO, um software que recebe um TEXTO (a princípio, um texto livre, não estruturado, como um conto ou uma página de jornal) como entrada e gera (podemos dizer informalmente que o software 'toca' via acionamento de funções de som) um conjunto de NOTAS CORRESPONDENTES AO TEXTO segundo alguns PARÂMETROS (timbre, ritmo – na forma de Beats por Minuto ou BPM, etc).

Os parâmetros são definidos através de um mapeamento de texto para informações musicais. Parte do mapeamento já está definida abaixo. O restante deve ser definido pelo grupo e documentado para conhecimento do professor.

O mapeamento PRÉ-DEFINIDO proposto é o seguinte:

Texto	Informação Musical ou Ação
(letra A ou a)	Nota Lá
(letra B ou b)	Nota Si
(letra C ou c)	Nota Dó
(letra D ou d)	Nota Ré
(letra E ou e)	Nota Mi
(letra F ou f)	Nota Fá
(letra G ou g)	Nota Sol
Caractere Espaço	Silêncio ou pausa

Texto	Informação Musical ou Ação
Caractere + (sinal de adição)	Aumenta volume para o DOBRO do volume atual
Caractere - (sinal de subtração)	Volume volta a ser volume default (o valor de início)
Qualquer outra letra vogal (O ou o, I ou i, U ou u)	Se caractere anterior era NOTA (A a G), repete nota; Caso contrário, fazer som de “Telefone tocando” (125)
Caracteres R+ (letra R seguida de sinal de adição)	Aumenta UMA oitava em relação à atual
Caracteres R- (letra R seguida de sinal de subtração)	Diminui UMA oitava em relação à atual
Caractere ? (ponto de interrogação)	Toca uma nota aleatória (de A a G), randomicamente escolhida
Caractere NL (nova linha)	Trocar instrumento (A DEFINIR pelo grupo como escolher o instrumento)
Caracteres BPM+ (letras BPM seguidas de sinal de adição)	Aumenta BPM em 80 unidades
Caractere “;” (ponto e vírgula)	Atribui um valor aleatório à BPM
ELSE (nenhum dos caracteres anteriores)	NOP (Continua a fazer o que está fazendo)

A entrada do texto DEVE ser via leitura em um campo texto na interface do software .

Para a saída sonora , todos podem usar alguma API ou biblioteca de funções de som, como (em Java) JavaSound ou JFugue.

Para se ter uma ideia de como funciona, uma demonstração de software similar ON-LINE pode ser achada no link listado no item (a) abaixo.

A bibliografia básica inclui:

a) Exemplo de software similar:

- <http://p22.com/musicfont/> - P22 Music Text Composition Generator (A free online music utility)

b) Exemplos de parâmetros que podem controlar a performance:

- Ver descrição em : <http://tones.wolfram.com/about/controls-side.html>
- Ver explicação na wikipedia sobre notação musical:

“Notation is the written expression of music notes and rhythms on paper using symbols. When music is written down, the pitches and rhythm of the music is notated, along with instructions on how to perform the music.” (<http://en.wikipedia.org/wiki/Music#Notation>)

- Ver descrição mais completa de parâmetros em http://en.wikipedia.org/wiki/Music_theory – ver Fundamentals of Music
- O código dos instrumentos (piano, “telefone tocando”, etc) está definido no General MIDI (ver https://pt.wikipedia.org/wiki/General_MIDI)

c) Como API de som, o grupo pode escolher. Para quem vai utilizar Java, sugiro JavaSound ou JFugue, mas seu uso não é obrigatório, apenas sugerido.

O grupo deve imaginar que haverá alguma mudança nas fases 2 e 3 do trabalho prático. Então, pense no software de modo a facilitar estas mudanças.

O trabalho pode ser dividido em partes:

1a Parte:

Elaborar uma **lista** (informal, em português) **dos requisitos** que o sistema deve possuir no ponto de vista dos autores do projeto. Cada requisito é um item da lista e pode ser do tipo funcional (relativo a funcionalidade do sistema) ou não-funcional (relativo a características de performance, usabilidade, cronograma, orçamento, limitações tecnológicas, entre outras). Neste trabalho em particular, o grupo deve elaborar uma lista de quais operações o sistema permitirá que sejam realizadas sobre quais elementos, deixando claro quais as consequências (alteração de aparência, mensagens ao usuário, etc).

2a Parte:

Nesta parte, o grupo deve **PROJETAR A PARTE FUNCIONAL** do sistema, preferencialmente **DEFININDO CLASSES** – com atributos e métodos - correspondentes a Tipos Abstratos de Dados que fazem parte do programa. Não se exige uma notação específica. A definição (em qualquer sintaxe) das classes deve conter os atributos e métodos (mesmo que depois isto mude, coloquem ao menos os definidos até agora)

3ª Parte:

Nesta parte, o grupo deve **PROJETAR A INTERFACE COM O USUÁRIO** do sistema. Para isto, deverá no mínimo estabelecer as tarefas principais que serão suportadas. Além disto, a documentação deverá incluir uma impressão (ou **croqui/maquete**) de cada tela/unidade de apresentação por página, seguida das explicações/argumentações que justifiquem seu *layout* e seu comportamento.

4ª Parte:

Nesta parte, o grupo deve **IMPLEMENTAR** um protótipo do sistema projetado nas partes anteriores. Não é definida *a priori* nenhuma tecnologia para ser usada mas o grupo pode alternativamente escolher a linguagem e a plataforma mais adequados para seu desenvolvimento, obviamente compatível com os elementos e restrições definidos pelo professor. Reuso de algum código existente é tolerado, mas deve ser sinalizado na documentação e as razões de sua utilização deverão ser explicadas ao professor na documentação entregue.. Implementar o protótipo com o máximo possível de funções que foram projetadas. Não esquecer de **TESTAR** e **DEPURAR** o que foi implementado **ANTES** de demonstrá-lo ao professor. Veremos isto em aula.

5ª Parte:

Nesta Parte, o grupo deve APRESENTAR as características principais de seu projeto e DEMONSTRAR o sistema para o professor. No final do semestre, cada grupo poderá ser convidado a apresentar as idéias principais de seu trabalho para a turma (20 min no máximo por apresentação). Além disto, a versão implementada do sistema deverá ser apresentada ao professor em uma sessão especial de demonstração em data a ser definida. Assume-se que a documentação final do trabalho será entregue ao professor no máximo até o dia da demonstração. O professor não apenas fará uso da documentação para utilizar o sistema e avaliá-lo mas também poderá questionar oralmente os componentes do grupo a respeito de todas as partes do trabalho. Deve ficar clara nas respostas a participação e o engajamento de todos os indivíduos no trabalho do grupo.

Observações:

O trabalho FINAL (ao fim das fases 1,2 e 3) a ser entregue inclui a documentação associada a TODAS as partes e deve conter também a identificação do grupo (número do grupo, nomes de todos componentes) e todas as suposições feitas durante a realização do trabalho. Definições mais precisas que porventura sejam necessárias serão acrescentadas no decorrer do semestre. Em caso de dúvidas, consulte primeiro a bibliografia disponível (preferencialmente), depois pesquise em outras bibliografias (use a biblioteca e a Internet) e, em caso de necessidade, consulte o professor pessoalmente ou via e-mail. **A disciplina prevê algumas aulas de reuniões com o professor para acompanhamento dos projetos e esclarecimento das dúvidas surgidas. As datas serão divulgadas em aula e via moodle.** Em caso de reunião fora do período de aula, lembre-se de agendar com o professor (via email) para garantir que será atendido. Sem agendamento prévio, a prioridade do professor será sempre atender aos compromissos agendados previamente. Bom trabalho!!

Datas importantes SERÃO DEFINIDAS EM AULA e DIVULGADAS VIA MOODLE

A Entrega da FASE 1 consiste de:

- Definições de classes (sem sintaxe específica, apenas listem as classes, atributos e métodos que estiverem definidos)
- Croquis (maquetes) de interface com usuário, de todas telas previstas
- Lista de requisitos funcionais: uma lista informal do que o software irá fazer (para ficar claro o que vcs têm em mente)

Tendo em vista esta entrega, pode-se notar que correspondem às partes 1,2 e 3 acima. NO ENTANTO, mesmo que NÃO seja necessária a entrega de alguma implementação nesta FASE1, o professor aconselha que tentem implementar as classes e começar a construir o código, pois MUITAS dúvidas surgem e MUITAS MUDANÇAS decorrem disto.

Além disto, AULAS DE ACOMPANHAMENTO serão agendadas para resolução de dúvidas ou discussão do trabalho com os colegas e o professor. Uma vez decididas, as datas serão postadas no moodle da disciplina.