

Introdução ao Lilypond

Hugo Ribeiro

7 de setembro de 2007

1 Introdução

Essa é a primeira versão de uma breve apostila que irá ajudá-los a se iniciarem no uso do programa de edição de partituras Lilypond. Este é um programa gratuito que roda tanto na plataforma Linux quanto Windows. As informações contidas nessa apostila poderão ser utilizadas em ambas as versões.

Há de se destacar, todavia, a vantagem do uso do Lilypond no Linux, ao possibilitar a junção de texto e música num único arquivo Lytex, que é compilado como um arquivo Latex com partituras. Essa apostila foi escrita dessa forma. Entretanto, esse recurso também pode ser explorado, de forma mais simplificada, ao usar a macro OOoLylipond dentro do OpenOffice, ou sua versão em português, o BrOffice. Informações sobre como instalar essa macro no BrOffice já está disponível em português no site www.hugoribeiro.com.br\lytex.html. A vantagem dessa macro é que pode ser usada tanto no Linux quanto no Windows.

Um dos poucos inconvenientes do Lilypond para algumas pessoas é o fato de que os comandos terem que ser escritos todos em inglês. Porém, nada que a experiência ou uma pequena apostila ao lado não possa resolver. Outro inconveniente é que você tem escrever o código do lilypond em um arquivo de texto (use preferencialmente o bloco de notas do windows, ou sua versão atualizada que é facilmente encontrada na Internet, o notepad2¹), e depois compilar² o código escrito em uma partitura no formato .ps (PostScript) ou no formato .pdf (Portable Document Format). A grande vantagem aqui está no formato final da partitura, que é um arquivo .pdf, que pode ser visualizado em praticamente qualquer computador³; e como o conteúdo de um arquivo pdf não pode ser alterado, não importa em qual computador você visualize o arquivo, ele sempre será exibido da mesma forma.

1.1 Informações básicas

O lilypond é um editor não gráfico, ou seja, diferentemente do Encore, Finale ou Sibelius, você não vê o que está escrevendo. No Lilypond você tem que dizer o que quer

¹A vantagem do notepad2 sobre o bloco de notas é que ele “entende” um código de programação, diferenciando em cores o que é um comando e o que é o conteúdo.

²Segundo a Wikipedia, uma definição clássica de compilar um arquivo seria traduzir um programa de uma linguagem textual facilmente entendida por um ser humano para uma linguagem de máquina, específica para um processador e sistema operacional. No caso do Lilypond significa transformar o código em partitura.

³Um arquivo .pdf é aberto em qualquer programa visualizador de .pdf, tal como o Acrobat Reader. Se seu computador não tiver o Acrobat Reader, ou similar, instalado, ele é facilmente encontrado na Internet de forma gratuita.

escrito na partitura, e o programa decide a melhor forma de organizar as informações numa partitura. Dessa forma, escrever uma escala no Lilypond é tão simples quanto digitar o nome das notas, tal como no Ex. 1.

A escrita do código de uma partitura no Lilypond segue a lógica de programação que, como o próprio nome já diz, tem uma lógica. A primeira informação básica a respeito do Lilypond é que todo o código tem que ser escrito dentro de chaves ‘{ }’. A segunda informação é que as notas são escritas como cifras, ou como era de se esperar, são escritas em inglês (dó = c, ré = d, mi = e, fá = f, sol = g, lá = a, si = b).

IMPORTANTE: o Lilypond, assim como qualquer linguagem de programação, distingue as letras maiúsculas e minúsculas. Portanto, preste muita atenção nos exemplos, e respeite o uso de maiúsculas e minúsculas. Por exemplo, se escrever as notas com letras maiúsculas, vai dar erro! Considere-se avisado.

2 Como rodar o Lilypond no Windows

Em primeiro lugar, instale o programa :). Depois abra o bloco de notas, escreva o código tal como está escrito no Ex. 1, e salve o arquivo com um nome qualquer com a extensão .ly (exemplo, escala.ly). O resultado seria um arquivo representado por uma colcheia preta, tal como a figura 1. Mas pode ser que o resultado seja um arquivo .txt, fácil de identificar pelo ícone que o representa, figura 2. Se este for o seu caso (fig. 02), você precisará modificar uma opção padrão do Windows. Nada muito difícil. Primeiro, abra o Windows Explorer. Acesse através do menu ‘Ferramentas’ – ‘Opções de pasta’. Clique na aba ‘Modos de exibição’, e na janela de ‘Configurações avançadas’ desmarque a opção ‘Ocultar as extensões dos tipos de arquivos conhecidos’. Clique ok para fechar a janela, e pronto.

Uma informação importante é que, ao mudar a extensão de um arquivo, você estará mudando o tipo de associação arquivo-programa. Ou seja, um arquivo com a extensão .txt abre no Bloco de Notas. Um arquivo com a extensão .doc irá abrir no Microsoft Word. Um arquivo .odt irá abrir com o BrOffice. E um arquivo .ly está associado com o programa Lilypond.

Se você efetuou as alterações necessárias, a partir de agora você estará apto a visualizar o nome do arquivo e sua extensão. Clique no arquivo ‘escala.ly.txt’, aperte F2 (ou clique com o botão da direita e selecione ‘renomear’) e apague a extensão .txt, deixando somente o nome ‘escala.ly’. Aperte Enter. Se você fez tudo certo, o ícone do Bloco de Nota deixará de existir e em seu lugar aparecerá o ícone do Lilypond.



Fig. 01 – Ícone do Lilypond



Fig. 02 – Ícone do Bloco de Notas

Uma vez criado o arquivo Lilypond, basta clicar duas vezes sobre o ícone que uma janela preta irá aparecer. Isso significa que o Lilypond está compilando o código e transformando-o em uma partitura. A depender do tamanho de seu arquivo e da velocidade de processamento do seu computador, o processo pode ser muito rápido ou um pouco demorado. Terminado esse processo de compilação, você terá quatro arquivos, o original ‘escala.ly’, e três novos: ‘escala.log’, ‘escala.ps’ e ‘escala.pdf’.

2.1 O arquivo log

O arquivo 'escala.log' contém as informações sobre o processo de compilação. Em geral você não vai se importar com ele, e pode apagá-lo sempre que quiser. Ele só te será útil quando seu arquivo contiver algum erro de sintaxe. Nesse caso, não será gerado nenhum arquivo .ps nem um arquivo .pdf ou, se esses arquivos já existirem, não serão atualizados pelas novas mudanças. Digamos que você tenha criado um arquivo 'escala.ly' na raiz do disco rígido 'D' com o seguinte conteúdo: { c de f g }. Ao clicar duas vezes sobre o ícone irá gerar somente um arquivo .log com as seguintes informações:

```
# -*-compilation-*-
Changing working directory to 'D:/'
Processing 'D:/escala.ly'
Parsing...
D:/escala.ly:2:2: error: syntax error, unexpected STRING
c
  de f g
D:/escala.ly:1:0: error: errors found, ignoring music expression
{
D:/escala.ly: 0: warning: no \verb| \version statement found, please add
\version "2.10.25"
for future compatibility
error: failed files: "D:\\escala.ly"
```

A linha com a informação "error: syntax error, unexpected STRING" está informando que houve um erro de sintaxe, e em seguida mostra onde está o erro, que é facilmente visualizado pois ele quebra a linha exatamente no ponto anterior ao erro. Nesse caso o erro foi a falta de espaço entre a nota ré (d) e a nota mi (e).

Sempre que quiser modificar o conteúdo de um arquivo Lilypond, ou seja, com a extensão .ly, não dê dois cliques, pois isso irá compilar o arquivo. Para a modificar o conteúdo você tem duas opções: 1) clicar com o botão direito do mouse e em seguida clicar em 'abrir' ou em 'edit source'. Dessa forma irá abrir um Bloco de Notas com o código fonte. Altere o que quiser, depois salve o arquivo. 2) Abra o programa Bloco de Notas ou o Notepad2, e abra o arquivo 'escala.ly'.

2.2 Os arquivos .ps e .pdf

Os arquivos 'escala.ps' e 'escala.pdf' são a partitura tão desejada. O arquivo .pdf pode ser visualizado no Acrobat Reader, ou num programa similar. O arquivo .ps pode ser visualizado no Ghostview ou similar. Como ambos os arquivos são idênticos, com a diferença de extensão (o que irá exigir programas diferentes para sua visualização) e de tamanho (em geral o arquivo .ps é bem maior que o arquivo .pdf), eu aconselho que, caso tudo corra normalmente, após finalizar a partitura, se apague os arquivos .log e .ps, que não terão nenhuma utilidade posterior.

IMPORTANTE: Para compilar o código do Lilypond é necessário fechar o arquivo .pdf. Não é preciso fechar o programa Acrobat Reader, somente o arquivo. Se você estiver visualizando a partitura, quiser modificar alguma coisa ou acrescentar mais compassos, você pode abrir o código e fazer qualquer tipo de acréscimo ou alteração que desejar e ao final salvar o arquivo Lilypond. Mas isso não significa que as alterações ou acréscimos irão aparecer na sua tela de imediato. Após modificar ou alterar o código é preciso compilar o arquivo novamente para que as modificações ou alterações tenham efeito. Mas, antes de compilar o código, lembre-se de fechar o arquivo .pdf, senão dá

erro.

Para compilar o arquivo Lilypond não é necessário fechar o programa que está mostrando o código. Só é preciso lembrar de salvar o arquivo para que as modificações tenham efeito.

3 Exemplos Simples

A partir dessa seção eu irei mostrar alguns exemplos simples e fazer observações sobre o código. Experimente fazer cada um desses exemplos e modificá-los à vontade.

Exemplo 1

```
{ c d e f g f e d }
```



Neste primeiro exemplo note que o código é muito simples, e consiste somente das chaves e do nome das notas. Automaticamente o Lilypond acrescenta: 1) as linhas do pentagrama, 2) a clave de Sol, 3) o compasso quaternário, 4) as figuras rítmicas em semínima e, 5) inicia a melodia no dó2.

Esse é o padrão do Lilypond. Se quiser que alguma coisa seja escrita diferente do padrão, é preciso especificar o que e onde.

Exemplo 2

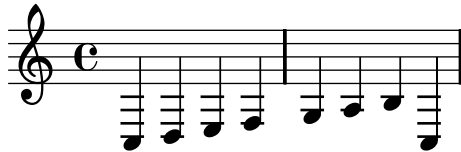
```
{ \clef bass  
c d e f  
g f e d }
```



Neste segundo exemplo eu repeti as notas do exemplo anterior, mas fiz duas alterações. A primeira foi o acréscimo do comando `\clef bass`. Esse comando diz ao Lilypond que eu quero uma clave de baixo, ou seja, uma clave de fá ao invés da clave de sol padrão. Outra modificação foi que eu dividi a melodia em duas linhas, cada linha com quatro notas. Dessa forma eu posso visualizar no código onde começa e onde termina cada compasso. A divisão do código em linhas não afeta em nada o resultado final. É somente uma forma de organizar melhor o código. Isso pode parecer sem efeito em exemplos pequenos, mas quando você tem uma música com cinquenta compassos, e encontra um erro no compasso 33, você vai querer ter um código bem organizado, para encontrar rapidamente o compasso desejado.

Exemplo 3

```
{ c
d
e
f           g           a
b
c }
```



Esse exemplo está com o código propositadamente bagunçado, para que você perceba que o Lilypond não leva em consideração a quantidade de espaços entre as notas, nem tampouco a quebra de linha. O único requisito mínimo é haver um espaço entre as notas. Mas, escrever dois, três ou dez espaços entre as notas, ou quebrar as linhas de forma aleatória não irá mudar o resultado final da partitura. Na hora de compilar, o Lilypond “lê” o código fonte da esquerda para a direita e de cima para baixo. O que importa é que as informações estejam escritas nessa ordem lógica.

Outra informação interessante nesse exemplo é que o Lilypond usa como padrão o registro de oitava referente ao dó₂. Isto é, toda nota escrita sem nenhuma informação adicional será interpretada como pertencente ao registro de oitava dó₂. Por isso, neste exemplo 3, a primeira nota é um dó₂ e a última nota também é o dó₂, pois o código escrito é igual. Para mudar de registro de oitava é preciso acrescentar um apóstrofe (') para uma oitava acima, ou uma vírgula para uma oitava abaixo.

Exemplo 4

```
{ c d e f
g a b c'
c' d' e' f'
g' a' b' c'' }
```



Este exemplo volta a ter o código organizado, sendo cada linha referente à um compasso. A mudança do registro de oitava do dó₂ para o dó₃ foi atingido acrescentando um apóstrofe para cada nota da escala, ou no caso da última nota, um dó₄, duas apóstrofes. Seguindo o mesmo pensamento, todas as notas do registro de oitava referente ao dó₄-si₄ precisariam de duas apóstrofes depois da nota.

Exemplo 5

```
{ c c'' d' e  
f' d'' e' f'''}  
\clef bass  
c b a d  
e, f, g' a, }
```



O exemplo 5 mostra notas em registros diversos com o uso de apóstrofes e vírgulas para oitavas graves. Perceba que depois da segunda linha foi escrito o comando `\clef bass`. Dessa forma a partitura começou com a clave de sol padrão, e a partir do terceiro compasso mudou para a clave de fá. Simples assim...

Exemplo 6

```
\relative {  
c d e f  
g a b c  
d e f g  
a b c d  
}
```



Este exemplo 6 introduz o comando `\relative`. Esse comando faz com que o Lilypond interprete uma nota em relação à anterior. Ou seja, qualquer nota escrita será relativa à nota anteriormente escrita. E essa relação é sempre da nota mais próxima possível, isto é, se eu escrevo a nota si e depois a nota dó, o Lilypond vai procurar o dó mais próximo desse si. Dessa forma, com o comando `\relative` fica muito mais simples escrever melodias simples com poucos saltos. Note que, se eu escrever somente o comando `\relative`, o Lilypond assume o dó3 como dó inicial.

Exemplo 7

```
\relative c {  
c d e f  
g a b c  
d e f g  
a b c d  
}
```



Neste exemplo, semelhante ao exemplo anterior, eu só acrescentei uma nota dó (c) após o comando `\relative`. Neste caso, o Lilypond volta a interpretar o dó2 como dó padrão.

Exemplo 8

```
\relative c' {  
c d e f  
g a b c  
d e f g  
a b c d  
}
```



Uma das vantagens do comando `\relative`, é o de indicar qual o registro de oitava inicial da música. Ao inserir um apóstrofe no dó após `\relative`, nós informamos ao Lilypond que o dó inicial passa a ser o dó3. Logicamente, se quisermos iniciar uma melodia no dó4, basta inserir duas apóstrofes (ex. `\relative c''`). Perceba que o `\relative` (ex. 06) acaba tendo o mesmo efeito do `\relative c'` (ex. 08). Para não criar confusão nem dúvidas, eu aconselho sempre usar o `\relative` seguido do registro da oitava que se pretende.

Exemplo 9

```
\relative c' {  
c d c e  
c f c g  
c a c b  
}
```



Este exemplo 9 mostra bem como é que o Lilypond entende o comando `relative`. No primeiro compasso, a nota ré (segunda nota do compasso) mais próxima do dó³ (primeira nota do compasso) é um intervalo de segunda, ou seja, o ré³. Da mesma forma, o mi (quarta nota do compasso) mais próximo do dó³ é o mi³ uma terça acima. O mesmo ocorre com o fá do segundo compasso. Todavia, o sol (quarta nota do segundo compasso) mais próximo do dó³ (terceira nota do segundo compasso) é o sol², uma quarta abaixo. Mesmo que a intenção musical fosse escrever um sol³, o Lilypond compara e percebe que o sol³ fica uma quinta justa do dó³, enquanto que o sol² fica uma quarta justa abaixo do dó³. Logo, o sol² é mais próximo do dó³ do que o sol³. O mesmo ocorre com o lá² (uma terça abaixo do dó³) e o si² (uma segunda abaixo do dó³).

Exemplo 10

```
\relative c' {  
c d c e  
c f c g'  
c, a' c, b'  
}
```



O Exemplo 10 mostra como podemos escrever corretamente o exemplo anterior. A partir dessa constatação podemos perceber que o comando `relative` procura notas até um intervalo de quarta da nota anterior. Se eu escrevo uma nota dó³, e em seguida quero um salto de sexta, ou seja, um lá³ (ver terceiro compasso), eu preciso escrever um apóstrofe na nota lá (a') para o programa pegar essa nota lá mais próxima do dó³, que seria um lá², e escrever uma oitava acima. Da mesma forma, depois desse lá³, se eu escrever um dó sem apóstrofe, ele iria interpretar como uma terça acima do lá³, isto é, um dó⁴. Por isso eu coloco uma vírgula no dó (terceira nota do quarto compasso) para ele escrever esse dó⁴ uma oitava abaixo, ou seja, dó³.

Exemplo 11

```
\relative c'' {  
c b a g  
f e d c  
}
```



Esse exemplo 11 faz uso do `\relative c''` para indicar que o dó inicial é o dó4.

Exemplo 12

```
\relative c''' {  
c b a g  
f e d c  
}
```



O exemplo 12, de forma similar ao exemplo 11, faz uso do `\relative c'''` (note os três apóstrofes) para indicar que o dó inicial é o dó5.

Exemplo 13

```
\relative c''' {  
c b a g | a c, d e | a, c' b a | g e, f g |  
}
```



O exemplo 13 é mais um exemplo do uso da vírgula e do apóstrofe para mudar a oitava da nota que seria escrita (pelo princípio da nota mais próxima relativa à nota anterior). Note que nesse exemplo foi inserido uma barra a cada quatro notas, simbolizando a barra de compasso. Na verdade, esse símbolo é desnecessário já que o Lilypond interpreta corretamente a posição das barras de compasso. Nesse caso serve mais como um auxílio para organizar o código de forma diferente.

Exemplo 14

```
\relative c' {  
c ees d fis | g a bes b | cis cisis dis e | ees eeses des c |  
}
```



O exemplo 14 introduz os comandos para escrever o símbolo de sustenido (is), o bemol (es), o duplo sustenido (isis) e o duplo bemol (eses). Não insira nenhum espaço entre a nota e sua alteração.

Exemplo 15

```
\relative c' {  
\key d \major  
d e fis g | a b cis d | c b a g | f e d c |  
}
```



Neste exemplo podemos notar o comando para mudar a tonalidade da música ‘\key d \major’. Nesse caso o comando \key (tonalidade em inglês) informa que vai haver uma mudança de tonalidade. A informação que segue diz qual a tonalidade, nesse caso o ré maior (\major significa maior em inglês. Se quiser uma tonalidade menor é só escrever \minor).

Note que no terceiro compasso aparece um dó bequadro. Quando olhamos o código percebe-se que o dó no terceiro compasso não tem nenhum sinal de alteração, e que o fá sustenido e o dó sustenido do primeiro e segundo compasso, respectivamente, precisaram que o código tivesse o complemento `is` para serem interpretados como tal. Daí aprendemos que o Lilypond não assume que, uma vez que você diz que a tonalidade é ‘x’, ele passa a colocar automaticamente os acidentes da armadura em cada nota. Na verdade, o comando `\key d \major` só faz imprimir a armadura de clave, mas as notas da melodia precisam de sustenidos ou bemóis para serem interpretadas como tal, senão serão interpretadas como sendo naturais, fazendo o Lilypond inserir bequadrinhos onde for necessário.

Exemplo 16

```
\relative c' {  
  \key d \major  
  d fis a bes | d c b! a! | cis d c b | cis! d e d |  
}
```



Este exemplo 16 introduz o comando ‘!’. Esse sinal de exclamação depois da nota “exige” que o Lilypond escreva o acidente da nota em questão. Perceba que no segundo compasso tem um dó natural, ou seja, um dó com um bequadro. No terceiro compasso volta a ser dó sustenido, mas como está na armadura, não há nenhum sinal na nota. Lembre-se que qualquer alteração só vale dentro do compasso para aquela oitava específica. No quarto compasso eu “exijo” que o Lilypond escreva o acidente do dó, mesmo que esteja na armadura. O mesmo vale para o bequadro no si do segundo compasso. Essa é uma boa solução para alguns casos, mas na maioria das vezes o que a gente está procurando é um sinal de precaução, o que é mais elegantemente escrito como no exemplo 17.

Exemplo 17

```
\relative c'' {  
  \key aes \major  
  \time 3/4  
  aes bes c | d f ees! | c des? d | ees? e fis |  
}
```



Neste exemplo o acidente de precaução, solicitado pelo comando ‘?’ (um acidente entre parênteses) no terceiro e quarto compasso, está misturado com o acidente “exigido” pelo comando ‘!’. Convenhamos que o símbolo entre parênteses é muito mais elegante que o símbolo sozinho, uma vez que ele já se encontra na armadura de clave. Outra informação nova nesse exemplo é o comando para alterar o sinal de compasso com o comando `\time 3/4`. Neste caso, é importante notar o uso da barra comum / na fração numérica que representa o sinal de compasso desejado.

Exemplo 18

```
\relative c'' {  
  \key g \major  
  \time 6/8  
  e8 d c d e fis | e g fis a4. | b8 a16 g fis8 e4. |  
}
```



Neste exemplo vemos pela primeira vez como mudar a figura rítmica. Como dito anteriormente, o Lilypond tem como padrão a semínima. Se quisermos mudar a figura, é preciso informar ao programa. E isso é feito de forma fácil e simples. Cada figura é representada pelo número de sua fração em relação à semibreve. Logo, a semibreve = 1, mínima = 2, semínima = 4, colcheia = 8, e assim por diante. Outra informação interessante é que, uma vez que você informa que mudou de figura rítmica, o programa irá permanecer naquela figura até você informar uma nova mudança. Se você observar atentamente ao código, o número 8 só aparece na primeira nota do primeiro compasso, e as notas seguintes não têm nenhum número associado. A próxima mudança só ocorre na quarta nota do segundo compasso, com o número 4 seguido de um ponto anexados à nota lá. Note que o ponto funciona da mesma forma que na escrita convencional, adicionando metade do valor à figura correspondente, nesse caso, uma semínima pontuada.

Exemplo 19

```
\relative c {  
  \key e \major  
  \time 2/4  
  \clef bass  
  e fis8 a | gis4 r8 dis | e4 r | r8. fis16 e4 |  
}
```



Neste exemplo aparece o 'r', o comando referente à pausa (rest em inglês). Perceba que assim como uma nota comum, ele mantém o valor da figura precedente (terceiro compasso), ou é alterado à sua vontade (segundo e quarto compassos).

Exemplo 20

```
\relative c' {
\key f \major
\time 2/2
\clef alto
c2( d | e1) | r2 f( | e) d ~ | d1 ~ | d2 bes |
}
```



Este exemplo apresenta dois símbolos novos: 1) a ligadura de expressão ‘(‘)’, 2) a ligadura de valor ‘~’. A ligadura de expressão à vezes causa uma certa confusão, pois o símbolo que inicia a ligadura ‘(‘)’ deve vir após a nota que ela vai tocar, e não antes da nota. Já o símbolo que finaliza a ligadura de expressão ‘)’ deve vir depois da nota que ela finaliza. Observe na partitura que a ligadura começa na primeira nota (dó), mas no código o símbolo vem depois da nota.

Exemplo 21

```
\relative c {
\key bes \major
\time 3/4
\clef bass
f( ees d)
ees8 f( g16 a bes c e4)
d2. \bar "||"
\key d \major
\time 2/4
\clef tenor
r4 d8 c
b4 cis8 d ~
d4 fis
d2 \bar "|."
}
```



Esse é o último exemplo, um pouco mais complexo que os demais, com quase todos os comandos mostrados até então. Note que, após o terceiro compasso, há uma mudança de tonalidade, compasso e de clave, tudo isso bem claro no código. O único elemento novo é a barra dupla e a barra final:

`\bar "||"` = barra dupla
`\bar "|."` = barra final

A partir de agora você já está apto a escrever melodias simples no Lilypond. Lembre-se que a prática leva à perfeição, ou no nosso caso, à memorização dos comandos. Boa sorte.

4 Como evitar os erros mais comuns

1. Abrir e fechar chaves ‘{ }’ (lembrar de escrever o código dentro das chaves).
2. Dar espaço entre uma nota e outra. Se escrever duas notas grudadas uma na outra (dd ou cg) o programa dá erro e não gera o arquivo PDF.
3. Escrever alguma nota com a letra maiúscula.
`{ A C D }` = errado
`{ a c d }` = certo
4. Usar a barra invertida (`\`) para invocar algum comando do Lilypond. Não use a barra normal (`/`).
5. Não dar espaço entre a barra invertida e o comando do lilypond
`\ relative` = errado
`\relative` = certo
6. A ordem de escrita da nota e suas propriedades deve ser sempre a seguinte: nota+alteração+oitava+figura rítmica (ex. `cis'16`). Qualquer mudança nessa ordem vai dar erro no programa (ex. `c16'`).
7. Antes de compilar o código tenha certeza de que fechou o arquivo .pdf. Se tentar compilar o arquivo Lilypond com o arquivo pdf aberto vai dar erro.

5 Resumo dos códigos

(dó = c, ré = d, mi = e, fá = f, sol = g, lá = a, si = b)
\clef treble (clave de sol, padrão)
\clef bass (clave de fá)
\clef alto (clave de dó na 3° linha)
\clef tenor (clave de dó na 4° linha)
\time 3/4 (sinal de compasso)
\key d \major (armadura de clave)
\bar "||" (barra dupla)
\bar "|." (barra final)
is = sustenido
es = bemol
isis = duplo sustenido
eses = duplo bemol
! = mostra o acidente
? = acidente de precaução
1 = semibreve
2 = mínima
4 = semínima
8 = colcheia
(= início da ligadura de expressão
) = final da ligadura de expressão
~ = ligadura de valor