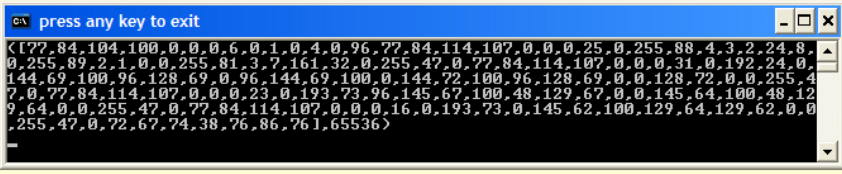


```

9 Start proc
10 #musica = [ [ 96, // ppq
11              3, // numerador da formula de compasso
12              4, // denominador da formula de compasso
13              2, // armadura de clave, sol maior(tipo 2- ver tabela)
14              120], // metrônomo
15
16              [[1,1,0,0], // nota->1, acorde ->0 ou 2
17              [24,24,24,24], // instrumento [0..127]
18              [4,4,4,4], // figura [1..]
19              [69,500,69,72], // nota [0..127], pausa -> 500
20              [100,100,100,100], // volume [0..127]
21              [0,0,0,0] // canal [0..15]
22            ],
23            [[1,1,1,1],
24            [73,73,73,73],
25            [4,8,8,4],
26            [500,67,64,500],
27            [100,100,100,100],
28            [1,1,1,1]
29            ],
30            [[1,1],
31            [73,73],
32            [2,4],
33            [62,500],
34            [100,100],
35            [1,1]
36            ] ]
37
38
39
40
41 # listaMIDI = geraMidiF1 musica
42 # arqSalvar = map toChar listaMIDI
43 # vetorSalva = {x\\x<-arqSalvar}
44 # (_,file3,proc)= fopen "C:\\geraMidiF1.mid" FWriteData proc
45 # file3 = fwrites vetorSalva file3
46 # (_,proc)= fclose file3 proc
47 # (_,proc)= OSStopMusic proc
48 # (_,proc)= playMid "C:\\geraMidiF1.mid" proc
49 = (listaMIDI,proc)

```



Listagem 6.15 – Programa que utiliza a função geraMidiF1, e resultado.

O código 2 para tonalidade equivale à tonalidade de Sol Maior. Este código foi definido na função **mostraTonalidade**. Esta função tem como argumento este código (quarto elemento do cabeçalho fornecido, e devolve o código do metaevento de armadura de clave equivalente (255 88- FF 58). A seguir é mostrada a função **mostraTonalidade**.

```

381 mostraTonalidade :: .Int -> [.Int]
382 mostraTonalidade keysign
383 |keysign == 0 = [0,255,89,2,0,0]// "DoM"
384 |keysign == 1 = [0,255,89,2,0,1]// "Lam"
385 |keysign == 2 = [0,255,89,2,1,0]// "SolM"
386 |keysign == 3 = [0,255,89,2,1,1]// "Mim"
387 |keysign == 4 = [0,255,89,2,2,0]// "ReM"
388 |keysign == 5 = [0,255,89,2,2,1]// "SiM"
389 |keysign == 6 = [0,255,89,2,3,0]// "LaM"
390 |keysign == 7 = [0,255,89,2,3,1]// "Fa#m"
391 |keysign == 8 = [0,255,89,2,4,0]// "MiM"
392 |keysign == 9 = [0,255,89,2,4,1]// "Do#m"
393 |keysign == 10 = [0,255,89,2,5,0]// "SiM"
394 |keysign == 11 = [0,255,89,2,5,1]// "Sol#m"
395 |keysign == 12 = [0,255,89,2,6,0]// "Fa#M"
396 |keysign == 13 = [0,255,89,2,6,1]// "Re#m"
397 |keysign == 14 = [0,255,89,2,7,0]// "Do#M"
398 |keysign == 15 = [0,255,89,2,7,1]// "La#m"
399
400 |keysign == 16 = [0,255,89,2,255,0]// "FaM"
401 |keysign == 17 = [0,255,89,2,255,1]// "Rem"
402 |keysign == 18 = [0,255,89,2,254,0]// "SibM"
403 |keysign == 19 = [0,255,89,2,254,1]// "Solm"
404 |keysign == 20 = [0,255,89,2,253,0]// "MibM"
405 |keysign == 21 = [0,255,89,2,253,1]// "Dom"
406 |keysign == 22 = [0,255,89,2,252,0]// "LabM"
407 |keysign == 23 = [0,255,89,2,252,1]// "Fam"
408 |keysign == 24 = [0,255,89,2,251,0]// "RebM"
409 |keysign == 25 = [0,255,89,2,251,1]// "Sibm"
410 |keysign == 26 = [0,255,89,2,250,0]// "SolbM"
411 |keysign == 27 = [0,255,89,2,250,1]// "Mibm"
412 |keysign == 28 = [0,255,89,2,249,0]// "DobM"
413 |keysign == 29 = [0,255,89,2,249,1]// "Labm"
414 |otherwise = [0,255,89,2,0,0]

```

Listagem 6.16 – Função “mostraTonalidade”

6.4 Implementação de um Leitor MIDI formato 0 ou 1 e Conversor Didático de arquivos MIDI escritos em Hexadecimal e Decimal para SMF formato 0 ou 1.

1- PROBLEMA:

Existe no mercado, uma carência de uma ferramenta didática que facilite o aprendizado, em baixo nível, do protocolo MIDI, e, em especial, os SMF (Standard MIDI Files), bem como manipular arquivos MIDI existentes. O que se faz normalmente é utilizar um editor binário qualquer, o qual não é aderente às reais necessidades de um usuário que está querendo aprender a ler e escrever arquivos MIDI SMF. Assim, implementou-se, como estudo de caso e como ferramenta aplicativa, um software dedicado a este fim, ou seja:

1. que permita abrir um arquivo MIDI e exibí-lo em Hexadecimal ou Decimal;
2. que permita editar esse arquivo (modificando-o ou iniciando o mesmo a partir do zero);
3. inserir comentários e explicações, permitindo salvar o mesmo como texto para reedição;
4. que permita salvar o arquivo criado em baixo nível (em hexadecimal) pra arquivo MIDI SMF formato 0 ou 1;
5. que permita ler do arquivo MIDI os parâmetros do cabeçalho (ppq, fórmula de compasso, tonalidade, nr de tracks);
6. que permita ler um SMF formato 0 ou 1 em formato textual, cujo texto é o arquivo MIDI apresentado em canais e dividido, cada canal, em forma de compassos musicais, de acordo com a fórmula de compasso do arquivo;
7. que permita ao usuário converter ppq em delta time para escrita de seus arquivos SMF em baixo nível e vice versa (delta time para ppq);
8. que possua um player MIDI para que se possa ver o resultado dos trabalhos ou para escutar um arquivo SMF existente.

2- Solução: O APLICATIVO “CONVERSOR DIDÁTICO MIDI”

Este aplicativo tem como objetivo implementar um sistema que efetive as ações citadas nos problemas supra mencionados neste estudo de casos.

A interface do aplicativo: A figura 6.19 mostra a interface do aplicativo “Conversor didático MIDI”. Na mesma estão listadas as diversas potencialidades que a mesma apresenta para um estudante do protocolo MIDI e criação de SMF de qualquer formato.

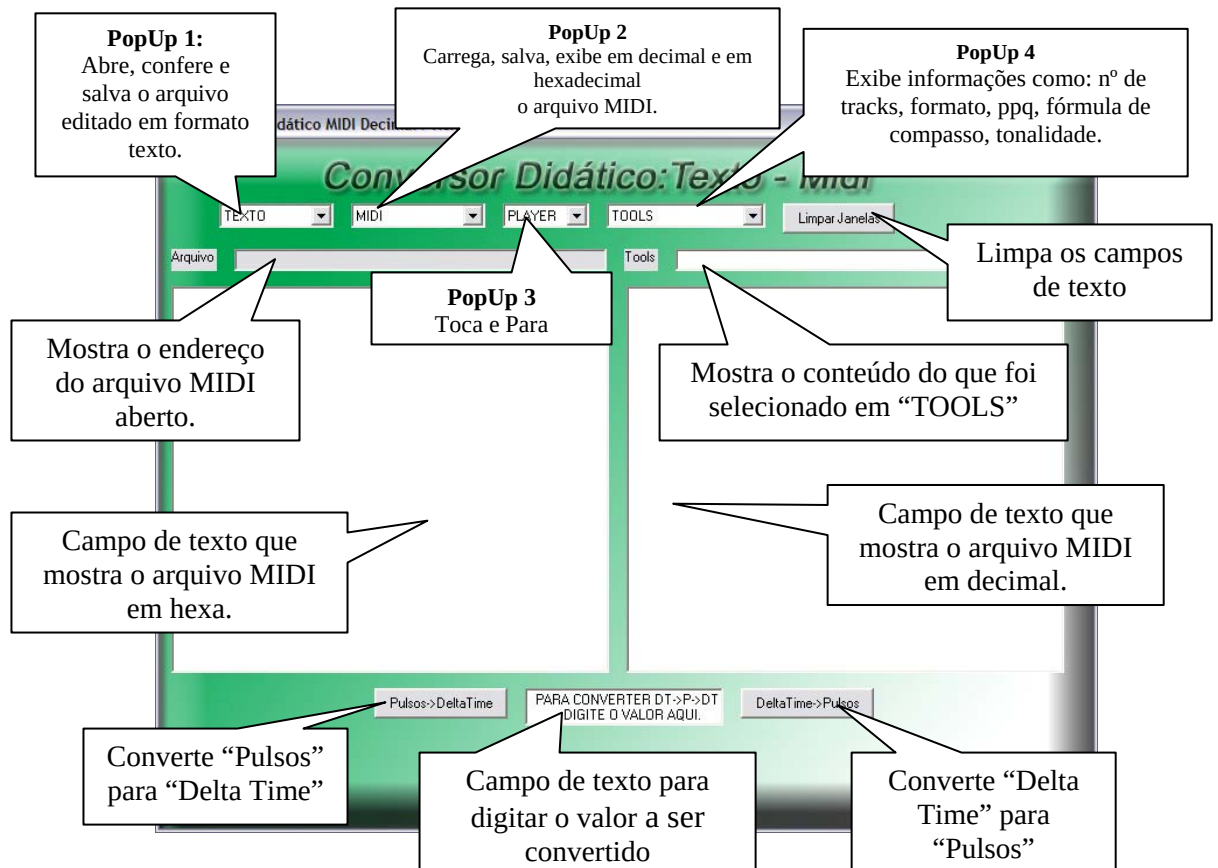


Figura 6.19 – Interface do Conversor Didático

6.4.1 Potencialidades das ferramentas da interface.

Algumas das potencialidades do aplicativo são disponibilizada no PopUp 2, o qual permite que se escolham algumas ações, tal como: carregar um arquivo MIDI na memória do computador para posterior manipulação, abrir o arquivo na janela de texto em formato Hexadecimal ou em Decimal e salvar o arquivo editado em formato MIDI.

6.4.2 PopUp 1 (Texto): Ações.

Este popUp disponibiliza as ações de abrir um texto (em especial, arquivos editados anteriormente e salvos como texto). Permite conferir se os Bytes hexadecimais estão bem formados (se possuem dois símbolos hexadecimais → 0,1,2,...,A,B,...,F) e permite salvar a janela editada no campo em Hexadecimal, para que se possa editá-lo posteriormente. A figura 6.20 mostra as ações deste popUp.

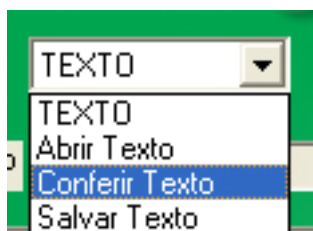


Figura 6.20 – Popup de abertura de textos MIDI

6.4.3 PopUp 2 (MIDI) -Ações:

1-Para ativar uma das ações deste popUp, basta clicar no respectivo item, como mostrado no exemplo da figura 6.21 com a ação de abrir e exibir um arquivo MIDI em Hexadecimal ou Decimal. Estas duas ações, bem como qualquer outra que trabalhe com arquivo MIDI, exigem, primeiro, que o arquivo MIDI seja aberto e armazenado na memória do computador.

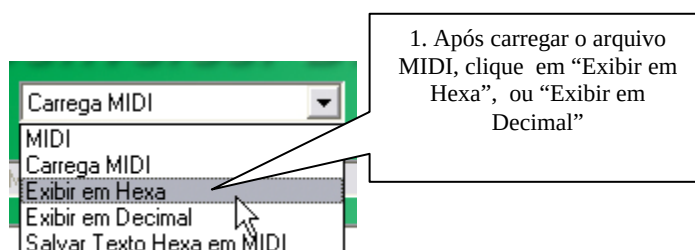


Figura 6.21 – Abrindo arquivos MIDI em Hexadecimal

A figura 6.22 mostrar o arquivo aberto em Hexadecimal e decimal.



Figura 6.22 – Arquivo MIDI “cap2f1.mid” aberto em decimal e hexadecimal

- 2- Após digitar o texto, inclusive com comentários pertinentes, pode-se convertê-lo para MIDI. Os comentários devem vir entre os sinais: (* e *). A figura 6.23 ilustra esta ação.

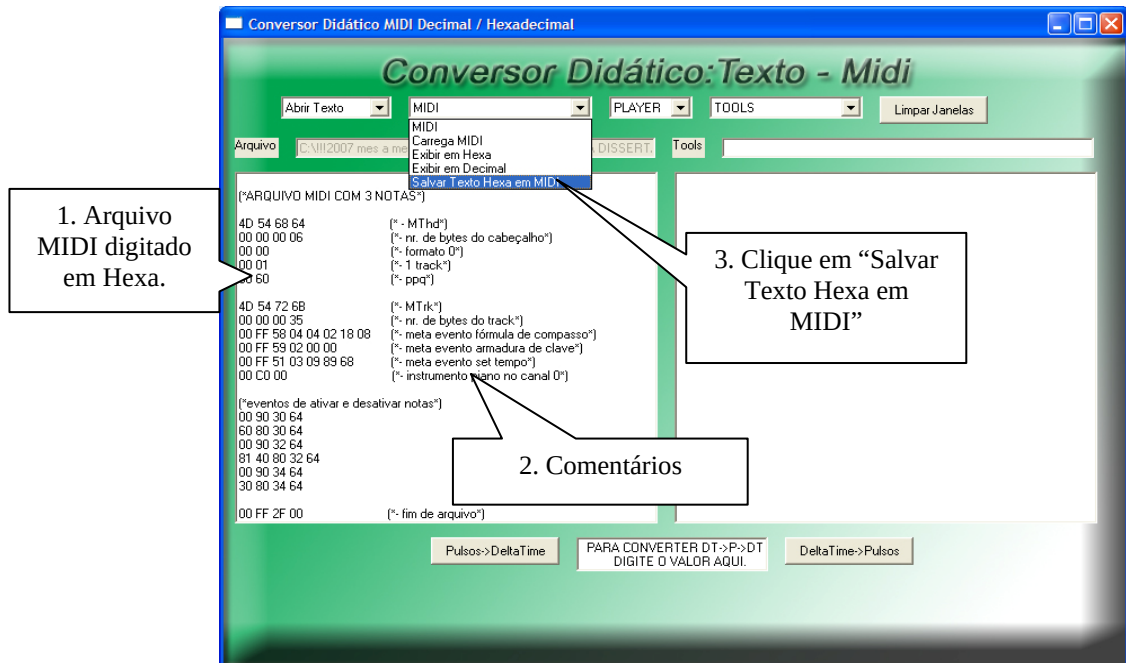


Figura 6.23 – Arquivo MIDI digitado com comentários.

6.4.4 PopUp 3 (PLAYER): Ações.

Este popUp permite se tocar e interromper a execução de um arquivo MIDI, conforme mostram os itens da figura 6.24

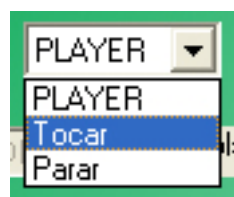


Figura 6.24 – Popup de tocar e parar um arquivo MIDI

6.4.5 PopUp 4 (TOOLS): Ações

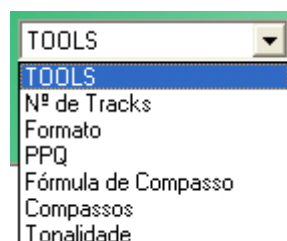


Figura 6.25 – Popup de ferramentas do Conversor Didático

Este popUp, mostrado na figura 6.25, além de permitir ler dados do cabeçalho do arquivo e alguns dados musicais, possui uma ação importante:

- permite abrir um arquivo MIDI em formato de compassos e notas musicais.

Isto, para um músico e para o programador não músico, é uma opção fundamental, algo que não se consegue com um editor binário existente. Assim, pode-se analisar o arquivo em uma notação que tanto o músico quanto o programador entendam, já que ler partituras musicais é uma tarefa complexa que demanda um bom tempo de aprendizado musical. Para tanto, basta escolher no popUp 4 a ação: **Compassos** (depois de já se ter aberto o arquivo MIDI na opção **Carrega MIDI** do popUP 2, conforme já descrito anteriormente).

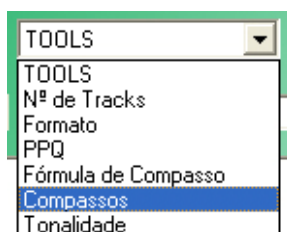


Figura 6.26 – Ferramenta “Compassos”

A figura 6.27, a seguir, mostra o arquivo “cap2f1.mid”, do CD em anexo, utilizado como exemplo deste estudo de caso. O mesmo é aberto pelo Conversor MIDI em texto hexadecimal, decimal e em forma de compassos. Os resultados obtidos pelo Conversor Didático podem ser comparados com a partitura aberta pelo programa Finale para validação dos resultados.

COMPASSOS

```

0
{ la3 (440Hz) sm , p sm , la3 (440Hz) sm - do4 sm }
1
{ p sm , sol3 c , mi3 c , re3 m , p sm }

```

Track 0-1

Track 0-2

Figura 6.27 – Comparação do arquivo aberto no Conversor com o software Finale

6.4.6 Calculadora: Delta Time <-> PPQ.

Esta ferramenta foi incluída neste aplicativo para permitir ao usuário aprender, quantificar o valor correto, em inteiro, da duração de cada evento (mensagem MIDI), bem como, tendo um valor de um delta time em inteiro (uma ppq), convertê-lo para delta time de até quatro bytes, o que é necessário para se editar um arquivo SMF corretamente. A figura 6.28 mostra dois exemplos: um de conversão de inteiro para delta time e outro de delta time para inteiro.

6.4.6.1- Convertendo ppq para DeltaTime.

Para tanto, siga os passos a seguir:

- Digite o valor em delta time e converta para ppq, conforme figura 6.28.

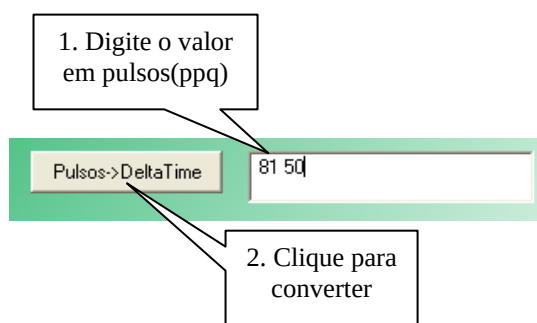


Figura 6.28 – Digitando o valor em delta time e convertendo para ppq

- Feito isto, o resultado em Delta time é mostrado juntamente com o valor original em ppq (em forma de comentário).

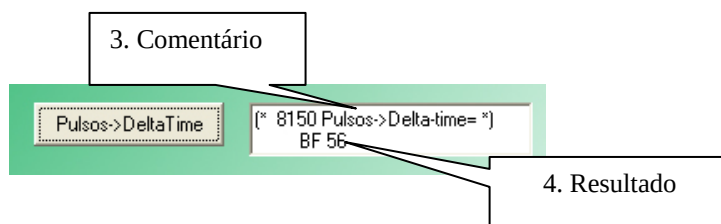


Figura 6.29 – Resultado da conversão de ppq para delta time

6.4.6.2 - Convertendo Delta Time para ppq (pulsos).

Para tanto, siga os passos a seguir:

- Digite o valor em ppq (pulsos) e converta para delta time, conforme figura 6.30

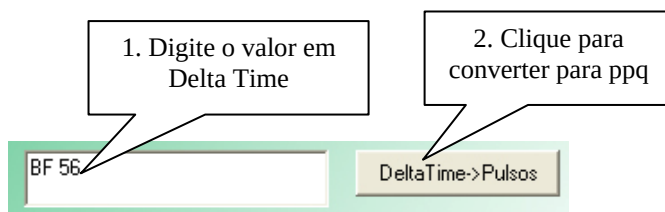


Figura 6.30 – Convertendo delta time para ppq

- Feito isto, o resultado em ppq é mostrado, juntamente com o valor original em delta time (em forma de comentário).

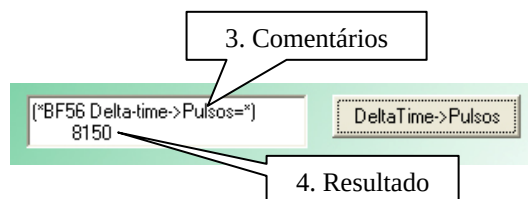


Figura 6.31 – Resultado da conversão de delta time para ppq

Observe que depois da conversão, fica um comentário do que foi feito entre os delimitadores (* e *). Desta forma, pode-se realizar a operação inversa sem ter que apagar o comentário, como mostra a figura 6.32.

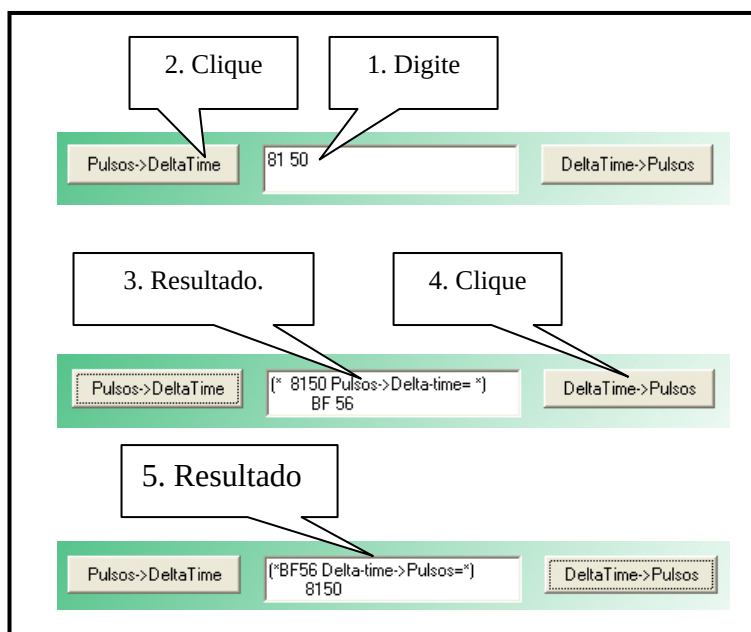


Figura 6.32 – Sequência de ações para conversão delta time $\Leftrightarrow$ ppq

6.5 Implementação de um Editor de Partituras Multi-Instrumental e Multi-Canal. Com salvamento em formato proprietário e em SMF formato 1.

Um dos propósitos do desenvolvimento deste editor é mostrar como utilizar em conjunto os conceitos, funções aplicativos desenvolvidos nos capítulos anteriores, assim como, também, mostrar os princípios básicos necessários para se implementar um editor de partituras multi-instrumental e polifônico.

No mesmo são mostrados os vários recursos utilizados em editores profissionais, tais como utilização de teclado virtual para entrada de notas em pentagramas, como utilizar o ponteiro do mouse para manipular notas, como, por exemplo, o ato de apagar uma nota ao clicar na mesma, como criar barra de menus com ícones para escolha de figura musical, para um *player* MIDI, para mudança de instrumento, volume e outros.

O editor também permite o salvamento em arquivo MIDI SMF formato 1 e em formato proprietário (.bhl). Não se implementou a leitura de arquivo MIDI no editor, apesar de já se ter feito tal função, conforme já mostrado no capítulo anterior, bastando apenas formatar os dados de entrada de acordo com as listas do editor. No caso do arquivo proprietário, além de salvar o trabalho, pode-se abri-lo posteriormente para edição.

O editor apresenta, também, como fazer um *refresh* da tela e os campos de armazenamento das estruturas a serem salvas, tais como:

- Notas;
- Figuras musicais;
- Canal MIDI;
- Volume;
- Estrutura de nota e acorde;
- Instrumento por nota;
- Pentagrama;
- Pentagramas.

Este editor, de uma forma simples e direta, também permite:

- Que o usuário escolha para cada nota musical um instrumento diferente, mesmo em acordes;
- Que se escolha um canal MIDI para cada nota registrada.

Ações estas não comuns nos editores existentes.

Este aplicativo utiliza a função **geraMidiF1**, mostrada anteriormente, para a geração do arquivo MIDI SMF formato 1. A figura 6.33 mostra a interface do programa, e, na figura 6.34 a discriminação de suas partes componentes.

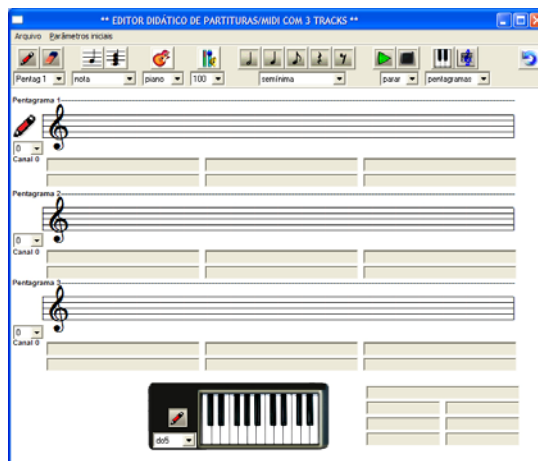


Figura 6.33 – Interface do editor didático de partituras

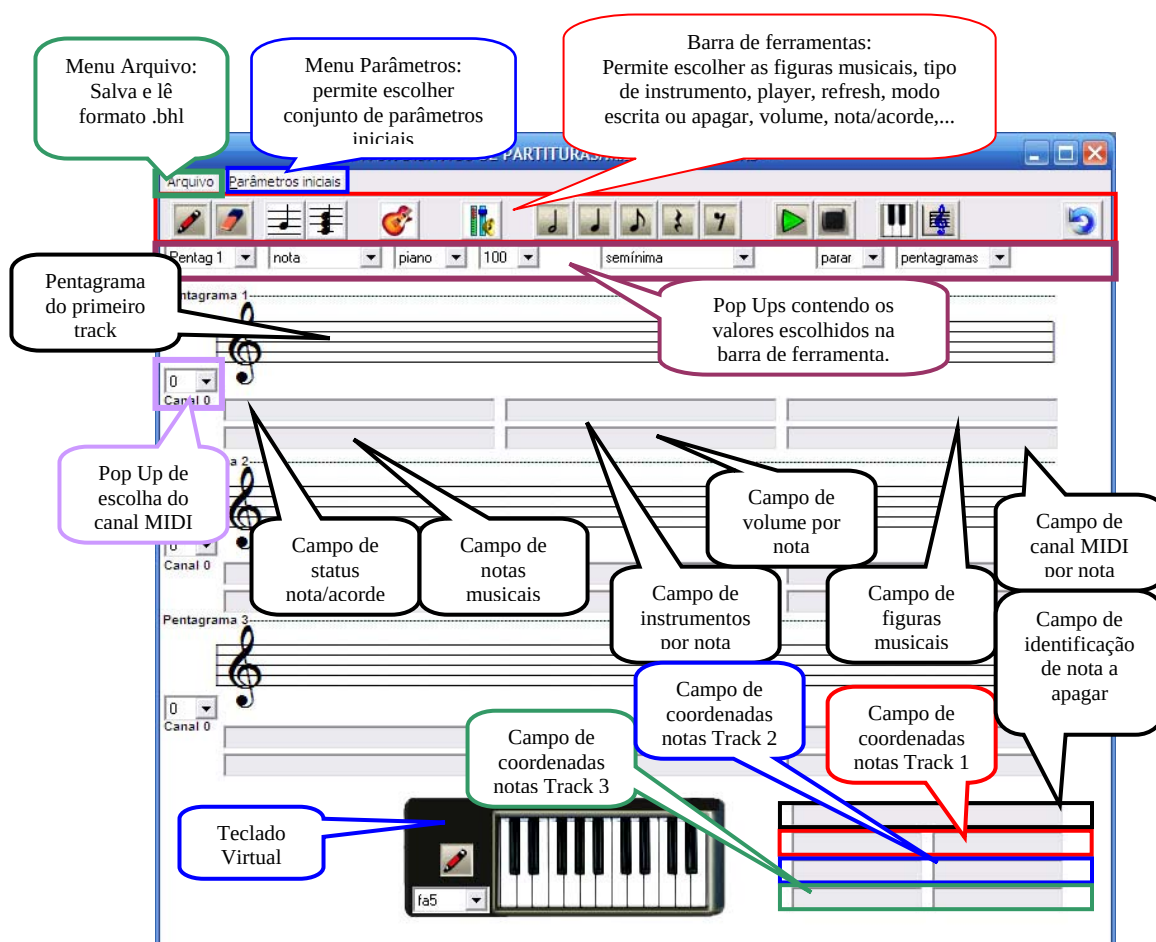


Figura 6.34 – Discriminação da Interface do editor didático de partituras

Como se pode perceber, esta interface possui praticamente todos os conceitos e recursos básicos necessários para se construir um editor de partituras profissional. Para se construí-la foram utilizados apenas os recursos disponibilizados pela própria linguagem e os conceitos e aplicativos apresentados e implementados neste capítulo. O código completo deste editor é apresentado no CD em anexo, bem como o aplicativo. O mesmo não foi colocado no corpo da dissertação por ser bastante extenso.

6.5.1 Descrição dos componentes deste editor:

1- Menu arquivo:

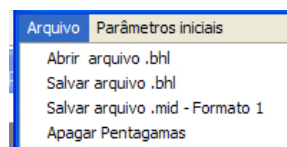


Figura 6.35 – Menu “Arquivo”

Neste Menu podem-se escolher as opções:

- Abrir um arquivo editado no formato .bhl (formato do editor);
- Salvar um arquivo no formato .bhl (este formato será mostrado no exemplo posterior);
- Salvar o trabalho em SMF formato 1;
- Apagar as edições feitas no primeiro Pentagrama⁶⁴

2- Menu Parâmetros iniciais.

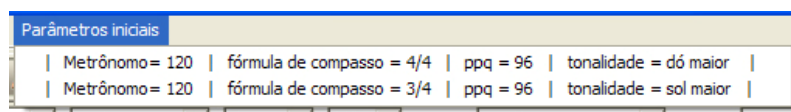


Figura 6.36 – Parâmetros iniciais

Neste menu foram colocadas inicialmente duas opções. Uma delas, a de fórmula de compasso $\frac{3}{4}$, para atender a configuração do exemplo dado no final do capítulo 2 para explicar os formatos 0 e 1.

3- Lápis e Borracha.

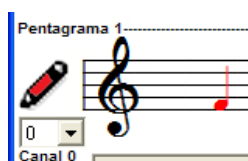
⁶⁴ Apagar notas nos demais pentagramas não foi implementado por ser apenas o trabalho de se copiar os mesmos procedimentos feitos para o pentagrama 1, devendo-se apenas variar a coordenada y do ponteiro do mouse e fazer a leitura nos campos de coordenadas das notas dos demais pentagramas.



Figura 6.37 – Lápiz e Borracha

Estas duas ferramentas, lápis e borracha definem o modo de edição. O menu pop-up logo abaixo aos ícones define em qual pentagrama será liberada a ação de edição escolhida. Quando o pentagrama é selecionado, o ícone de lápis ou de borracha é colocado à esquerda do mesmo indicando sua seleção e modo de edição.

Modo Escrita:



Modo Apagar:

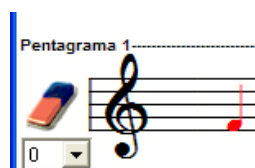


Figura 6.38 – Modos: Estrita e Apagar

4- Modo de edição de notas/Acordes.

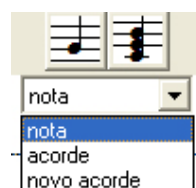


Figura 6.39 – Menu “nota”, “acorde”, “novo acorde”

Ao clicar no símbolo de uma nota, a edição no pentagrama passa a ser de nota em nota. Caso se clique no símbolo de acorde, as notas editadas formarão um acorde. Se desejar iniciar um novo acorde, a partir de um acorde, basta clicar no símbolo de acorde novamente que um novo acorde será iniciado.

5- Escolha de instrumento.

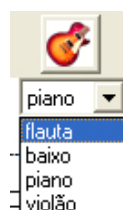


Figura 6.40 – Escolha do instrumento

Cada vez que se clica no símbolo do instrumento, um instrumento novo é selecionado. Como sempre, pode-se escolher o instrumento direto do pop up correspondente. Foram colocados inicialmente apenas quatro instrumentos, mas, de uma forma simples, poder-se-ia ter colocado todos os instrumentos previstos na especificação MIDI.

6- Escolha do volume.

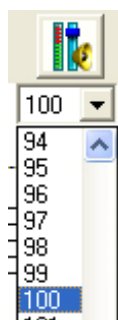


Figura 6.41 – Escolha do volume

Ao clicar no símbolo do *slider* (controle deslizante) de volume, o volume muda entre os valores: 0, 50, 100 e 127. Caso se deseje um volume diferente dos pré-configurados, basta escolher direto no popup equivalente.

7- Escolha da figura musical.



Figura 6.42 – Figura musical

Ao clicar em um dos símbolos de figuras musicais, o mesmo é selecionado para a nota que será inserida no pentagrama escolhido, ao clicar em uma tecla do teclado virtual (estando no modo de escrita)

8- Player.



Figura 6.43 – Player

Foram colocado dois botões no *Player*: um para tocar a música escrita no pentagrama. Para tanto, o sistema salva um arquivo temporário na raiz do disco C (geraMidiF1.mid) e depois o toca. O outro botão apenas interrompe a execução do arquivo.

9- Pentagrama/teclado.



Figura 6.44 – Pentagrama e Teclado

Estes dois botões permitem que seja impresso na tela, novamente os três pentagramas ou o teclado virtual. Os mesmos estão aí apenas de forma didática, para se mostrar como entrar com novos elementos na interface. Ao se minimizar a janela, necessita-se fazer um refresh na tela quando a mesma for maximizada novamente. Neste caso, se em vez de utilizar o botão de refresh (será visto logo a seguir) se clicar no símbolo do teclado ou do pentagrama, os mesmos serão inseridos na interface.

10- Atualização da janela (refresh).



Figura 6.45 – Botão Refresh

Ao se clicar neste botão, é feito uma atualização da janela. Isto é necessário quando se minimiza a janela. Ao se maximizar, o refresh não é feito automaticamente.

11- Escolha do canal por nota.



Figura 6.46 – Escolha do Canal

Este *popup*, à esquerda de cada pentagrama, permite a escolha do canal MIDI para cada nota que será inserida nos pentagramas.

12- Teclado Virtual.



Figura 6.47 – Teclado Virtual

A figura anterior mostra o teclado virtual com a tecla re5 pressionada, conforme mostra o menu *popup* à direita das teclas. O ícone do lápis acima do menu *popup* indica que o editor está no modo de escrita (no modo apagar este espaço fica preto, sem nenhum ícone, já que no modo apagar, este ato é feito pelo mouse e não pelo teclado. No modo apagar o teclado também fica desativado.

A figura 6.48, a seguir, mostra um exemplo de notas editadas em um pentagrama ainda no modo de escrita, como indica o ícone do lápis à esquerda do pentagrama. Observe que cinco notas foram inseridas no pentagrama e que cada campo abaixo do pentagrama registra as escolhas feitas pelo usuário para cada uma delas.

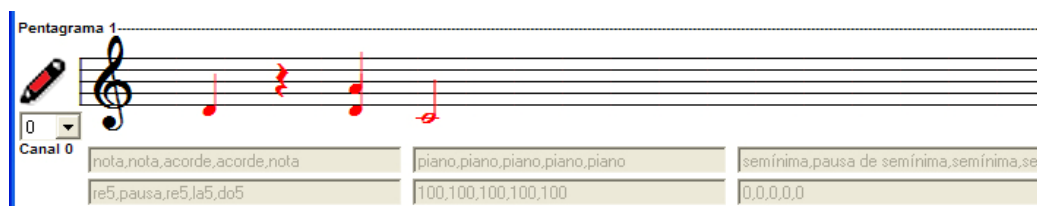


Figura 6.48 – Notas no pentagrama

Para que se possa utilizar este editor e avaliar a complexidade de uma interface deste tipo, a seguir é apresentado um exemplo de utilização deste editor:

Exemplo 1 : Escrevendo notas em linhas melódicas.

- Entre no modo de escrita, para tanto, siga as instruções das figuras a seguir

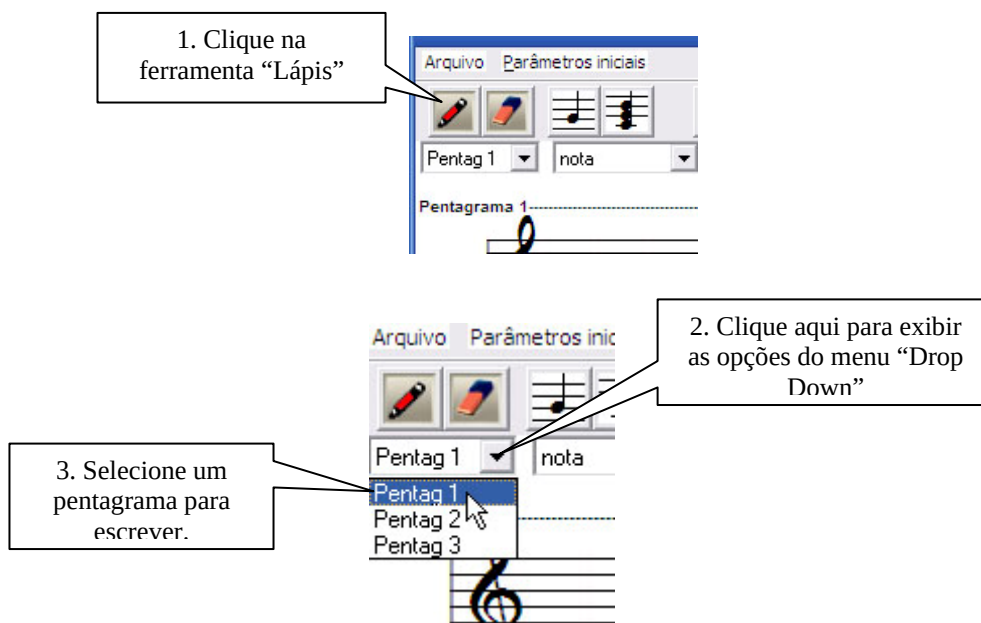


Figura 6.49 – Selecionar pentagrama

- Escolha o modo de edição (nota ou acorde), o tipo de instrumento, o volume da nota e a figura musical

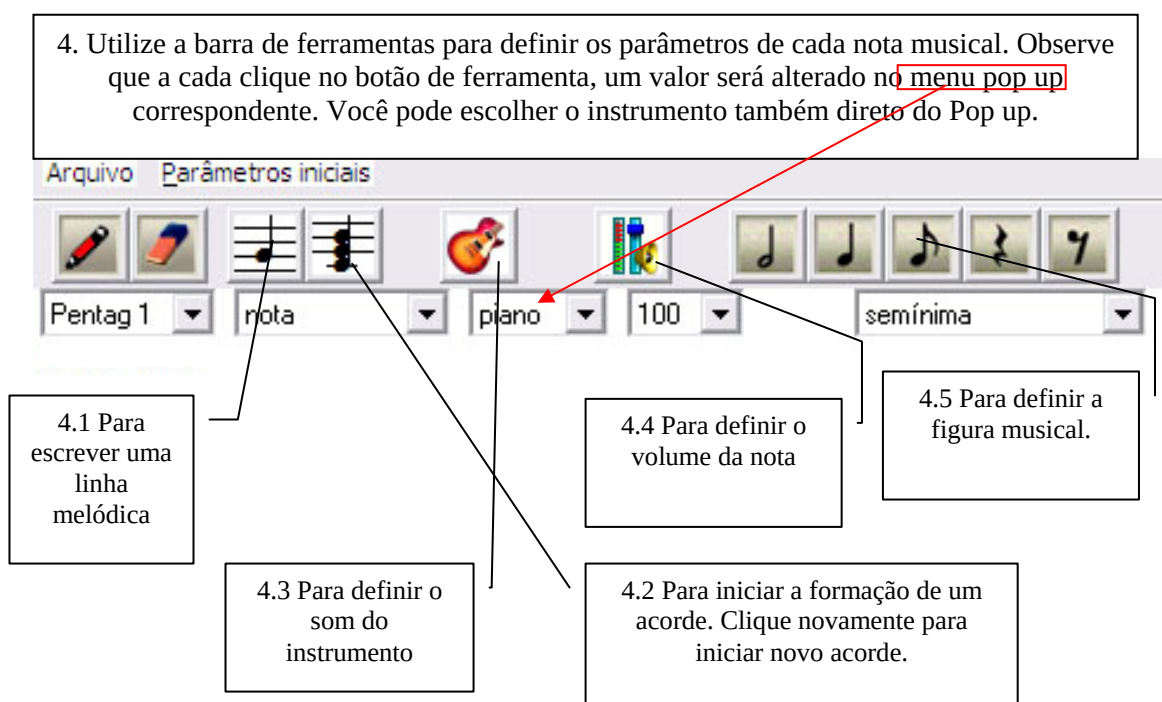


Figura 6.50 – Configurando o Editor

-Escolha o canal MIDI desejado para a nota ou para o pentagrama (se não for mudar)

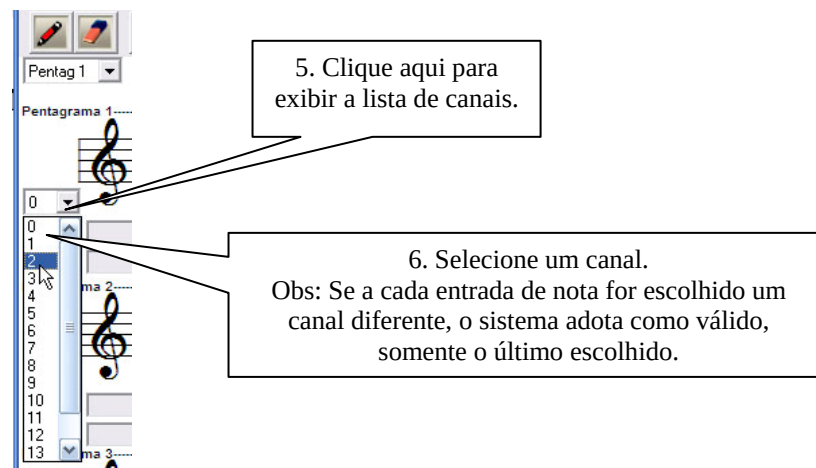


Figura 6.51 – Escolha do Canal

- Escolha a nota a ser grafada no pentagrama escolhido

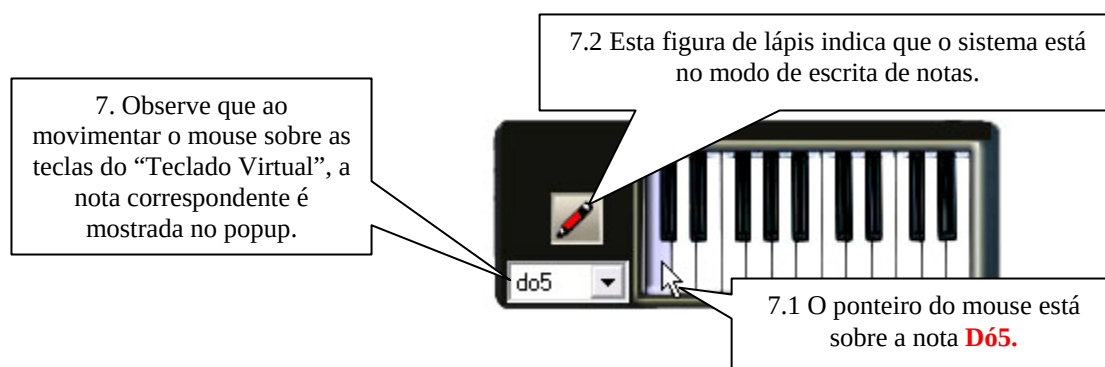


Figura 6.52 – Entrando com a nota via teclado virtual

- Ao fazer isto, a nota é inserida no pentagrama e os campos de registros dos parâmetros da nota são atualizados.

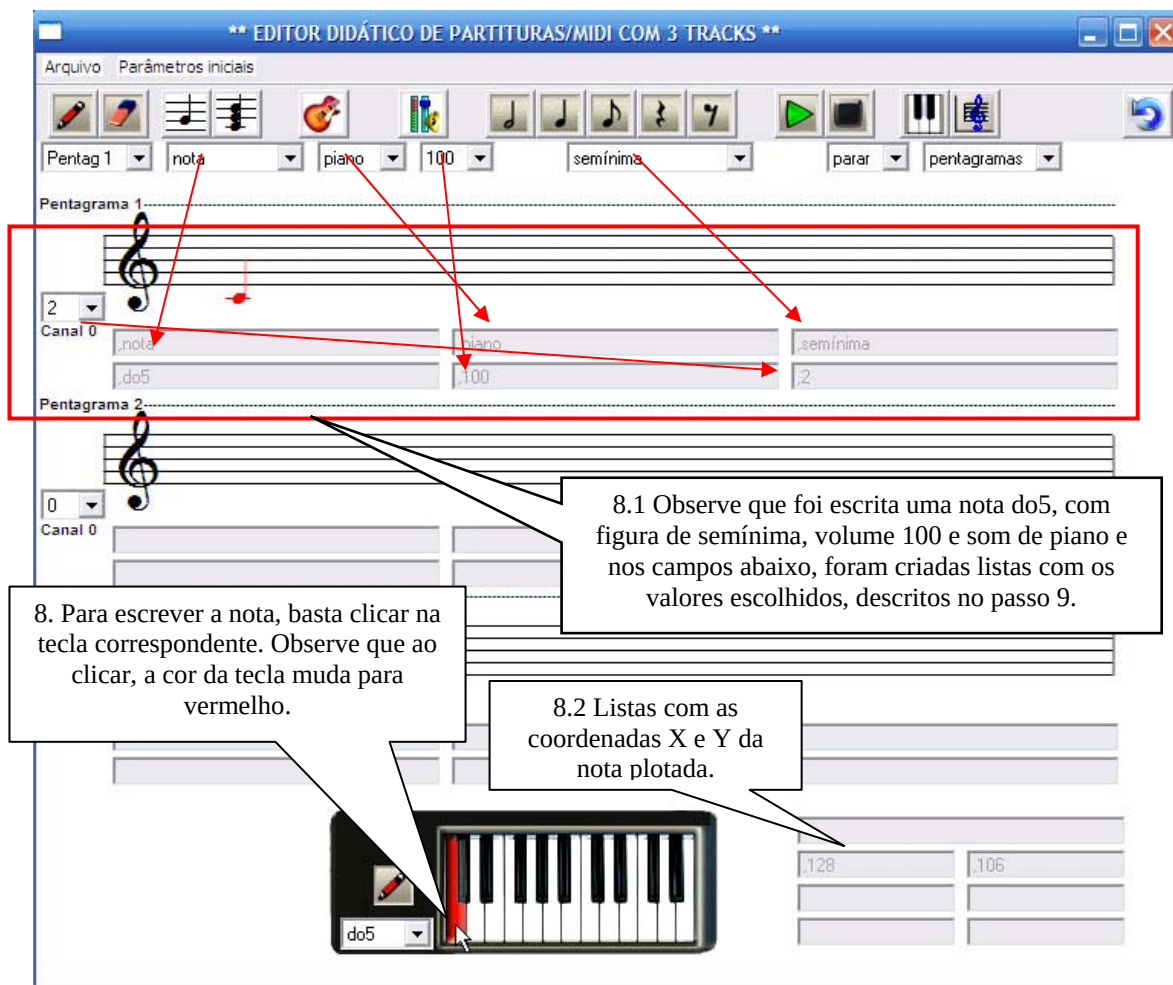


Figura 6.53 – Nota inserida e campos atualizados

- Observe com mais detalhes os campos atualizados com a inserção da nova nota musical

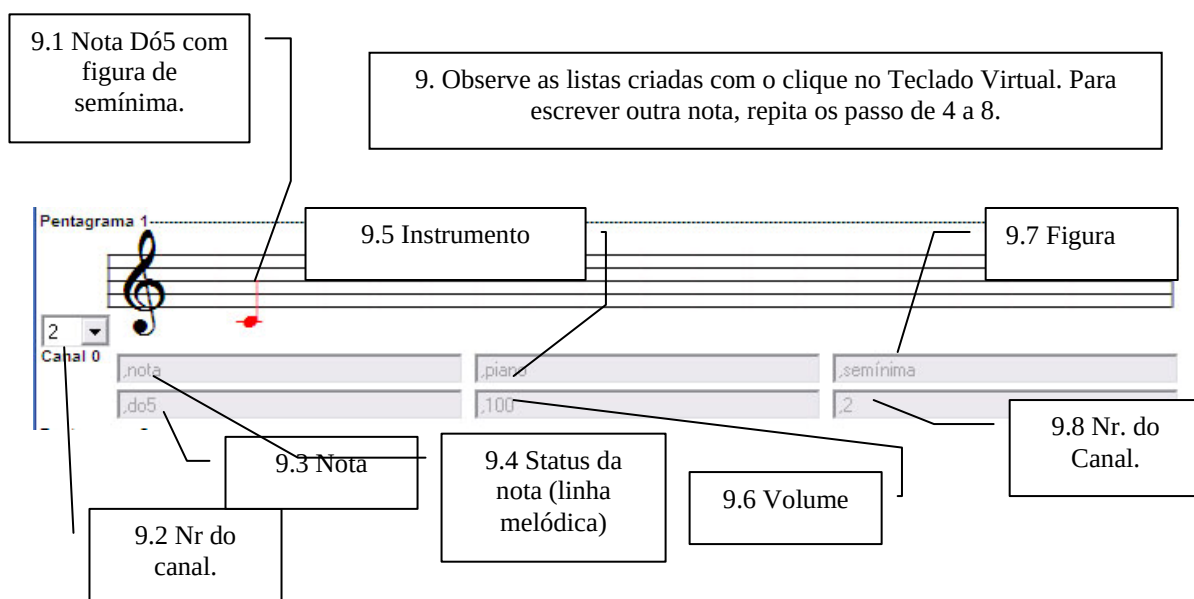


Figura 6.54 – Campos atualizados

- Entre agora com um acorde, para tanto, escolha o modo de escrita **acorde**.

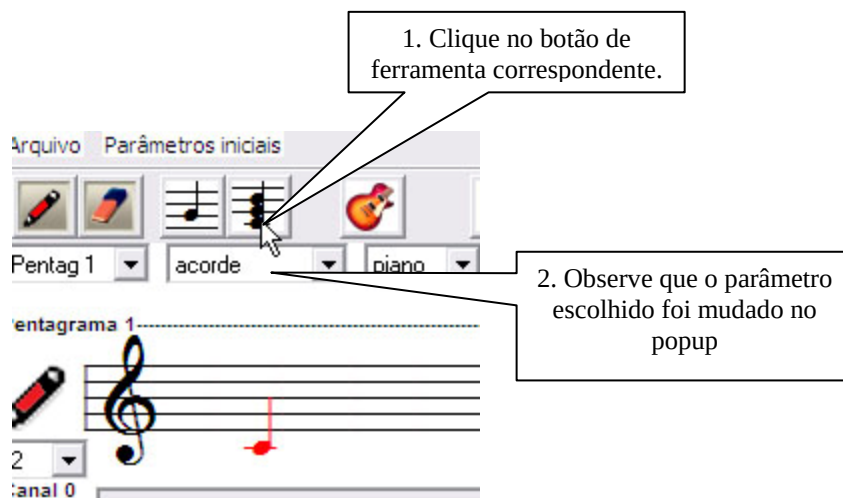


Figura 6.55 – Entrando com um acorde

- Entre com algumas notas, por exemplo: do5, mi5 e sol5, conforme figura a seguir

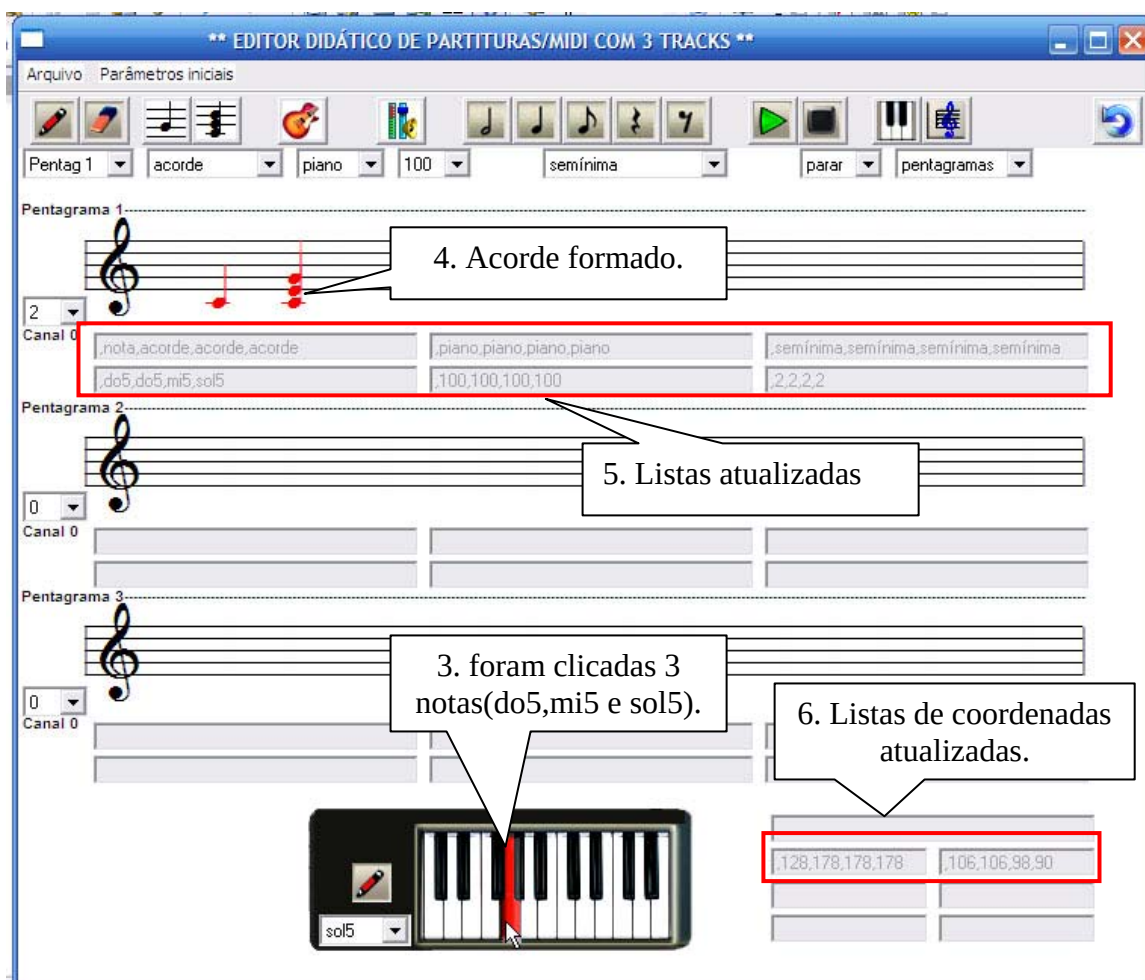


Figura 6.56 – Acorde formados e campos atualizados

- Entre com um novo acorde, para tanto, repita os dois passos anteriores. Neste modo, acorde, pausas serão recusadas.

- Ouvindo a música escrita.

Para tanto, basta clicar no botão de play para tocar e no de stop para parar.



Figura 6.57 – Play e Stop

- Salve o arquivo para posterior edição. Para tanto, ative o menu arquivo e clique na opção “Salvar arquivo .bhl”.

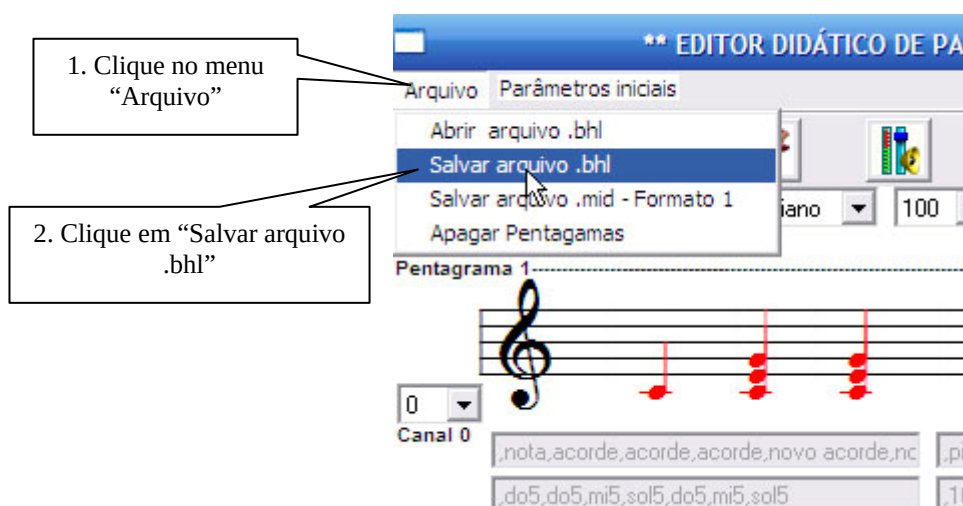


Figura 6.58 – Menu salvar em formato .bhl

- Ao fazer isto, a janela do explorador é aberta para que você escolha onde salvar sua música.

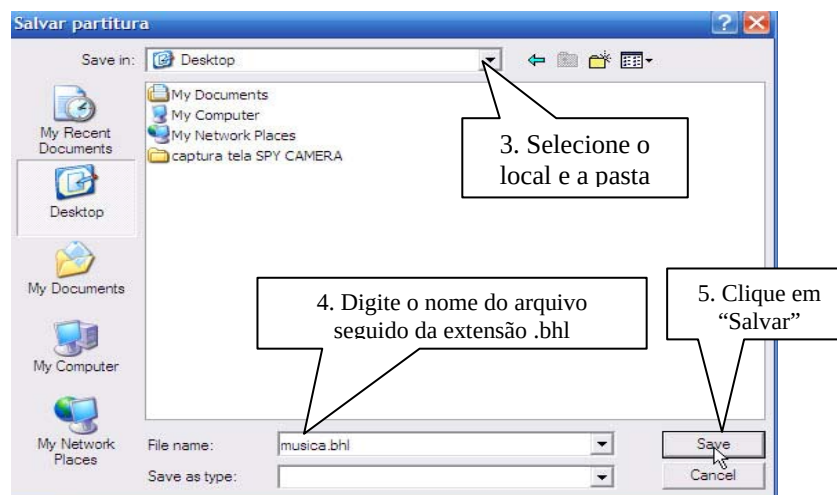


Figura 6.59 – Janela “Salvar partitura”

Um arquivo é salvo no formato texto, o qual permite ser editado de qualquer editor de texto. Na primeira linha do arquivo é colocada a string **BHL** para que o sistema teste se o arquivo ao ser aberto está no formato bhl, independentemente da extensão.

O arquivo bhl desta música é dado a seguir:

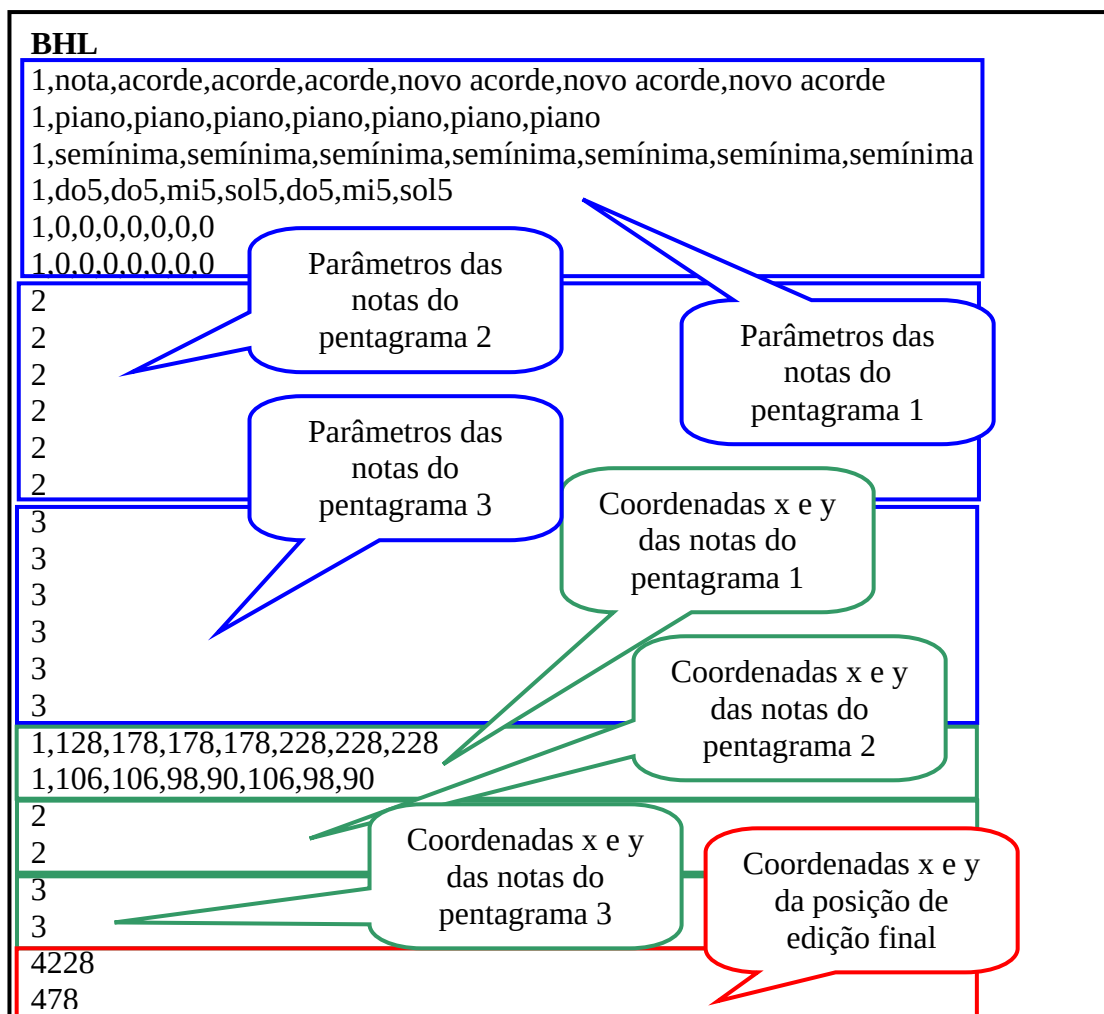


Figura 6.60 – Arquivo formato bhl

-Para apagar a edição feita, basta ativar novamente o menu Arquivo e clicar na opção “Apagar Pentagramas”.

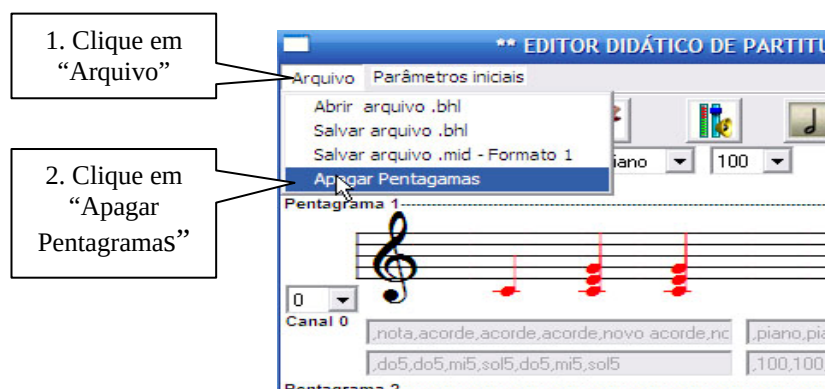


Figura 6.61 – Apagar pentagrama