

Construção de Instrumentos e Melodias/Efeitos

Introdução à Computação Sônica - Trabalho 2

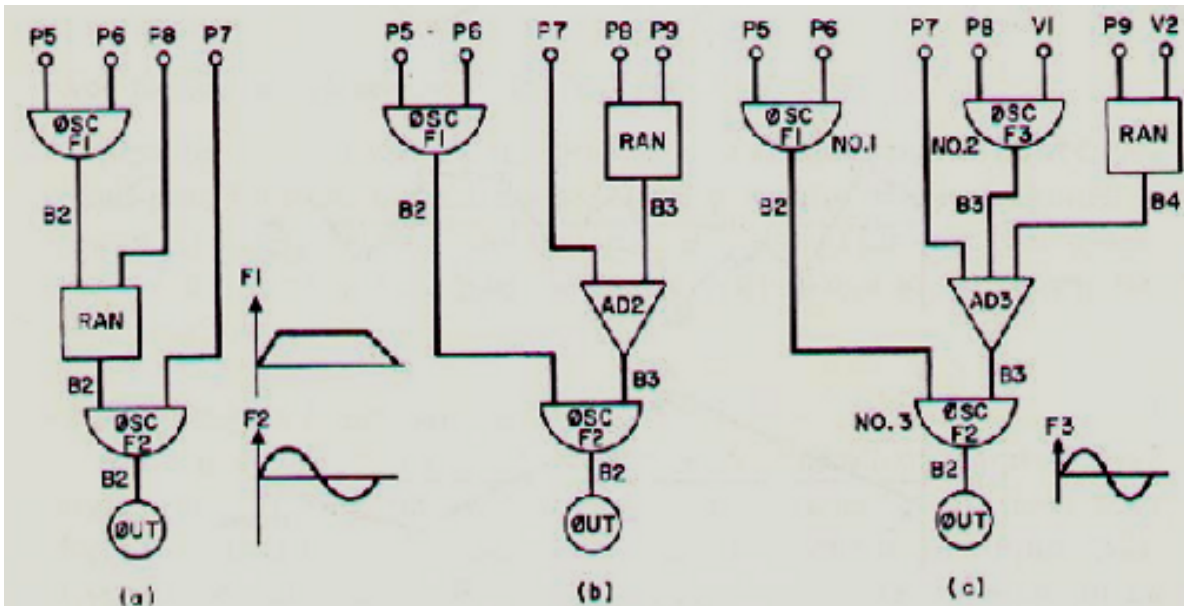
Responsáveis: *Hélio Santana - 140142959*

Rodrigo Guimarães - 140170740

Sumário

- Problema;
- Diagrama das Classes;
- Características Comuns dos Instrumentos;
- Arquitetura do Instrumento 1;
- Arquitetura do Instrumento 2;
- Arquitetura do Instrumento 3;
- Interface Gráfica;
- Melodias Escolhidas;
- Executando.

Problema



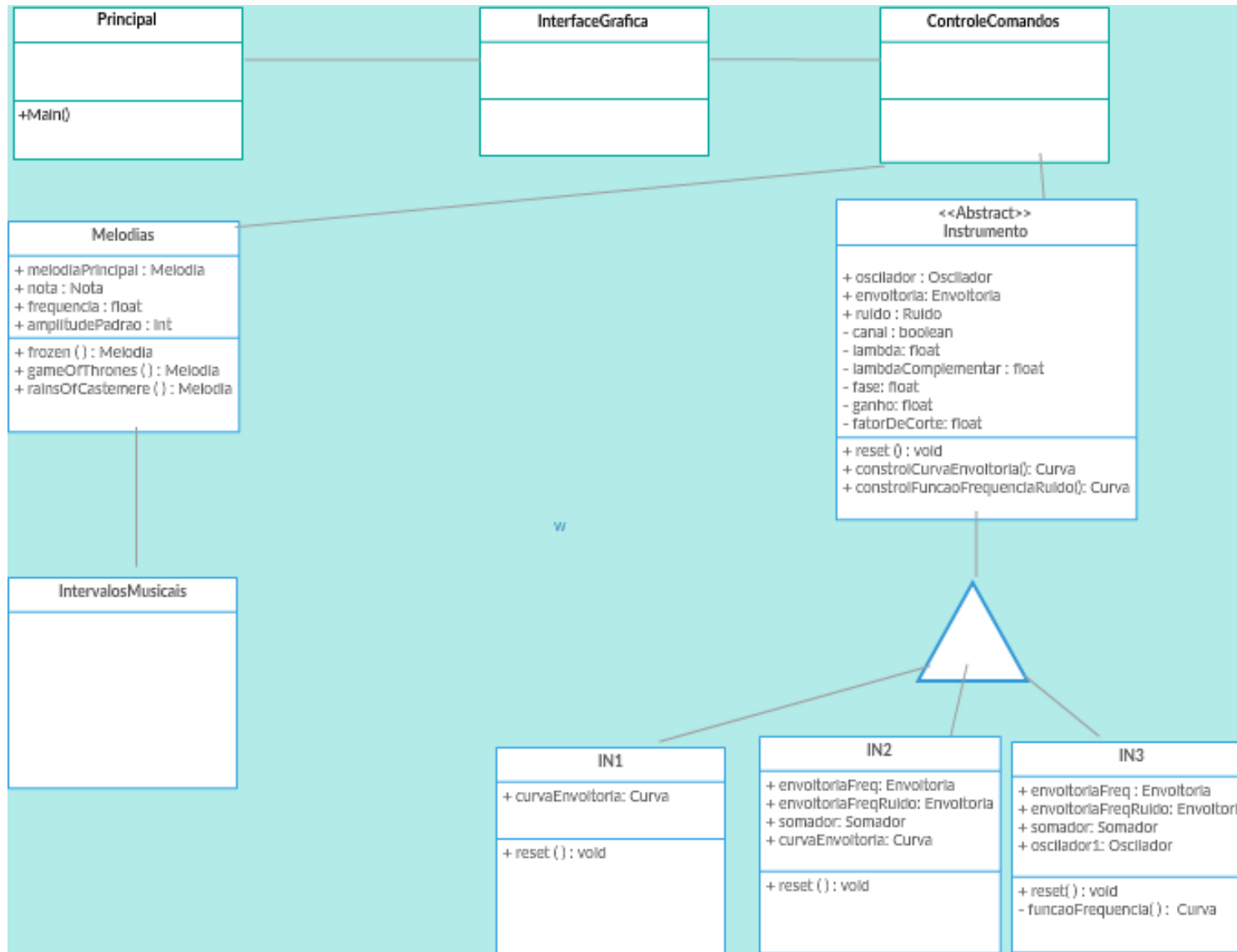
(a) INS 0 1 ;
 ØSC P5 P6 B2
 F1 P30 ;
 RAN B2 P8 B2
 P29 P27 P26 ;
 ØSC B2 P7 B2
 F2 P28 ;
 ØUT B2 B1 ;
 END ;

(b) INS 0 2 ;
 ØSC P5 P6 B2
 F1 P30 ;
 RAN P8 P9 B3
 P29 P27 P26 ;
 AD2 P7 B3 B3 ;
 ØSC B2 B3 B2
 F2 P28 ;
 ØUT B2 B1 ;
 END ;

(c) INS 0 3 ;
 ØSC P5 P6 B2
 F1 P30 ;
 ØSC P8 V1 B3
 F3 P29 ;
 RAN P9 V2 B4
 P28 P26 P25 ;
 AD3 P7 B3 B4 B3 ;
 ØSC B2 B3 B2
 F2 P27 ;
 ØUT B2 B1 ;
 END ;

- Dados os três instrumentos (a), (b) e (c) da figura ao lado, escreva um programa que implemente os três instrumentos

Diagrama de Classes Simplificado



Características Comuns dos Instrumentos

- Criada a Classe Instrumento, de modo abstrato. Por conter elementos comuns aos outros instrumentos.
- A F1 comum aos 3 instrumentos.

Arquitetura do Instrumento 1

```
@Override
public void reset(){

    curvaDeEnvoltoria = constroiCurvaDeEnvoltoria();
    this.envoltoria.setCURVA(curvaDeEnvoltoria);
    this.envoltoria.setDuracao (this.duracao);
    this.envoltoria.reset();

    this.ruido.setDispositivoAmplitude (this.envoltoria);
    this.ruido.setFrequencia ((float) this.frequencia * this.fatorCorte);
    this.ruido.setFase (this.fase);
    this.ruido.setDuracao (this.duracao);
    this.ruido.reset();

    this.oscilador.setDispositivoAmplitude (this.ruido);
    this.oscilador.setFrequencia ((float) this.frequencia);
    this.oscilador.setFase (this.fase);
    this.oscilador.setDuracao (this.duracao);
    this.oscilador.reset();
}
```

Arquitetura do Instrumento 2

```
public void reset() {
    curvaDeEnvoltoria = constroiCurvaDeEnvoltoria();
    this.envoltoria.setCURVA(curvaDeEnvoltoria);
    this.envoltoria.setDuracao(this.duracao);
    this.envoltoria.reset();

    envoltoriaFreq = new Envoltoria();
    this.envoltoriaFreq.setCURVA(constroiFuncaoFrequenciaRuido());
    this.envoltoriaFreq.setDuracao(this.duracao);
    this.envoltoriaFreq.reset();

    envoltoriaFreqRuido = new Envoltoria();
    this.envoltoriaFreqRuido.setCURVA(constroiFuncaoFrequenciaRuido());
    this.envoltoriaFreqRuido.setDuracao(this.duracao);
    this.envoltoriaFreqRuido.reset();

    this.ruido = new Ruido(envoltoriaFreqRuido);
    this.ruido.setFrequencia ((float) this.frequencia * this.fatorCorte);
    this.ruido.setFase (this.fase);
    this.ruido.setDuracao (this.duracao);
    this.ruido.reset();

    this.somador = new Somador(this.ruido, this.envoltoriaFreq );
    this.somador.setDuracao(this.duracao);
    this.somador.reset();

    this.oscilador = new Oscilador(this.envoltoria, this.somador);
    this.oscilador.setFase(this.fase);
    this.oscilador.setDuracao(this.duracao);
    this.oscilador.reset();
}
```

Arquitetura do Instrumento 3

```
public void reset(){
    this.envoltoria.setCURVA(constroiCurvaDeEnvoltoria());
    this.envoltoria.setDuracao(this.duracao);
    this.envoltoria.reset();
    envoltoriaFreq = new Envoltoria();
    this.envoltoriaFreq.setCURVA(funcaoFrequencia());
    this.envoltoriaFreq.setDuracao(this.duracao);
    this.envoltoriaFreq.reset();
    envoltoriaFreqRuido = new Envoltoria();
    this.envoltoriaFreqRuido.setCURVA(constroiFuncaoFrequenciaRuido());
    this.envoltoriaFreqRuido.setDuracao(this.duracao);
    this.envoltoriaFreqRuido.reset();
    this.ruido = new Ruido(envoltoriaFreqRuido);
    this.ruido.setFrequencia ((float) this.frequencia * this.fatorCorte);
    this.ruido.setFase (this.fase);
    this.ruido.setDuracao (this.duracao);
    this.ruido.reset();
    this.somador = new Somador(this.ruido, this.envoltoriaFreq );
    this.somador.setDuracao(this.duracao);
    this.somador.reset();
    this.oscilador1= new Oscilador(0.5f, 270, 90);
    this.oscilador1.setFase(this.fase);
    this.oscilador1.setDuracao(this.duracao);
    this.oscilador1.reset();
    this.somador = new Somador(this.somador, this.oscilador1 );
    this.somador.setDuracao(this.duracao);
    this.somador.reset();
    this.oscilador = new Oscilador(this.envoltoria, this.somador);
    this.oscilador.setFase(this.fase);
    this.oscilador.setDuracao(this.duracao);
    this.oscilador.reset();
}
```


Interface Gráfica

Abrir:

Melodias:

Instrumentos:

Som:

Salvar: ☐

Salvar

Tocar

Visualizar

Configurar:

Melodias:

Andamento:

1.0

0.01.252.53.755.0

Transposição:

1.0

0.00.51.01.52.0

Inverter: ☐

Instrumentos:

Fator de Corte:

1.0

0.00.250.50.751.0

Fase:

0.0

090180270360

Esteriofonia:

0.5

0.00.250.50.751.0

Ganho:

1.0

Melodias Escolhidas

- **Let it Go** - Kristen Anderson-Lopez e Robert Lopez
- **Game of Thrones Opening** - Ramin Djawadi
- **The Rains of Castamere** - Ramin Djawadi

Executando

- O programa construído terá suas funcionalidades demonstradas agora, comentários são bem-vindos.

Repositório

- Todo o trabalho está disponível online:
- https://github.com/helio-luk/Trabalho2_ICS/

FIM