



UNIVERSIDADE
ESTADUAL DE LONDRINA

Departamento de Computação
Relatório de Estágio Curricular

RENAN DUPAS DE MATOS

**IMPLEMENTAÇÃO DE UM SOFTWARE PARA GERAÇÃO
DE HORÁRIO ACADÊMICO COM A UTILIZAÇÃO DE
ALGORITMO GENÉTICO**

Londrina
2007

RENAN DUPAS DE MATOS

**IMPLEMENTAÇÃO DE UM SOFTWARE PARA GERAÇÃO
DE HORÁRIO ACADÊMICO COM A UTILIZAÇÃO DE
ALGORITMO GENÉTICO**

Estágio obrigatório desenvolvido durante o 4º ano do Curso de Graduação em Ciência da Computação como requisito parcial à obtenção do título de Bacharel.

Orientador: Pedro Paulo da Silva Ayrosa

2007

RENAN DUPAS DE MATOS

**IMPLEMENTAÇÃO DE UM SOFTWARE PARA GERAÇÃO
DE HORÁRIO ACADÊMICO COM A UTILIZAÇÃO DE
ALGORITMO GENÉTICO**

COMISSÃO EXAMINADORA

Prof. Dr. Pedro Paulo da Silva Ayrosa
Universidade Estadual de Londrina

Prof. Dra. Maria Angélica de O. C.
Brunetto
Universidade Estadual de Londrina

Prof. Ms. Rafael Robson Negrão
Universidade Estadual de Londrina

Londrina, 13 de novembro de 2007

*A Deus, à minha família
e à minha eterna namorada...*

AGRADECIMENTOS

À minha namorada Tatiana pelo apoio em todo tempo, para que esse trabalho fosse concluído.

À minha família pelo apoio e suporte.

Aos meus amigos, pela amizade nesses anos de curso.

Ao orientador Prof. Dr. Pedro Paulo, pelo auxílio nesse trabalho.

RESUMO

Este relatório de estágio curricular descreve o desenvolvimento de um software de construção de grades de horário, mais especificamente para o Curso de Ciência da Computação – UEL. O processo de construção de horário atualmente é realizado manualmente e despende de muito tempo para se obter um horário aceitável. O emprego de Algoritmos Genéticos (AG) é de forma a se obter uma melhor solução para a elaboração dessa grade assim como a utilização de outros softwares com o mesmo fim, como base para essa implementação. Este trabalho mostra, através de sua construção, que os resultados que obtivemos com a utilização de AG é satisfatório.

Palavras-chave: algoritmos genéticos, problema de geração de horário, inteligência artificial.

ABSTRACT

This curricular stage report describes the development of a software to build schedules, specifically for the course of Computer Science - UEL. The process of building schedule now, is done manually, and wastes a lot of time to get an acceptable schedule. The use of Genetic Algorithms (GA) a way of getting a better solution to the development of timetable and the use of other software with the same purpose as a basis for this implementation. This work shows, through its construction, the results we have obtained with the use of AG is satisfactory.

Keywords: genetic algorithms, timetabling problem, artificial intelligence

LISTA DE FIGURAS

Figura 1: Algoritmo Genético	2
Figura 2: Modelo entidade-relacionamento do banco de dados	6
Figura 3: Interface principal do sistema	7
Figura 4: Cadastro de professores	8
Figura 5: Cadastro de matérias	8
Figura 6: Cadastro de disciplinas	9
Figura 7: Exclusão de disciplinas	9
Figura 8: Exclusão de matérias	10
Figura 9: Exclusão de professores	10
Figura 10: Editar Disciplinas	11
Figura 11: Editar Matérias	11
Figura 12: Editar Professores	12
Figura 13: Edição de preferência	13
Figura 14: Preferência do professor	13
Figura 15: Grade de horários	14
Figura 16: Classe CadastroDisciplina	18
Figura 17: Classe CadastroMatéria	18
Figura 18: Classe CadastroProfessor	18
Figura 19: Classe EditarDisciplina	19
Figura 20 Classe TelaPrincipal	19
Figura 21: Classe EditarMateria	20

LISTA DE TABELAS

Tabela 1: Grade Horária - 1º Ano	14
Tabela 2: Grade Horária - 2º Ano	15
Tabela 3: Grade Horária - 3º Ano	15
Tabela 4: Grade Horária - 4º Ano	15
Tabela 5: Horário Professor ALAN	15
Tabela 6: Horário Professora CYNTHIAN	15
Tabela 7: Horário Professor DIRCEU.....	16
Tabela 8: Horário Professora ELIANA	16
Tabela 9: Horário Professor ELIANDRO	16
Tabela 10: Horário Professor ELIESER.....	16
Tabela 11: Horário Professora JANDIRA	16
Tabela 12: Horário Professora KATIA.....	16
Tabela 13: Horário Professor LUCIO	16
Tabela 14: Horário Professora LYNEAR	16
Tabela 15: Horário Professora MARIA ANGELICA	16
Tabela 16: Horário Professora MARIA TEREZA.....	16
Tabela 17: Horário Professor MARIO.....	16
Tabela 18: Horário Professora NEYVA.....	16
Tabela 19: Horário Professor PEDRO PAULO.....	17
Tabela 20: Horário Professor RODOLFO.....	17
Tabela 21: Horário Professor SAKURAY.....	17
Tabela 22: Horário Professora SAMNANTHA	17
Tabela 23: Horário Professor SERGIO	17
Tabela 24: Horário Professor STURION.....	17
Tabela 25: Horário Professor TEMPORARIO.....	17
Tabela 26: Horário Professor VITOR	17

SUMÁRIO

1	INTRODUÇÃO	1
1.1	CONSIDERAÇÕES INICIAIS	1
1.2	ORGANIZAÇÃO DO TRABALHO	1
2	DESENVOLVIMENTO.....	2
2.1	FUNCIONAMENTO DO SISTEMA.....	2
2.1.1	<i>Dados de Entrada.....</i>	<i>4</i>
2.1.2	<i>Processamento do Algoritmo Genético</i>	<i>4</i>
2.1.3	<i>Dados de Saída.....</i>	<i>5</i>
2.2	FERRAMENTAS UTILIZADAS	5
2.3	MODELO DO BANCO DE DADOS	6
3	SISTEMA	7
3.1	INTERFACES DO SISTEMA	7
3.1.1	<i>Tela Principal.....</i>	<i>7</i>
3.1.2	<i>Cadastro de Professores.....</i>	<i>8</i>
3.1.3	<i>Cadastro de Matérias</i>	<i>8</i>
3.1.4	<i>Cadastro de Disciplinas</i>	<i>9</i>
3.1.5	<i>Exclusão de Disciplinas</i>	<i>9</i>
3.1.6	<i>Exclusão de Matérias</i>	<i>10</i>
3.1.7	<i>Exclusão de Professores.....</i>	<i>10</i>
3.1.8	<i>Edição de Disciplinas.....</i>	<i>11</i>
3.1.9	<i>Edição de Matérias.....</i>	<i>11</i>
3.1.10	<i>Edição de Professores</i>	<i>12</i>
3.1.11	<i>Edição de Preferências.....</i>	<i>12</i>
3.1.12	<i>Preferências dos Professores</i>	<i>13</i>
3.1.13	<i>Grade de Horários</i>	<i>13</i>
3.2	MODELO DAS CLASSES	17
4	DIFICULDADES ENCONTRADAS.....	24
5	RESULTADOS OBTIDOS.....	25
6	RESULTADOS NÃO OBTIDOS.....	26
7	APLICAÇÕES FUTURAS	27
8	CONCLUSÃO.....	28
	REFERÊNCIAS	29

1 INTRODUÇÃO

1.1 CONSIDERAÇÕES INICIAIS

Instituições de Ensino Superior, a cada início de semestre ou ano, precisam montar suas grades horárias. Essa é uma tarefa não trivial, pois devemos construí-las analisando-as, como argumenta Júnior (2000, p. 1), “como relações entre conjuntos de professores, disciplinas, alunos, períodos de tempo e recursos. Estas relações estão sujeitas ainda a uma série de restrições, cujas combinações aumentam a complexidade do problema”.

Devido a essa dificuldade de se construir manualmente uma grade horária, que pode demandar até vários dias de trabalho (CISCON, 2006, p.3), propõe-se a implementação de um software que utiliza uma técnica heurística que utiliza conceitos das Teorias da Evolução e Genética, denominada Algoritmo Genético, o qual tem sido utilizado com êxito para construção de grades horárias.

1.2 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado da seguinte forma:

O Capítulo 2 apresenta o desenvolvimento do software, exemplificando seu funcionamento, mostrando o diagrama de classes do mesmo e o modelo do banco de dados utilizado, assim como a apresentação de todas as interfaces do sistema desenvolvido.

A conclusão desse trabalho é mostrada no Capítulo 3.

2 DESENVOLVIMENTO

2.1 FUNCIONAMENTO DO SISTEMA

O software proposto implementa um algoritmo genético para obter uma grade horária. Veja a figura 1 abaixo, que exemplifica o funcionamento de um algoritmo genético:

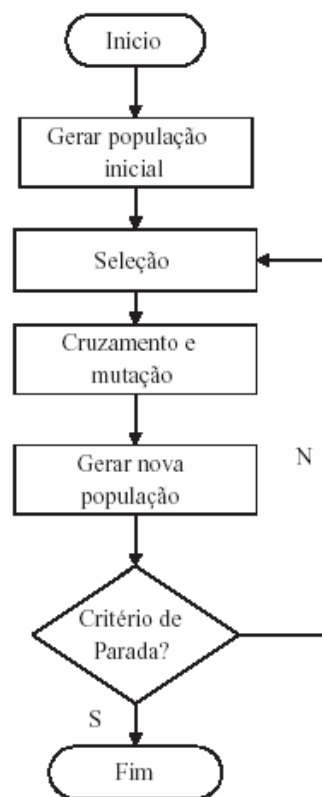


Figura 1: Algoritmo Genético

O algoritmo genético utiliza uma população (várias soluções geradas aleatoriamente) para a busca da melhor solução para o problema. Na figura acima podemos ver que essa população sofre uma seleção, ou seja, de acordo com algum critério estabelecido previamente, algumas possíveis soluções, as quais são denominadas indivíduos, são separadas, para que possam ser avaliadas posteriormente. Após essa seleção, esses indivíduos selecionados sofrem o que chamados de cruzamento, que é a troca de partes entre indivíduos, para uma diversificação das possíveis

soluções encontradas. Após isso, os mesmos sofrem uma mutação, que também é uma troca, porém entre partes do mesmo indivíduo. Depois de se passar por essas etapas, tem-se agora uma nova população de indivíduos diferente da inicial (mais diversificada).

Agora, com esses novos indivíduos, faz-se uma avaliação, baseada em algumas restrições, em cada um deles. Vejamos as restrições consideradas para avaliação dos indivíduos:

- Bloco de disciplinas: verifica a ocorrência de mais de duas aulas da mesma disciplina em sequência;
- 4º Ano - aulas à tarde: verifica a existência de aulas à tarde para a turma do 4º ano;
- Aulas na quarta-feira à tarde: verifica a existência de aulas na quarta-feira à tarde em todas as turmas;
- Horários de supervisão de TCC e Estágio: os horários de supervisão de TCC e de Estágio devem ser aulas seguidas;
- Horários de TCC e Estágio – aulas dos professores: os horários de supervisão de TCC e Estágio não podem coincidir com os horários de aula dos professores supervisores;
- Salas: duas turmas ou mais não podem ocupar a mesma sala durante o mesmo horário;
- Horários de preferência dos professores: verifica os horários de preferência de aulas dos professores.

Com essas restrições consideradas, avalia-se agora cada indivíduo, ou seja, calcula-se o *fitness de cada um, que é o valor do indivíduo aplicado a alguma função, ou nas restrições consideradas*. Agora, após a avaliação dos indivíduos, o laço do algoritmo é executado novamente. O critério de parada utilizado foi a execução do algoritmo 100 vezes. Esse foi o critério utilizado, pois se aumentarmos muito o número de gerações, ou o *loop* do algoritmo, o algoritmo demorará a retornar uma solução.

Ao terminar a execução, o algoritmo retorna o melhor horário obtido. Com isso, temos a solução para o problema proposto.

2.1.1 Dados de Entrada

O sistema possui uma interface gráfica para facilitar a inclusão dos dados de entrada. Como dados de entrada, temos:

- Professores;
- Matérias;
- Disciplinas;
- Horários preferenciais dos professores.

2.1.2 Processamento do Algoritmo Genético

Após a entrada dos dados, o algoritmo genético é executado. A execução do algoritmo compreende o fluxo mostrado na figura 2. Após sua execução, tem-se uma grade de horários final.

2.1.3 Dados de Saída

O horário final obtido pela execução do algoritmo genético é mostrado em uma tabela (figura 15), que pode ser salva em arquivo no formato HTML¹. Também se pode salvar essa grade de horários no mesmo formato HTML, porém os horários obtidos serão mostrados sob a visão dos professores.

2.2 FERRAMENTAS UTILIZADAS

Para o desenvolvimento dessa aplicação, foi utilizada a linguagem Java. Essa linguagem foi adotada, pois é uma linguagem totalmente orientada a objetos, e isso nos permite uma reutilização posterior de código. Além disso, como é uma linguagem interpretada, isso permite que se possa executar um programa em Java em qualquer plataforma que possua um interpretador Java.

Para a implementação desse software utilizamos de início o ambiente de programação Eclipse 3.2.2, porém, a partir do início da fase de construção da interface do software, foi utilizado outro ambiente, o NetBeans 5.5, pois o mesmo se mostrou bem mais prático para esse fim.

O banco de dados utilizado o PostgreSQL 8.2 em conjunto com o PgAdmin III para interface direta com o banco de dados, e o SQL Manager 2005 for PostgreSQL, utilizado para o mesmo fim. Também foi utilizado um plugin postgresql-8.2-505.jdbc4.jar para conexão do programa com banco de dados.

¹ *HyperText Markup Language* - Linguagem de Marcação de Hipertexto

2.3 MODELO DO BANCO DE DADOS

Conforme as necessidades do sistema foram surgindo, o banco de dados também foi sendo criado e modificado. A seguir é apresentado o modelo do banco de dados final, após todas as correções necessárias.

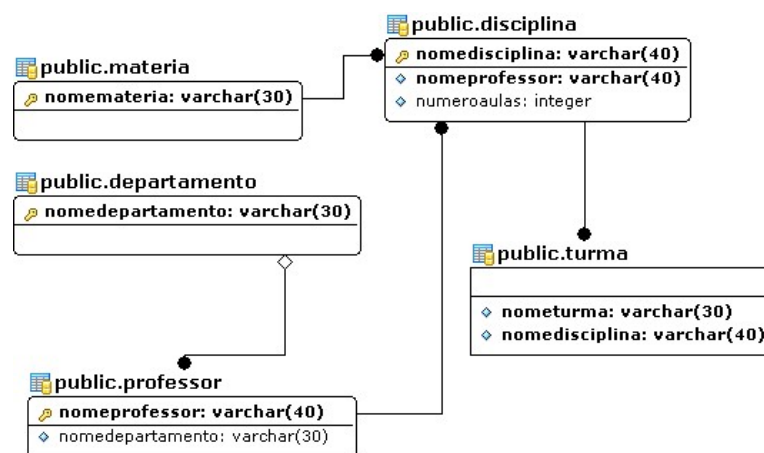


Figura 2: Modelo entidade-relacionamento do banco de dados

3 SISTEMA

3.1 INTERFACES DO SISTEMA

Baseando-se nos requisitos necessários para a criação desse sistema (cadastro, edição, exclusão, etc.) as interfaces do sistema foram desenvolvidas. Diversas modificações foram feitas nas mesmas até a conclusão desse trabalho. As versões finais dessas interfaces são mostradas nas seções a seguir.

3.1.1 Tela Principal

Essa interface é responsável por fornecer um conjunto de opções de utilização do sistema para o usuário. Cada opção mostrada (Arquivo, Editar, Grade de Horário) possui um menu, que é mostrado quando o usuário clica sobre ela.



Figura 3: Interface principal do sistema

3.1.2 Cadastro de Professores

Essa interface é responsável pelo cadastro de professores, e é disparada pelo item 'Cadastro de Professores' da opção 'Arquivo'.

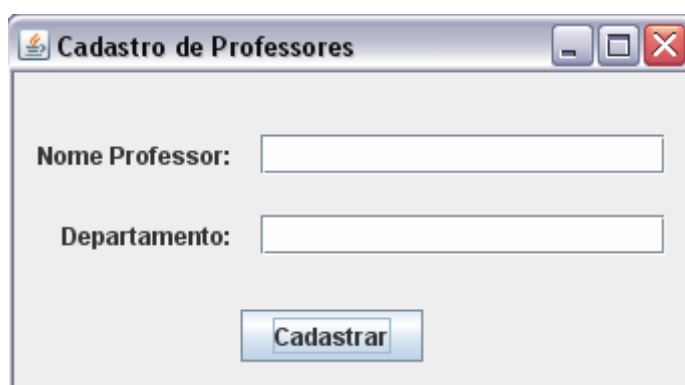
A screenshot of a Windows-style application window titled "Cadastro de Professores". The window has a standard title bar with minimize, maximize, and close buttons. The main content area is light gray and contains two text input fields. The first field is labeled "Nome Professor:" and the second is labeled "Departamento:". Below these fields is a blue button with the text "Cadastrar".

Figura 4: Cadastro de professores

3.1.3 Cadastro de Matérias

Essa interface é responsável pelo cadastro de matérias, e é disparada pelo item 'Cadastro de Matérias' da opção 'Arquivo'.

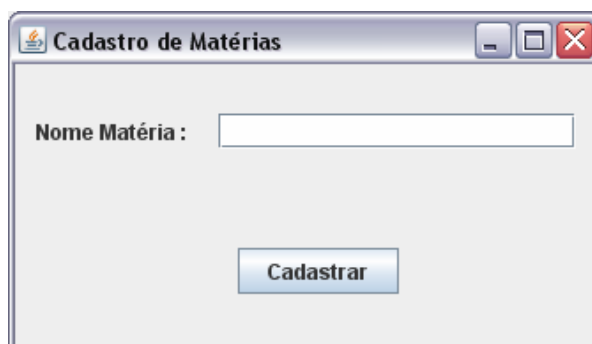
A screenshot of a Windows-style application window titled "Cadastro de Matérias". The window has a standard title bar with minimize, maximize, and close buttons. The main content area is light gray and contains a single text input field labeled "Nome Matéria :". Below this field is a blue button with the text "Cadastrar".

Figura 5: Cadastro de matérias

3.1.4 Cadastro de Disciplinas

Essa interface é responsável pelo cadastro de disciplinas, e é disparada pelo item 'Cadastro de Disciplinas' da opção 'Arquivo'.

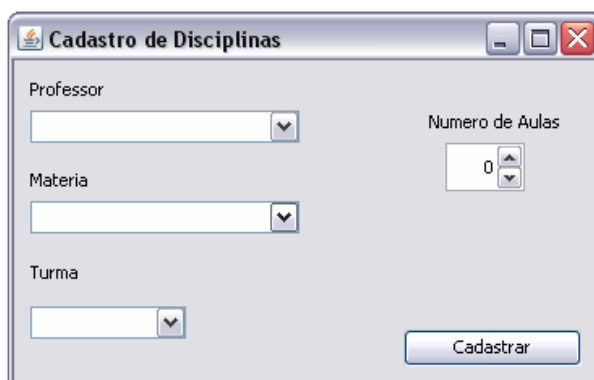
A janela 'Cadastro de Disciplinas' possui um título com ícone de fogo. Contém três campos de seleção rotulados 'Professor', 'Materia' e 'Turma'. À direita, há um campo rotulado 'Numero de Aulas' com o valor '0' e setas de incremento/decremento. Um botão 'Cadastrar' está na parte inferior direita.

Figura 6: Cadastro de disciplinas

3.1.5 Exclusão de Disciplinas

Essa interface é responsável pela exclusão de disciplinas, sendo disparada pelo item 'Disciplina' da opção 'Excluir'.

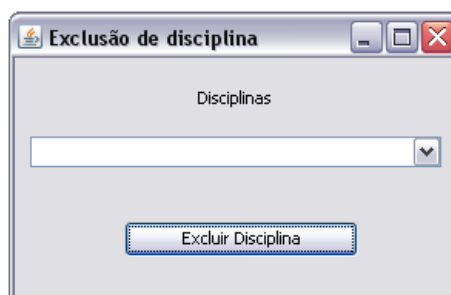
A janela 'Exclusão de disciplina' possui um título com ícone de fogo. Contém um campo de seleção rotulado 'Disciplinas'. Abaixo dele, há um botão 'Excluir Disciplina'.

Figura 7: Exclusão de disciplinas

3.1.6 Exclusão de Matérias

Essa interface é responsável pela exclusão de matérias, sendo disparada pelo item 'Matéria' da opção 'Excluir'.

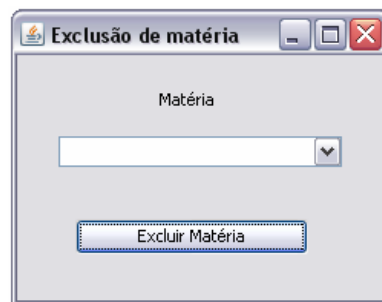


Figura 8: Exclusão de matérias

3.1.7 Exclusão de Professores

Essa interface é responsável pela exclusão de professores, sendo disparada pelo item 'Professor' da opção 'Editar'.

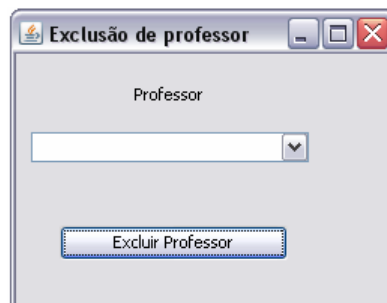


Figura 9: Exclusão de professores

3.1.8 Edição de Disciplinas

Essa interface é responsável pela edição de disciplinas, e é disparada pelo item 'Disciplina' da opção 'Editar'.

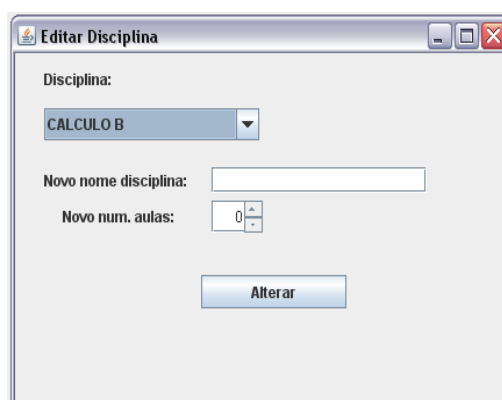
A janela 'Editar Disciplina' possui um título com ícone e botões de controle. O formulário contém: 'Disciplina:' com um menu suspenso selecionando 'CALCULO B'; 'Novo nome disciplina:' com um campo de texto vazio; 'Novo num. aulas:' com um campo de entrada numérica contendo '0' e botões de incremento/decremento; e um botão 'Alterar' na base.

Figura 10: Editar Disciplinas

3.1.9 Edição de Matérias

Essa interface é responsável pela edição de matérias, sendo disparada pelo item 'Matéria' da opção 'Editar'.

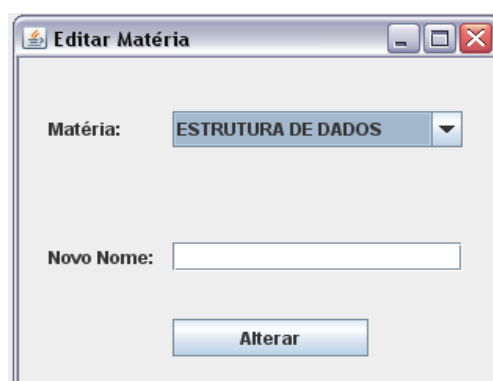
A janela 'Editar Matéria' possui um título com ícone e botões de controle. O formulário contém: 'Matéria:' com um menu suspenso selecionando 'ESTRUTURA DE DADOS'; 'Novo Nome:' com um campo de texto vazio; e um botão 'Alterar' na base.

Figura 11: Editar Matérias

3.1.10 Edição de Professores

Essa interface é responsável pela edição de professores, sendo disparada pelo item 'Professor' da opção 'Editar'.

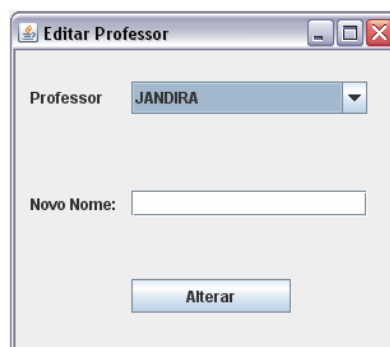


Figura 12: Editar Professores

3.1.11 Edição de Preferências

Essa interface é responsável pela escolha do professor para edição de preferência de aulas do mesmo, sendo disparada pelo item 'Preferência Professor' da opção 'Editar'.

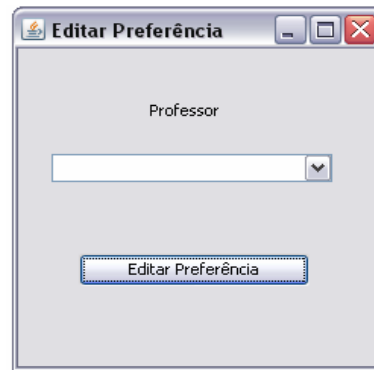


Figura 13: Edição de preferência

3.1.12 Preferências dos Professores

Essa interface é responsável pela definição dos horários preferenciais de cada professor. É disparada pelo botão 'Editar Preferência' da Interface 'Edição de Preferência'.

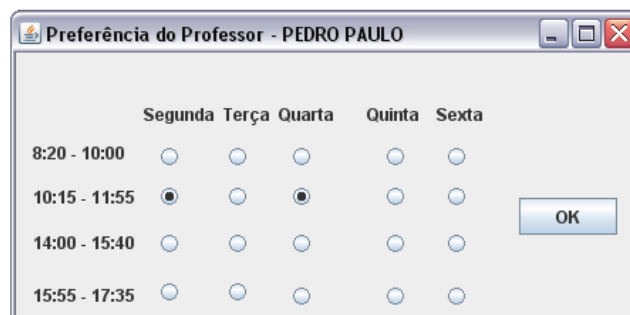


Figura 14: Preferência do professor

3.1.13 Grade de Horários

Essa interface é responsável pela execução do Algoritmo genético que retorna um horário para as quatro turmas, o qual é mostrado nessa Tabela da figura 15, e é disparada pelo item ‘Gerar Grade Horária’ da opção ‘Grade de Horários’. Somente após a geração de um horário, as opções ‘Salvar Grade de Horários’, ‘Salvar Horário dos Professores’ e ‘Gerar outra Grade de Horários’ são habilitadas para o usuário, o qual pode, através da primeira opção salvar essa tabela de horários em um arquivo HTML, através da segunda, salvar essa tabela no formato que mostre os horários de cada professor e através da terceira opção gerar outro horário.

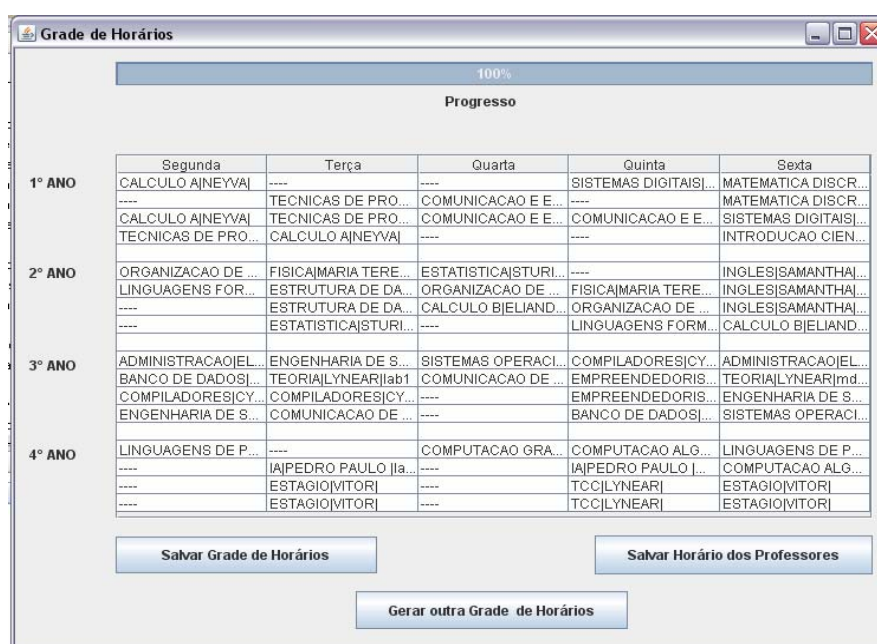


Figura 15: Grade de horários

Vejamos uma exemplificação de uma grade horária salva através da opção ‘Salvar Grade de Horários’:

Segunda	Terça	Quarta	Quinta	Sexta
COMUNICACAO E EXPRESSAO TEMPORARIO 853	----	----	----	----
COMUNICACAO E EXPRESSAO TEMPORARIO 853	CALCULO A NEYVA lab1	TECNICAS DE PROGRAMACAO JANDIRA mdc1	TECNICAS DE PROGRAMACAO JANDIRA lab2	SISTEMAS DIGITAIS LUCIO 155a
CALCULO A NEYVA	----	CALCULO A NEYVA	MATEMATICA DISCRETA SERGIO 853	TECNICAS DE PROGRAMACAO JANDIRA mdc1
SISTEMAS DIGITAIS LUCIO 155a	MATEMATICA DISCRETA SERGIO	COMUNICACAO E EXPRESSAO TEMPORARIO 853	INTRODUCAO CIENCIA COMPUTACAO DIRCEU mdc1	----

Tabela 1: Grade Horária - 1º Ano

Segunda	Terça	Quarta	Quinta	Sexta
FISICA MARIA TEREZA 484	ORGANIZACAO DE COMPUTADORES ELIESER mdc1	LINGUAGENS FORMAIS CYNTHIAN mdc1	INGLES SAMANTHA 123	CALCULO B ELIANDRO
ESTATISTICA STURION 123	----	ESTATISTICA STURION 484	ORGANIZACAO DE COMPUTADORES ELIESER mdc1	----
LINGUAGENS FORMAIS CYNTHIAN mdc1	INGLES SAMANTHA 123	INGLES SAMANTHA 123	ORGANIZACAO DE COMPUTADORES ELIESER mdc1	FISICA MARIA TEREZA 484
CALCULO B ELIANDRO lab1	ESTRUTURA DE DADOS RODOLFO mdc1	----	ESTRUTURA DE DADOS RODOLFO mdc2	----

Tabela 2: Grade Horária - 2º Ano

Segunda	Terça	Quarta	Quinta	Sexta
ENGENHARIA DE SOFTWARE RODOLFO mdc2	SISTEMAS OPERACIONAIS SAKURAY lab2	EMPREENDEDORISMO JANDIRA lab1	ENGENHARIA DE SOFTWARE RODOLFO mdc2	SISTEMAS OPERACIONAIS SAKURAY lab1
BANCO DE DADOS VITOR lab2	BANCO DE DADOS VITOR mdc1	TEORIA LYNEAR lab1	COMUNICACAO DE DADOS MARIO mdc2	TEORIA LYNEAR lab2
COMUNICACAO DE DADOS MARIO mdc2	ENGENHARIA DE SOFTWARE RODOLFO lab2	----	COMPILADORES CYNTHIAN mdc2	COMPILADORES CYNTHIAN mdc2
ADMINISTRACAO ELIANA 466	ADMINISTRACAO ELIANA 466	----	COMPILADORES CYNTHIAN lab2	EMPREENDEDORISMO JANDIRA mdc2

Tabela 3: Grade Horária - 3º Ano

Segunda	Terça	Quarta	Quinta	Sexta
----	COMPUTACAO ALGEBRICA MARIA ANGELICA mdc2	----	IA PEDRO PAULO mdc1	COMPUTACAO GRAFICA ALAN lab2
COMPUTACAO ALGEBRICA MARIA ANGELICA mdc1	LINGUAGENS DE PROGRAMA-CAO KATIA mdc2	----	LINGUAGENS DE PROGRAMACAO KATIA lab1	IA PEDRO PAULO lab1
ESTAGIO VITOR	ESTAGIO VITOR	----	----	TCC LYNEAR
ESTAGIO VITOR	ESTAGIO VITOR	----	----	TCC LYNEAR

Tabela 4: Grade Horária - 4º Ano

Agora, vejamos uma exemplificação de uma grade horária salva através da opção ‘Salvar Horário dos Professores’:

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	4 ANO
---	---	---	---	---
---	---	---	---	---
---	---	---	---	---

Tabela 5: Horário Professor ALAN

Segunda	Terça	Quarta	Quinta	Sexta
---	---	2 ANO	---	---
---	---	---	---	---
2 ANO	---	---	3 ANO	3 ANO
---	---	---	3 ANO	---

Tabela 6: Horário Professora CYNTHIAN

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
---	---	---	---	---
---	---	---	---	---
---	---	---	1 ANO	---

Tabela 7: Horário Professor DIRCEU

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
---	---	---	---	---
---	---	---	---	---
3 ANO	3 ANO	---	---	---

Tabela 8: Horário Professora ELIANA

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	2 ANO
---	---	---	---	---
---	---	---	---	---
2 ANO	---	---	---	---

Tabela 9: Horário Professor ELIANDRO

Segunda	Terça	Quarta	Quinta	Sexta
---	2 ANO	---	---	---
---	---	---	2 ANO	---
---	---	---	2 ANO	---
---	---	---	---	---

Tabela 10: Horário Professor ELIESER

Segunda	Terça	Quarta	Quinta	Sexta
---	---	3 ANO	---	---
---	---	1 ANO	1 ANO	---
---	---	---	---	1 ANO
---	---	---	---	3 ANO

Tabela 11: Horário Professora JANDIRA

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
---	4 ANO	---	4 ANO	---
---	---	---	---	---
---	---	---	---	---

Tabela 12: Horário Professora KATIA

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
---	---	---	---	1 ANO
---	---	---	---	---
1 ANO	---	---	---	---

Tabela 13: Horário Professor LUCIO

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
---	---	3 ANO	---	3 ANO
---	---	---	---	4 ANO
---	---	---	---	4 ANO

Tabela 14: Horário Professora LYNEAR

Segunda	Terça	Quarta	Quinta	Sexta
---	4 ANO	---	---	---
4 ANO	---	---	---	---
---	---	---	---	---
---	---	---	---	---

Tabela 15: Horário Professora MARIA ANGELICA

Segunda	Terça	Quarta	Quinta	Sexta
2 ANO	---	---	---	---
---	---	---	---	---
---	---	---	---	2 ANO
---	---	---	---	---

Tabela 16: Horário Professora MARIA TEREZA

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
---	---	---	3 ANO	---
3 ANO	---	---	---	---
---	---	---	---	---

Tabela 17: Horário Professor MARIO

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
---	1 ANO	---	---	---
1 ANO	---	1 ANO	---	---
---	---	---	---	---

Tabela 18: Horário Professora NEYVA

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	4 ANO	---
---	---	---	---	4 ANO
---	---	---	---	---
---	---	---	---	---

Tabela 19: Horário Professor PEDRO PAULO

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
---	---	---	---	---
---	---	---	1 ANO	---
---	1 ANO	---	---	---

Tabela 23: Horário Professor SERGIO

Segunda	Terça	Quarta	Quinta	Sexta
3 ANO	---	---	3 ANO	---
---	---	---	---	---
---	3 ANO	---	---	---
---	2 ANO	---	2 ANO	---

Tabela 20: Horário Professor RODOLFO

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
2 ANO	---	2 ANO	---	---
---	---	---	---	---
---	---	---	---	---

Tabela 24: Horário Professor STURION

Segunda	Terça	Quarta	Quinta	Sexta
---	3 ANO	---	---	3 ANO
---	---	---	---	---
---	---	---	---	---
---	---	---	---	---

Tabela 21: Horário Professor SAKURAY

Segunda	Terça	Quarta	Quinta	Sexta
1 ANO	---	---	---	---
1 ANO	---	---	---	---
---	---	---	---	---
---	---	1 ANO	---	---

Tabela 25: Horário Professor TEMPORARIO

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	2 ANO	---
---	---	---	---	---
---	2 ANO	2 ANO	---	---
---	---	---	---	---

Tabela 22: Horário Professora SAMNANTHA

Segunda	Terça	Quarta	Quinta	Sexta
---	---	---	---	---
3 ANO	3 ANO	---	---	---
4 ANO	4 ANO	---	---	---
4 ANO	4 ANO	---	---	---

Tabela 26: Horário Professor VITOR

3.2 MODELO DAS CLASSES

O modelo das principais classes desenvolvidas é demonstrado a seguir. Todas essas classes implementam as interfaces expostas anteriormente.

CadastroDisciplina
<i>Atributos</i> privado JButton CadastroDisciplinaOK privado JLabel jLabel1 privado JComboBox NomesMaterias privado JComboBox NomesProfessores privado JComboBox NomesTurmas privado JSpinner NumeroAulas privado JLabel jLabel2 privado JLabel jLabel3 privado JLabel jLabel4
<i>Operações</i> publico CadastroDisciplina() privado void initComponents() privado void formWindowOpened(WindowEvent evt) privado void CadastroDisciplinaOKMouseClicked(MouseEvent evt) publico void main(String args[0..*])

Figura 16: Classe CadastroDisciplina

CadastroMateria
<i>Atributos</i> privado JButton CadastroMateriaOK privado JTextField NomeMateria privado JLabel jLabel1
<i>Operações</i> público CadastroMateria() privado void initComponents() privado void CadastroMateriaOKMouseClicked(MouseEvent evt) público void main(String args[0..*])

Figura 17: Classe CadastroMatéria

CadastroProfessor
<i>Atributos</i> privado JButton CadastroProfessorOK privado JTextField Departamento privado JTextField NomeProfessor privado JLabel jLabel1 privado JLabel jLabel2
<i>Operações</i> público CadastroProfessor() privado void initComponents() privado void CadastroProfessorOKMouseClicked(MouseEvent evt) público void main(String args[0..*])

Figura 18: Classe CadastroProfessor

EditarDisciplina
<i>Atributos</i> privado JButton AlterarDisciplina privado JLabel JLabel1 privado JComboBox NomesDisciplinas privado JTextField NovoNomeDisciplina privado JSpinner NumeroAulas privado JLabel jlLabel1 privado JLabel jlLabel2
<i>Operações</i> público EditarDisciplina() privado void initComponents() privado void formWindowOpened(WindowEvent evt) privado void AlterarDisciplinaMouseReleased(MouseEvent evt) público void main(String args[0..*])

Figura 19: Classe EditarDisciplina

TelaPrincipal
<i>Atributos</i> privado JMenuItem CadastroDisciplina privado JMenuItem CadastroMateria privado JMenuItem CadastroProfessor privado JMenuItem EditarDisciplina privado JMenuItem EditarMateria privado JMenuItem EditarProfessor privado JMenuItem ExcluirDisciplina privado JMenuItem ExcluirMateria privado JMenuItem ExcluirProfessor privado JMenuItem GerarGradeHoraria privado JMenuItem PreferenciaProfessor privado JMenuItem Sair privado JMenuItem jMenu1 privado JMenuItem jMenu2 privado JMenuItem jMenu3 privado JMenuItem jMenu4 privado JMenuItemBar jMenuBar1
<i>Operações</i> público TelaPrincipal() privado void initComponents() privado void EditarDisciplinaMouseReleased(MouseEvent evt) privado void EditarMateriaMouseReleased(MouseEvent evt) privado void EditarProfessorMouseReleased(MouseEvent evt) privado void PreferenciaProfessorActionPerformed(ActionEvent evt) privado void PreferenciaProfessorMouseReleased(MouseEvent evt) privado void EditarMateriaActionPerformed(ActionEvent evt) privado void EditarProfessorActionPerformed(ActionEvent evt) privado void GerarGradeHorariaMouseReleased(MouseEvent evt) privado void ExcluirMateriaMouseReleased(MouseEvent evt) privado void ExcluirDisciplinaMouseReleased(MouseEvent evt) privado void ExcluirProfessorActionPerformed(ActionEvent evt) privado void SairMouseReleased(MouseEvent evt) privado void ExcluirProfessorMouseReleased(MouseEvent evt) privado void CadastroProfessorMenuDragMouseExited(MenuDragMouseEvent evt) privado void CadastroDisciplinaMouseReleased(MouseEvent evt) privado void CadastroMateriaMouseReleased(MouseEvent evt) privado void CadastroProfessorMouseReleased(MouseEvent evt) público void main(String args[0..*])

Figura 20 Classe TelaPrincipal

EditarMateria
<i>Atributos</i>
privado JButton AlterarMateria
privado JLabel JLabel1
privado JComboBox NomesMaterias
privado JTextField NovoNomeMateria
privado JLabel jLabel1
<i>Operações</i>
público EditarMateria()
privado void initComponents()
privado void formMouseReleased(MouseEvent evt)
privado void formWindowOpened(WindowEvent evt)
privado void AlterarMateriaMouseReleased(MouseEvent evt)
público void main(String args[0..*])

Figura 21: Classe EditarMateria

EditarProfessor
<i>Atributos</i>
privado JButton AlterarProfessor
privado JLabel JLabel1
privado JComboBox NomesProfessores
privado JTextField NovoNomeProfessor
privado JLabel jLabel1
<i>Operações</i>
público EditarProfessor()
privado void initComponents()
privado void formWindowOpened(WindowEvent evt)
privado void AlterarProfessorMouseReleased(MouseEvent evt)
público void main(String args[0..*])

Figura 22: Classe EditarProfessor

ExcluirDisciplina
<i>Atributos</i>
privado JComboBox Disciplinas
privado JButton ExcluirDisciplina
privado JLabel JLabel1
<i>Operações</i>
público ExcluirDisciplina()
privado void initComponents()
privado void formWindowOpened(WindowEvent evt)
privado void ExcluirDisciplinaMouseReleased(MouseEvent evt)
público void main(String args[0..*])

Figura 23: Classe ExcluirDisciplina

ExcluirMateria
<i>Atributos</i>
privado JButton ExcluirMateria
privado JLabel JLabel1
privado JComboBox NomesMaterias
<i>Operações</i>
público ExcluirMateria()
privado void initComponents()
privado void formWindowOpened(WindowEvent evt)
privado void ExcluirMateriaMouseReleased(MouseEvent evt)
público void main(String args[0..*])

Figura 24: Classe ExcluirMateria

ExcluirProfessor
<i>Atributos</i>
privado JButton ExcluirProfessor
privado JLabel JLabel1
privado JComboBox NomesProfessores
<i>Operações</i>
público ExcluirProfessor()
privado void initComponents()
privado void formWindowOpened(WindowEvent evt)
privado void ExcluirProfessorMouseReleased(MouseEvent evt)
público void main(String args[0..*])

Figura 25: Classe ExcluirProfessor

EscolheProfessor
<i>Atributos</i>
privado JButton EditarPreferenciaProfessor
privado JLabel JLabel1
privado JComboBox NomesProfessores
<i>Operações</i>
público EscolheProfessor()
privado void initComponents()
privado void formWindowOpened(WindowEvent evt)
privado void EditarPreferenciaProfessorMouseReleased(MouseEvent evt)
público void main(String args[0..*])

Figura 26: Classe EscolheProfessor

TelaDeProgresso	
<i>Atributos</i>	
privado JButton	GerarOutraGrade
privado JButton	SalvarHorario
privado JButton	SalvarHorarioProfessores
privado JTable	Tabela
privado JLabel	jLabel1
privado JLabel	jLabel2
privado JLabel	jLabel3
privado JLabel	jLabel4
privado JLabel	jLabel5
privado JProgressBar	JProgressBar1
privado JScrollPane	jScrollPane1
<i>Operações</i>	
público	TelaDeProgresso()
privado void	initComponents()
privado void	SalvarHorarioProfessoresMouseReleased(MouseEvent evt)
privado void	GerarOutraGradeMouseReleased(MouseEvent evt)
privado void	SalvarHorarioMouseReleased(MouseEvent evt)
privado void	formWindowOpened(WindowEvent evt)
público void	main(String args[0..*])

Figura 27: Classe TelaDeProgresso

PreferenciaProfessores
<p><i>Atributos</i></p> <p><u>público boolean prefProfessor[0..* 0..* 0..*]</u></p> <p><u>público String Professores[0..*]</u></p> <p>privado JButton OK</p> <p>privado JRadioButton Quar1</p> <p>privado JRadioButton Quar2</p> <p>privado JRadioButton Quar3</p> <p>privado JRadioButton Quar4</p> <p>privado JRadioButton Quin1</p> <p>privado JRadioButton Quin2</p> <p>privado JRadioButton Quin3</p> <p>privado JRadioButton Quin4</p> <p>privado JRadioButton Seg1</p> <p>privado JRadioButton Seg2</p> <p>privado JRadioButton Seg3</p> <p>privado JRadioButton Seg4</p> <p>privado JRadioButton Sex1</p> <p>privado JRadioButton Sex2</p> <p>privado JRadioButton Sex3</p> <p>privado JRadioButton Sex4</p> <p>privado JRadioButton Ter1</p> <p>privado JRadioButton Ter2</p> <p>privado JRadioButton Ter3</p> <p>privado JRadioButton Ter4</p> <p>privado JLabel Label1</p> <p>privado JLabel Label2</p> <p>privado JLabel Label3</p> <p>privado JLabel Label4</p> <p>privado JLabel Label5</p> <p>privado JLabel Label6</p> <p>privado JLabel Label7</p> <p>privado JLabel Label8</p> <p>privado JLabel Label9</p>
<p><i>Operações</i></p> <p>público PreferenciaProfessores()</p> <p>privado void initComponents()</p> <p>privado void formWindowOpened(WindowEvent evt)</p> <p>privado void OKMouseReleased(MouseEvent evt)</p> <p>público void iniciaPrefProfessores()</p> <p><u>público void main(String args[0..*])</u></p> <p>público int EncontraIndiceProfessor(String Professor)</p>

Figura 28: Classe PreferenciaProfessores

4 DIFICULDADES ENCONTRADAS

Ao se iniciar este estágio, foi necessária a definição de uma estrutura de dados em que a grade horária fosse alocada. De acordo com Matos (2007, p. 12), a estrutura utilizada para manipular os dados foi uma matriz de três dimensões.

Outra dificuldade encontrada foi a integração do banco de dados utilizado com o software, pois foi necessária a utilização de um driver JDBC². O que dificultou de início foi a configuração do driver e o estabelecimento da conexão com o banco.

² *Java Database Connectivity* – Conectividade de banco de dados Java

5 RESULTADOS OBTIDOS

Ao finalizarmos o desenvolvimento dessa aplicação, obtivemos um software que constrói de forma simples uma grade de horário, sendo necessária para isso algumas entradas de dados, como professores, matérias e disciplinas. Ao executarmos o sistema, pode ver que o tempo necessário que o mesmo apresente uma solução é bem curto (em média 30 segundos).

Após a geração de uma grade horária, pode-se salvar essa grade obtida e a mesma grade (ver tabelas 1 a 4), porém na visão dos professores (ver tabelas 5 a 26), em um arquivo html.

6 RESULTADOS NÃO OBTIDOS

No início da construção do software desejava-se implementar dois operadores genéticos de cruzamento, o operador com corte em 1 ponto e o operador com corte em 2 pontos. Após ter-se implementado o primeiro operador e visto os resultados, decidiu-se não implementar o outro operador, visto que o operador em 1 ponto apresentou resultados satisfatórios.

7 APLICAÇÕES FUTURAS

No desenvolvimento desse trabalho, percebeu-se que a abordagem para o desenvolvimento dessa aplicação foi bem específica (curso de Ciência da Computação). Que quisermos utilizá-lo para a geração de horários de outros cursos, por exemplo, o sistema não apresentará bons resultados para esses cursos, pois o algoritmo aborda algumas restrições específicas do curso de Computação. Assim, o que pode ser feito é aplicar uma abordagem mais generalista para o algoritmo, conseguindo-se assim um resultado mais abrangente.

Também pode-se implementar outros operadores e analisar os resultados obtidos com isso.

8 CONCLUSÃO

Através da implementação da aplicação proposta, conseguiu-se desenvolver um sistema de geração automática de grade de horários do curso de Ciência da Computação, bastando somente definir alguns dados, como os professores, disciplinas e preferência dos professores, para obtermos uma grade de horários em um tempo breve, em comparação ao desenvolvimento de uma solução de forma manual, o qual seriam necessárias várias horas para isso.

Com relação ao algoritmo programado, vários conceitos estudados foram colocados em prática e testados, e os mesmos puderam ser comprovados como válidos. Entre esses conceitos, podemos citar o operador de seleção em um ponto e operador de mutação.

Além disso, as proposições futuras colocadas podem melhorar ainda mais a solução desse tipo de problema.

REFERÊNCIAS

CISCON, L. A. **O PROBLEMA DE GERAÇÃO DE HORÁRIOS: UM FOCO NA ELIMINAÇÃO DE JANELAS E AULAS ISOLADAS**. Monografia (Graduação em Ciência da Computação) - Universidade Federal de Lavras, Lavras, 2006. Disponível em: <http://www.comp.ufla.br/monografias/ano2005/O_problema_de_geracao_de_horarios_um_foco_na Eliminacao_de_janelas_e_aulas_isoladas.pdf>. Acesso em 11 julho 2007.

JÚNIOR, O. O. B. **OTIMIZAÇÃO DE HORÁRIOS EM INSTITUIÇÕES DE ENSINO SUPERIOR ATRAVÉS DE ALGORITMOS GENÉTICOS**. Dissertação (Mestrado em Engenharia de Produção) - Universidade Federal de Santa Catarina. Florianópolis, 2000. Disponível em: <<http://teses.eps.ufsc.br/defesa/pdf/3436.pdf>>. Acesso em: 14 julho 2007.

MATOS, R. D. **UTILIZAÇÃO DE ALGORITMO GENÉTICO PARA RESOLUÇÃO DO PROBLEMA DE GERAÇÃO DE HORÁRIOS**. Trabalho de Conclusão de Curso. Universidade Estadual de Londrina, Londrina, 2007.