



UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS EXATAS, NATURAIS E DA SAÚDE
DEPARTAMENTO DE COMPUTAÇÃO

Tulio Machado Müller

**ESTUDO DO IMPACTO DA
ALEATORIEDADE E MOVIMENTOS NA
META-HEURÍSTICA GRASP COM
SIMULATED ANNEALING PARA O
PROBLEMA DE TABELA-HORÁRIO DO
DCOMP-CCENS-UFES**

Alegre - ES

2020

Tulio Machado Müller

**ESTUDO DO IMPACTO DA ALEATORIEDADE E
MOVIMENTOS NA META-HEURÍSTICA GRASP COM
SIMULATED ANNEALING PARA O PROBLEMA DE
TABELA-HORÁRIO DO DCOMP-CCENS-UFES**

Trabalho de conclusão de curso apresentado
ao Departamento de Computação do Centro
de Ciências Exatas, Naturais e da Saúde da
Universidade Federal do Espírito Santo, como
requisito parcial para obtenção do Grau de
Bacharel em Ciência da Computação.

Universidade Federal do Espírito Santo
Centro de Ciências Exatas, Naturais e da Saúde
Departamento de Computação

Orientador: Prof. Dr. Edmar Hell Kampke

Alegre - ES

2020

Tulio Machado Müller

ESTUDO DO IMPACTO DA ALEATORIEDADE E MOVIMENTOS NA META-HEURÍSTICA GRASP COM SIMULATED ANNEALING PARA O PROBLEMA DE TABELA-HORÁRIO DO DCOMP-CCENS-UFES

Trabalho de conclusão de curso apresentado
ao Departamento de Computação do Centro
de Ciências Exatas, Naturais e da Saúde da
Universidade Federal do Espírito Santo, como
requisito parcial para obtenção do Grau de
Bacharel em Ciência da Computação.

Trabalho Aprovado. Alegre - ES, 12 de Novembro de 2020:

Prof. Dr. Edmar Hell Kampke
Orientador

Prof. M.Sc. Valéria Alves da Silva
Universidade Federal do Espírito Santo

Prof. Dr. Rodrigo Freitas Silva
Universidade Federal do Espírito Santo

Alegre - ES
2020

Agradecimentos

Agradeço a todos que me apoiaram até aqui e não me deixaram desistir.

Resumo

O Problema de Tabela-Horário em Universidades (PTHU) possui grande relevância no âmbito acadêmico, por ser um processo complexo e desgastante que deve ser realizado todo semestre. O problema consiste em alocar um conjunto de aulas em salas disponíveis e em períodos pré-determinados, de tal forma que a solução encontrada atenda os anseios dos envolvidos da melhor forma possível. Sendo assim, este trabalho propõe o estudo do impacto da escolha do fator de aleatoriedade e de movimentos na meta-heurística híbrida *Greedy Randomized Adaptive Search Procedure* (GRASP) com *Simulated Annealing* (SA), que em outro trabalho já mostrou ser eficiente na resolução do PTHU do Departamento de Computação (DCOMP) do Centro de Ciências Exatas, Naturais e da Saúde (CCENS) da Universidade Federal do Espírito Santo (UFES). Dessa maneira, um novo movimento, denominado *Lecture Move*, foi implementado e o PTHU do DCOMP-CCENS-UFES, foi resolvido combinando diferentes fatores de aleatoriedade e movimentos. Os resultados obtidos são comparados com as melhores soluções disponíveis na literatura e também com as soluções manuais elaboradas pelos coordenadores de curso. Esses resultados indicam que a escolha do fator de aleatoriedade e dos movimentos usados na busca local impactam diretamente na qualidade das soluções obtidas.

Palavras-chaves: Problema de Tabela-Horário em Universidades; Meta-heurística; GRASP; Simulated Annealing; Lecture Move.

Abstract

The University Timetabling Problem (PTHU) has great relevance in the academic field, being a complex and exhausting process that must be done every semester. The problem consists in allocating a set of classes in rooms available at predetermined intervals, in such a way that a found solution meets the wishes of those involved in the best possible way. Therefore, this work proposes the study of the impact of the choice of the factor of randomness and movements in the hybrid metaheuristic Greedy Randomized Adaptive Search Procedure (GRASP) with Simulated Annealing (SA), which in another work has already proved to be efficient in solving the PTHU of the Computing Department (DCOMP) of the Center for Exact, Natural and Health Sciences (CCENS) of the Federal University of Espirito Santo (UFES). In this way, a new movement, called Lecture Move, was implemented and the PTHU of the DCOMP-CCENS-UFES, was solved by combining different factors of randomness and movements. The results obtained are compared with the best solutions available in the literature and also with the manual solutions developed by the course coordinators. These results indicate that the choice of the randomness factor and the movements used in the local search directly impact the quality of the solutions obtained.

Keywords: University timetabling problem; Metaheuristic; GRASP; Simulated Annealing; Lecture Move.

Lista de ilustrações

Figura 1 – Exemplo de uma iteração do construtor de soluções	24
Figura 2 – Exemplo de um movimento do tipo <i>Move</i>	27
Figura 3 – Exemplo de um movimento do tipo <i>Swap</i>	27

Lista de tabelas

Tabela 1 – Estrutura da entidade sala.	17
Tabela 2 – Estrutura da entidade professor.	17
Tabela 3 – Estrutura da entidade disciplina.	18
Tabela 4 – Estrutura da entidade tipo de sala.	18
Tabela 5 – Estrutura da entidade turma.	18
Tabela 6 – Estrutura da entidade horário.	18
Tabela 7 – Estrutura da entidade oferta	18
Tabela 8 – Representação de uma tabela-horário de sala com ofertas alocadas . . .	19
Tabela 10 – Representação de uma tabela-horário de turma com salas alocadas . .	20
Tabela 9 – Representação de uma tabela-horário de professor com salas alocadas .	20
Tabela 11 – Tabela-horário depois de uma iteração do procedimento de criar soluções.	25
Tabela 12 – Parâmetros e valores dos algoritmos	29
Tabela 13 – Penalizações e valores do PTHU	30
Tabela 14 – Melhores Soluções, médias das soluções e tempo para 2013/2 com $\alpha =$ 0.15	31
Tabela 15 – Melhores Soluções, médias das soluções e tempo para 2016/1 com $\alpha =$ 0.15	31
Tabela 16 – Melhores soluções, média das soluções e tempo para o Movimento Lecture Move com variação de alfa	32
Tabela 17 – Melhores soluções, média das soluções e tempo para o Movimento Lecture Move	32
Tabela 18 – Lista de locais disponíveis	37
Tabela 19 – Lista de professores	38
Tabela 20 – Lista de disciplinas.	39
Tabela 21 – Lista de tipos de salas	39
Tabela 22 – Lista de turmas	40
Tabela 23 – Lista de horários	40
Tabela 24 – Lista de ofertas.	43
Tabela 25 – Lista de locais disponíveis	44
Tabela 26 – Lista de professores	44
Tabela 27 – Lista de disciplinas.	46
Tabela 28 – Lista de tipos de salas	46
Tabela 29 – Lista de turmas	46
Tabela 30 – Lista de horários	46
Tabela 31 – Lista de ofertas.	49
Tabela 32 – Prédio Central - Sala 09	50

Tabela 33 – Prédio Central - Sala 04	51
Tabela 34 – Prédio Novo - Sala 03	51
Tabela 35 – Prédio Novo - Sala 01	52
Tabela 36 – Prédio Novo - Sala 12	53
Tabela 37 – Prédio Novo - Sala 09	53
Tabela 38 – Prédio Antigo - Sala 03	54
Tabela 39 – Prédio Central - Sala 0	54
Tabela 40 – ChiChiu - Laboratório 03	55
Tabela 41 – ChiChiu - Laboratório 02	56
Tabela 42 – ChiChiu - Laboratório 01	57
Tabela 43 – Reuni - Laboratório 07	58
Tabela 44 – Reuni - Laboratório 05	58
Tabela 45 – Ciência da Computação - 2º Período	59
Tabela 46 – Ciência da Computação - 4º Período	60
Tabela 47 – Ciência da Computação - 6º Período	61
Tabela 48 – Ciência da Computação - 8º Período	61
Tabela 49 – Sistemas de Informação - 2º Período	62
Tabela 50 – Sistemas de Informação - 4º Período	62
Tabela 51 – Sistemas de Informação - 6º Período	63
Tabela 52 – Sistemas de Informação - 8º Período	63
Tabela 53 – Matemática - 1º Período	64
Tabela 54 – Agronomia - 1º Período	64
Tabela 55 – Engenharia de Alimentos - 3º Período	65
Tabela 56 – Geologia - 7º Período	65
Tabela 57 – Engenharia Florestal - 1º Período	66
Tabela 58 – Sistemas de Informação - 1º Período	66

Índice de algoritmos

1	Algoritmo do GRASP apresentado por Feo e Resende (1989)	23
2	Algoritmo do <i>SA</i> adaptado de Rocha (2013)	26
3	Algoritmo do <i>Lecture Move</i>	28

Lista de abreviaturas e siglas

ALNS	Adaptive Large Neighborhood Search
CCENS	Centro de Ciências Exatas, Naturais e da Saúde
DCOMP	Departamento de Computação
FO	Função Objetivo
GRASP	Greedy Randomized Adaptive Search Procedure
ITC	International Timetabling Competition
LC	Lista de Candidatos
LM	Lecture Move
LRC	Lista Restrita de Candidatos
PTHU	Problema de Tabela-Horário em Universidades
SA	Simulated Annealing
UFES	Universidade Federal do Espírito Santo

Sumário

1	INTRODUÇÃO	12
1.1	O problema e sua importância	12
1.2	Objetivos	14
1.2.1	Objetivo Geral	14
1.2.2	Objetivos Específicos	14
2	REVISÃO DE LITERATURA	15
3	METODOLOGIA	17
3.1	Modelagem do Problema	17
3.2	Função Objetivo	20
3.3	GRASP	22
3.3.1	Construção da solução inicial	23
3.3.2	Busca Local	24
3.3.2.1	<i>Simulated Annealing</i>	25
3.3.2.2	<i>Lecture Move</i>	26
4	RESULTADOS COMPUTACIONAIS	29
4.1	Escolha de Parâmetros	29
4.2	Detalhes de Implementação	30
4.3	Análise dos Resultados	30
5	CONCLUSÕES	33
	REFERÊNCIAS	34
	APÊNDICE A – LISTA DE DADOS DO CCENS-UFES	37
A.1	Dados 2013/2	37
A.2	Dados 2016/1	44
	APÊNDICE B – TABELA DE HORÁRIOS OBTIDAS PELO GRASP PARA SALAS DO PERÍODO 2013/2	50
	APÊNDICE C – TABELA DE HORÁRIOS OBTIDAS PELO GRASP PARA TURMAS DO PERÍODO 2016/1	59

1 Introdução

O problema de tabela-horário é um problema de Otimização Combinatória (OC), que deriva do problema de escalonamento e foi definido por [Wren \(1995\)](#) como a alocação, submetida a restrições, de eventos em um número limitado de períodos de tempo e locais, de forma a satisfazer, tanto quanto possível, um conjunto de objetivos estabelecidos.

1.1 O problema e sua importância

O problema de tabela-horário pode ser aplicado a diversos tipos de situações, entre os quais pode-se citar: escalas de trabalhadores, escalas de condutores de veículos de transporte, escalas de competições esportivas, e tabelas de horário educacionais, sendo este último o foco deste trabalho, por ser um dos mais estudados na área, conforme pode ser observado em [Schaerf \(1999\)](#).

Os problemas de tabela-horário educacionais abordam, por exemplo, a geração de tabela-horário para escolas de ensino médio e universidades. O problema de tabela-horário de universidades possui diversas formulações. Isso ocorre pois cada instituição de ensino possui diferentes restrições do problema.

A complexidade do Problema de Tabela-Horário de Universidades (PTHU) é uma das maiores da área de otimização combinatória, e aumenta à medida que são adicionadas restrições, ou seja, quanto mais restrições o problema tiver, mais difícil será encontrar uma solução que atenda todas elas. Segundo [Schaerf \(1999\)](#), esse problema é classificado como NP-completo para grande parte das formulações, isso significa que a solução ótima só pode ser encontrada rapidamente para instâncias muito pequenas, o que não é a realidade da maioria das universidades brasileiras. Desta forma, busca-se minimizar, através de uma solução automática, o esforço manual na geração de tabela-horário, bem como isentar o processo de um possível viés na alocação dos horários, por parte dos docentes envolvidos.

Apesar das diferentes formulações, segundo [Santos e Souza \(2007\)](#), os problemas de tabela-horário educacionais possuem uma característica em comum: a separação das restrições em dois grupos, denominados de restrições fortes e restrições fracas. Isso é feito dessa maneira, pois geralmente não é possível encontrar uma solução que atenda todas as restrições impostas.

As restrições fortes são aquelas que não podem ser violadas. Este grupo restringe o conjunto de soluções para impedir situações irreais. Se uma tabela-horário não viola nenhuma restrição forte, ela é considerada uma solução viável.

As restrições fracas são aquelas cuja satisfação é desejável, mas caso não seja

possível atendê-las, a solução não é inviabilizada. Essas restrições possuem pesos para refletir sua importância na qualidade da solução.

Este trabalho leva em consideração o PTHU do caso real do Departamento de Computação (DCOMP) do Centro de Ciências Exatas, Naturais e da Saúde (CCENS) da Universidade Federal do Espírito Santo (UFES), no qual foram inicialmente identificadas 15 restrições por [Mariano \(2014\)](#), e posteriormente adicionadas mais 2 restrições por [Moreira et al. \(2016\)](#). A seguir, as 17 restrições consideradas neste trabalho são apresentadas.

- Restrições Fortes:

1. Conflitos de professor: um professor não poderá ministrar mais de uma disciplina no mesmo dia e horário;
2. Conflitos de turmas: uma turma não poderá assistir a mais de uma aula no mesmo dia e horário;
3. Conflitos de salas: uma sala de aula não poderá estar reservada para mais de uma disciplina no mesmo dia e horário;
4. Aulas fora do turno: uma aula não poderá ser alocada fora do turno da oferta (diurno ou noturno).
5. Capacidade da sala: uma turma não poderá ser alocada em uma sala cuja capacidade seja inferior ao número de alunos da turma;
6. Tipo incompatível de sala: as aulas não poderão ser alocadas em uma determinada sala que não é compatível ao tipo solicitado, por exemplo, aulas que deveriam ser realizadas em laboratórios e foram alocadas em salas de aula;
7. “Disciplinas especiais”: disciplinas com 3 horas aulas semanais deverão ser alocadas nos três primeiros horários do turno diurno, e nos três primeiros ou três últimos horários do turno noturno, permitindo assim que outras disciplinas possam ser alocadas;

- Restrições Fracas:

8. Intervalo de trabalho do professor: o intervalo entre o primeiro e o último dia da semana em que um professor ministrará as aulas deverá ser minimizado;
9. Janelas de horário: intervalos na grade de horários de cada turma, entre duas aulas, deverão ser reduzidos;
10. Período preferencial: as turmas diurnas deverão ter suas disciplinas concentradas no período da manhã ou da tarde. Assim, a quantidade de disciplinas ofertadas fora do turno “preferencial” de cada turma deverá ser minimizada;
11. Aulas seguidas: aulas repetidas de uma disciplina ministradas para uma turma no mesmo dia devem ser evitadas;

12. Intervalo entre períodos: a ocorrência de professores que ministram aula em um dia à noite e no dia seguinte pela manhã deverá ser minimizada;
13. Aulas seguidas de nível “difícil”: as aulas de complexidade “difícil” ministradas em horários sequenciais devem ser evitadas;
14. Aulas de nível “difícil” no último horário: aulas de complexidade “difícil” ministradas no último horário de cada dia deverão ser evitadas.
15. Aulas de carga horária par: aulas com 2 ou 4 horas do turno diurno deverão ser alocadas fora do primeiro horário do dia.
16. Aulas alocadas imediatamente antes, ou imediatamente depois, do horário de almoço devem ser evitadas. Ex: Uma aula com carga horária de 2 horas, sendo alocada de 11:00 às 13:00 ou 12:00 às 14:00
17. Aulas com alocação iniciada fora do horário padrão. Ex: Uma aula sendo alocada de 13:00 às 14:00 ou 15:00 às 16:00

1.2 Objetivos

Os objetivos deste trabalho são divididos em objetivo geral e objetivos específicos.

1.2.1 Objetivo Geral

Estudar o impacto da escolha do fator de aleatoriedade e de movimentos, entre eles o *Lecture Move*, na meta-heurística híbrida *Greedy Randomized Adaptive Search Procedure* (GRASP) com *Simulated Annealing* (SA), quando aplicado ao PTHU do DCOMP-CCENS-UFES

1.2.2 Objetivos Específicos

- (a) Estudar novos movimentos a serem usados na busca local do problema abordado;
- (b) Implementar um novo movimento na etapa de busca local;
- (c) Realizar experimentos computacionais;
- (d) Avaliar o desempenho do GRASP com SA usando o novo movimento implementado e diferentes fatores de aleatoriedade;
- (e) Comparar os resultados obtidos com os resultados apresentados por [Moreira et al. \(2016\)](#), bem como as soluções construídas manualmente pelos coordenadores de curso.

2 Revisão de Literatura

Na década de 60, [Gotlieb \(1962\)](#), através de resoluções a partir de análise combinatória, iniciou estudos relacionados ao problema de tabela-horário de instituições de ensino. Desde então, o tema ganhou atenção de pesquisadores ([SCHAERF, 1999](#); [LEWIS, 2008](#)).

Devido a grande variedade de restrições específicas a cada instituição de ensino, o problema de tabela-horário não possui uma formulação única.

De acordo com [Schmidt e Ströhlein \(1980\)](#) e [Souza \(2000\)](#), os primeiros trabalhos utilizavam heurísticas construtivas, mas desde então, pesquisadores passaram a utilizar outras técnicas para a resolução do problema, como exemplo, representando o problema com grafos e resolvendo com algoritmos de fluxo ([OSTERMANN; WERRA, 1982](#)), ou coloração ([WOOD, 1969](#); [NEUFELD; TARTAR, 1975](#); [ČANGALOVIĆ; SCHREUDER, 1991](#)). Outros trabalhos também usaram programação inteira mista ([TRIPATHY, 1984](#); [FERLAND; ROY, 1985](#)).

Em [Lewis \(2008\)](#) pode ser observado que grande parte dos trabalhos recentes tem utilizado meta-heurísticas, tanto pela simplicidade, quanto pelos bons resultados alcançados. *Simulated Annealing* ([MARIANO, 2014](#)), Busca Tabu ([MACHADO; BOERES, 2009](#)), Algoritmos Genéticos ([BURKE; ELLIMAN; WEARE, 1994](#)) e Algoritmos Meméticos ([BURKE; NEWALL; WEARE, 1995](#)) são as meta-heurísticas mais utilizadas. Em alguns trabalhos se observa também combinação de meta-heurísticas ([MOREIRA et al., 2016](#)).

Na primeira edição do campeonato internacional de tabela-horário (*International Timetabling Competition* - ITC), realizado em 2002 (ITC-2002), [Kostuch \(2004\)](#) desenvolve um algoritmo que constrói a tabela-horário em três etapas. Na primeira etapa é usado um algoritmo de coloração de grafos, que obtém uma solução inicial viável. Na segunda e terceira etapas aplica-se o *Simulated Annealing*, sendo que em cada etapa é utilizada uma estrutura diferente de vizinhança.

[Müller \(2009\)](#) resolve o problema de tabela-horário de universidades da terceira formulação da segunda edição do ITC, realizado em 2007 (ITC-2007), usando *Conflict-based Statistics* para gerar a solução inicial e *Hill Climbing* combinado com *Great Deluge* e *Simulated Annealing* para refinamento da solução. [Rocha \(2013\)](#) trata do mesmo problema aplicando a meta-heurística *Greedy Randomized Adaptive Search Procedure* (GRASP), sendo testados os métodos *Hill Climbing* e *Simulated Annealing* como métodos de busca local, e o método *Path-Relinking* também é aplicado, mas para intensificar a busca por soluções de boa qualidade. [Segatto et al. \(2017\)](#) expande o trabalho de [Rocha \(2013\)](#) e implementa novas vizinhanças. Dessa forma, proporcionou um maior entendimento do processo de busca no espaço de soluções e conseguiu melhorar o desempenho do algoritmo.

Elmohamed, Coddington e Fox (1997) investigaram diversas formas de aplicar o *Simulated Annealing* no PTHU da Universidade de Syracuse. Dentre as configurações investigadas, os melhores resultados foram obtidos com resfriamento adaptativo, reaquecimento e um algoritmo baseado em regras, que é usado para gerar uma boa solução inicial. Ceschia, Gaspero e Schaerf (2012) usam *Simulated Annealing* para resolver a terceira formulação do ITC-2007. Os autores conseguiram boas respostas para as instâncias usadas no ITC-2007, e em algumas instâncias foram obtidas melhores soluções que as conhecidas até aquele momento.

Mariano (2014) utilizou a meta-heurística *Adaptive Large Neighborhood Search* (ALNS), aplicada ao PTHU de um caso real do DCOMP-CCENS-UFES.

Ainda para o PTHU do DCOMP-CCENS-UFES, Moreira et al. (2016) apresentou uma proposta de utilizar a meta-heurística *Greedy Randomized Adaptive Search Procedure* (GRASP), com o *Simulated Annealing* na fase de busca local, tendo encontrado soluções melhores do que as feitas manualmente pelos coordenadores de curso. Carvalho et al. (2016) resolve o mesmo problema que Mariano (2014) e Moreira et al. (2016), porém usa um método guloso para construir uma solução inicial que é refinada através da meta-heurística *Simulated Annealing*.

3 Metodologia

Nesse capítulo é apresentada a metodologia proposta para o desenvolvimento deste trabalho, em que é descrita a forma como o problema foi modelado, bem como as instâncias utilizadas nos testes do estudo proposto.

3.1 Modelagem do Problema

A modelagem usada para representar o PTHU do DCOMP-CCENS-UFES foi baseada nos trabalhos de [Mariano \(2014\)](#) e [Moreira et al. \(2016\)](#), bem como as instâncias que estes trabalhos utilizaram (2013/2 e 2016/1). O problema é modelado com estruturas definidas pelo identificador de cada elemento relevante e das características seguintes ao identificador.

As estruturas são armazenadas em vetores que possibilitam estabelecer as relações entre os conjuntos de elementos que são trabalhados. Desse modo, as estruturas mais simples são: salas, professores, disciplinas, turmas e horários. Além dessas, é estabelecida uma estrutura denominada oferta, que promove a relação entre todas as outras. O levantamento de todas as entidades que referem-se aos períodos 2013/2 e 2016/1, usados nos testes deste trabalho, constam no Apêndice [A](#).

Para a entidade sala é definida uma estrutura capaz de armazenar as informações necessárias, de modo que a coluna ‘Prédio’ possui a sigla da localização, a coluna ‘Tipo’ significa se é um laboratório ou uma sala comum, além de seu número de identificação e a capacidade de alunos que ela comporta.

Id	Prédio	Tipo	Número	Capacidade
0	PC	0	9	40

Tabela 1 – Estrutura da entidade sala.

Para a entidade professor é definida uma estrutura capaz de armazenar o identificador e o nome.

Id	Nome
0	Edmar Hell Kampke

Tabela 2 – Estrutura da entidade professor.

Para a entidade disciplina é criada uma estrutura com o identificador da disciplina, bem como seu código oficial, seu nome e o nível de dificuldade atribuído à esta disciplina.

Id	Código	Nome	Nível
0	COM06842	Programação I	1

Tabela 3 – Estrutura da entidade disciplina.

Para a entidade tipo de sala é criada uma estrutura que armazena o identificador do tipo de sala e sua descrição, que pode ser uma sala comum ou um laboratório.

Id	Descrição
0	Sala

Tabela 4 – Estrutura da entidade tipo de sala.

Para a entidade turma é definida uma estrutura que possui além do identificador da turma, o nome do curso, representado por uma sigla, o período da turma em questão e seu turno preferencial (0, de manhã, 1, à tarde e 2, à noite).

Id	Curso	Período	Turno Preferencial
0	CC	1	1

Tabela 5 – Estrutura da entidade turma.

Para entidade horário, é criada estrutura para armazenar cada faixa de horário, de modo que corresponda a um período unitário. Desse modo, é definido um vetor com o identificador do horário, o horário inicial e final e a duração em minutos.

Id	Horário Inicial	Horário Final	Duração
0	07h00	08h00	60

Tabela 6 – Estrutura da entidade horário.

Para a entidade oferta, é criada uma estrutura capaz de conter vários elementos. Sendo assim, uma estrutura que representa uma oferta deve possuir o seu identificador, o identificador da disciplina que será ofertada, o número de turmas que poderá se matricular nesta oferta, bem como os identificadores destas turmas, o número de vagas que serão disponibilizadas para os alunos, o turno em que esta oferta será destinada, o identificador do professor responsável por esta oferta, além do tipo de sala que será necessária para conduzir as aulas e a carga horária da oferta, que no caso abaixo, representa 2 horas-aula seguidas.

Id	Disciplina	Nº de turmas	Turmas[]	Vagas	Turno	Professor	Tipo de Sala	CH
0	0	2	0 18	37	1	12	1	2

Tabela 7 – Estrutura da entidade oferta

Com todos os dados armazenados nas estruturas previamente definidas, o algoritmo inicia o processo de criação de uma solução inicial, que cria as três tabelas-horário essenciais para a solução final do PTHU, que são: a tabela-horário de cada sala, na qual deve constar todas as ofertas que serão lecionadas naquele local, a tabela-horário de cada professor, na qual ficam armazenados as ofertas que o professor leciona, além da tabela-horário de cada turma, que consta as ofertas das disciplinas da turma.

Nas Tabelas 8, 9 e 10 são apresentadas como as tabelas-horário, respectivamente, de uma sala, de um professor e uma turma são representadas. As células que possuem número positivos (destaque em cinza) são aquelas com ofertas alocadas. Esses números representam o identificador da oferta ali alocada.

	Sala X				
Horário	Segunda	Terça	Quarta	Quinta	Sexta
07:00-08:00	-1	-1	-1	-1	-1
08:00-09:00	-1	-1	-2	17	-2
09:00-10:00	-1	-1	-2	17	-2
10:00-11:00	-1	92	-2	27	-1
11:00-12:00	-1	92	-2	27	-1
13:00-14:00	-2	-2	-2	-2	18
14:00-15:00	-2	-2	-2	-2	18
15:00-16:00	-1	-2	2	-1	-2
16:00-17:00	-1	-2	2	-1	-2
18:00-19:00	-2	-2	-2	-2	-2
19:00-20:00	-2	-2	-2	-2	-2
20:00-21:00	-2	-2	-2	-2	-2
21:00-22:00	-2	-2	-2	-2	-2
22:00-23:00	-2	-2	-2	-2	-2

Tabela 8 – Representação de uma tabela-horário de sala com ofertas alocadas

	Turma Z				
Horário	Segunda	Terça	Quarta	Quinta	Sexta
07:00-08:00	-1	-1	-1	-1	-1
08:00-09:00	-1	-1	-1	-1	-1
09:00-10:00	-1	-1	-1	-1	-1
10:00-11:00	0	-1	-1	-1	2
11:00-12:00	0	-1	-1	-1	2
13:00-14:00	3	-1	7	9	-1
14:00-15:00	3	-1	7	9	-1
15:00-16:00	-1	-1	5	9	-1
16:00-17:00	-1	-1	5	9	-1
18:00-19:00	-1	-1	-1	-1	-1
19:00-20:00	-1	-1	-1	-1	-1
20:00-21:00	-1	-1	-1	-1	-1
21:00-22:00	-1	-1	-1	-1	-1
23:00-22:00	-1	-1	-1	-1	-1

Tabela 10 – Representação de uma tabela-horário de turma com salas alocadas

	Professor Y				
Horário	Segunda	Terça	Quarta	Quinta	Sexta
07:00-08:00	-1	-1	-1	-1	-1
08:00-09:00	11	-1	-1	-1	-1
09:00-10:00	11	-1	-1	-1	-1
10:00-11:00	11	-1	-1	-1	-1
11:00-12:00	11	-1	-1	-1	-1
13:00-14:00	-1	-1	-1	-1	-1
14:00-15:00	-1	-1	-1	-1	-1
15:00-16:00	-1	8	-1	4	-1
16:00-17:00	-1	8	-1	4	-1
18:00-19:00	-1	8	-1	-1	-1
19:00-20:00	-1	8	-1	-1	-1
20:00-21:00	-1	-1	-1	-1	-1
21:00-22:00	-1	-1	-1	-1	-1
22:00-23:00	-1	-1	-1	-1	-1

Tabela 9 – Representação de uma tabela-horário de professor com salas alocadas

3.2 Função Objetivo

Cada tabela-horário gerada recebe uma nota que reflete sua qualidade, a função f que calcula essa nota é chamada de Função Objetivo (FO). Cada restrição violada aumenta o valor da função objetivo de acordo com o peso da restrição. A melhor solução para um problema de tabela-horário é aquela que minimiza o valor da função objetivo.

Seja S uma solução do PTHU abordado, e considerando que $f_{RFt}(S)$ e $f_{RFc}(S)$ são o número total de violações das restrições fortes e fracas de S , respectivamente, então S é uma solução viável, se todas as restrições fortes são satisfeitas, ou seja $f_{RFt}(S) = 0$. De acordo com as restrições propostas por [Mariano \(2014\)](#) e posteriormente expandidas por [Moreira et al. \(2016\)](#), a função objetivo do PTHU do DCOMP-CCENS-UFES é calculada pela seguinte fórmula:

$$f(S) = f_{RFt}(S) + f_{RFc}(S)$$

sendo

$$f_{RFt}(S) = \omega_1 \sum_{p=1}^P CP_p + \omega_2 \sum_{t=1}^T CT_t + \omega_3 \sum_{s=1}^S CS_s + \omega_4 OFT + \omega_5 \sum_{s=1}^S VS_s + \omega_6 TSI + \omega_7 D3H$$

e

$$f_{RFc}(S) = \omega_8 \sum_{p=1}^P IT_p + \omega_9 \sum_{t=1}^T JH_t + \omega_{10} \sum_{t=1}^T PP_t + \omega_{11} \sum_{d=1}^D AS_d + \omega_{12} \sum_{p=1}^P ND_p + \\ \omega_{13} ASD + \omega_{14} ADU + \omega_{15} DHP + \omega_{16} AHA + \omega_{17} AHFP$$

Considerando que:

1. CP_p : número de conflitos do professor p , ou seja, o número de vezes que o professor p ministra aula no mesmo dia e horário;
2. CT_t : número de conflitos da turma t , ou seja, o número de vezes que os alunos da turma t assistem mais de uma aula no mesmo dia e mesmo horário;
3. CS_s : número de conflitos da sala s , ou seja, o número de vezes que a sala s está atribuída a mais de uma turma no mesmo dia e mesmo horário;
4. OFT : número de violações em que uma oferta é alocada fora do turno;
5. VS_s : número de violações na capacidade da sala s , ou seja, o número de turmas alocadas na sala s cujo número de alunos é maior que a capacidade da sala;
6. TSI : número de aulas alocadas em salas de tipo “incompatível”, ou seja, se 10 aulas devem ser em laboratório e foram alocadas em salas normais, $TSI = 10$;
7. $D3H$: número de disciplinas de 3 horas aulas semanais alocadas fora dos horários “padrão” (primeiro e último horário, tanto do dia quanto da noite);
8. IT_p : diferença entre o primeiro e o último dia em que o professor p ministra aulas em relação a um intervalo padrão I , que deverá ser um parâmetro de entrada. Nesse caso, deve-se contabilizar apenas o que exceder I . Ex: Considerando $I = 3$ e que o professor 1 dá aulas de segunda a quinta, logo $IT_1 = MAX(0, Quinta - Segunda + 1 - I) \rightarrow IT_1 = 1$; Caso o professor dê aulas de segunda a terça, $IT_1 = MAX(0, Terça - Segunda + 1 - I) \rightarrow IT_1 = 0$;

9. JH_t : número de janelas de horário da turma t , ou seja, o número de horários vagos entre aulas ao longo da semana para a turma t ;
10. PP_t : número aulas da turma t fora do seu período preferencial (M,T,N). Ex: o período preferencial para a turma 1 é a tarde (T), logo, PP_1 = número de aulas para essa turma alocadas no período da manhã;
11. AS_d : número de aulas seguidas da disciplina d , ou seja, o número de vezes que a disciplina d é repetida num mesmo dia;
12. ND_p : número de vezes que o professor p ministra aula à noite (qualquer horário) em um dia e pela manhã (qualquer horário) no dia seguinte;
13. ASD : número de aulas seguidas de disciplinas de nível difícil, ou seja, o número de vezes ao longo da semana em que duas disciplinas “difíceis” são consecutivas;
14. ADU : número de aulas de disciplinas de nível difícil ministradas no último horário da tarde ou da noite;
15. DHP : número de aulas de disciplinas, do turno diurno, com carga horária par, alocadas no primeiro horário do dia;
16. AHA : número de aulas de disciplinas, do turno diurno, alocadas entre o horário de almoço;
17. $AHFP$: número de aulas de disciplinas com alocação iniciada fora do horário padrão. Ex: uma aula sendo alocada de 13:00 às 14:00 ou 15:00 às 16:00.

O vetor $\omega = [\omega_1, \omega_2, \omega_3, \dots, \omega_{17}]$ contém os pesos das 17 restrições (7 fortes e 10 fracas) consideradas por [Moreira et al. \(2016\)](#) e apresentadas no Capítulo 1 deste trabalho.

3.3 GRASP

A meta-heurística *Greedy Randomized Adaptive Search Procedure* (GRASP), apresentada por [Feo e Resende \(1989\)](#) foi inicialmente utilizada para tratar o problema de cobertura de conjuntos.

De acordo com [Rocha \(2013\)](#), o GRASP já foi aplicado com sucesso em diversos problemas de otimização desde sua proposta inicial, tais como: conjunto independente máximo ([FEO; RESENDE; SMITH, 1994](#)), problema quadrático de alocação ([LI; PARDA-LOS; RESENDE, 1994](#)), satisfatividade ([RESENDE; FEO, 1996](#)), planarização de grafos ([RESENDE; RIBEIRO, 1997](#)), roteamento de circuitos virtuais ([RESENDE; RIBEIRO, 2003](#)), entre outros.

No Algoritmo 1 é apresentado o pseudocódigo genérico da meta-heurística GRASP.

Algoritmo 1: Algoritmo do GRASP apresentado por [Feo e Resende \(1989\)](#)

Entrada: $MaxIter, \alpha$
Saída: Solução S^{Melhor}

```

1 início
2    $FO^{Melhor} \leftarrow \infty$ ;
3   para  $i \leftarrow 1$  até  $MaxIter$  faça
4      $S^{Inicial} \leftarrow GeraSolucaoInicial(\alpha)$ ;
5      $S^{Atual} \leftarrow BuscaLocal(S^{Inicial})$ ;
6     se  $f(S^{Atual}) < FO^{Melhor}$  então
7        $S^{Melhor} \leftarrow S^{Atual}$ ;
8        $FO^{Melhor} \leftarrow f(S^{Atual})$ ;
9     fim se
10  fim para
11 fim

```

O algoritmo GRASP é um procedimento iterativo dividido em duas fases, que são detalhadas nas próximas seções. Na primeira fase constrói-se uma solução inicial, e na segunda fase aplica-se uma busca local para melhorá-la. A resposta final é a melhor obtida após a execução de todas as iterações ([FEO; RESENDE, 1995](#)). Para essa meta-heurística são utilizados dois parâmetros principais: o número máximo de iterações $MaxIter$ e o fator de aleatoriedade (α).

3.3.1 Construção da solução inicial

Inicialmente, uma solução vazia (sem ofertas alocadas) para o PTHU do DCOMP-CCENS-UFES é criada. Nessa solução inicial, todas as células de cada tabela-horário terá valor negativo, sendo -1 se o horário estiver disponível e -2 se naquele horário já há uma oferta de outro departamento da UFES alocada.

A partir disso, uma lista de ofertas, contendo todas as ofertas, é criada. Essa lista é ordenada de forma decrescente pelo número de vagas. A partir da lista inicial, são feitas simulações de alocação em todas as posições da matriz de salas, com a oferta que possui o maior número de vagas. Cada simulação tem seu custo calculado, e cada uma dessas simulações é inserida em uma lista denominada Lista de Candidatos (LC).

A LC possui a posição em que a simulação de alocação foi efetuada e seu custo de alocação. A LC serve de base para a formação de uma outra lista, denominada Lista Restrita de Candidatos (LRC), que é composta por um intervalo da LC, depois dela ter sido ordenada crescentemente pelo custo de alocação das simulações. Esse intervalo é definido de acordo com o fator de aleatoriedade α da seguinte forma: $[c^{min}, c^{min} + \alpha(c^{max} - c^{min})]$

em que c^{min} representa o menor custo de alocação, c^{max} representa o maior custo de alocação, e $\alpha \in [0, 1]$.

Quando $\alpha = 1$, LC tem seu elementos totalmente selecionados para LRC, se $\alpha = 0$, LRC é preenchida somente com o primeiro elemento de LC. Um horário é escolhido de LRC aleatoriamente e a oferta é acrescentada à solução. É importante destacar que quando $\alpha = 1$ a escolha do horário é feita de maneira totalmente aleatória, caso $\alpha = 0$, o horário de LC que possui o menor custo será sempre escolhido, o que representa uma escolha totalmente gulosa.

Após a alocação da oferta, ela é removida da lista de ofertas e o procedimento é repetido até que todas as ofertas sejam alocadas. Portanto, a construção da solução inicial termina quando todas as ofertas estiverem inseridas na tabela-horário.

Na Figura 1 é exibido um exemplo de uma iteração da fase de construção de uma solução inicial no método GRASP. Nesse exemplo, a LC da oferta ainda não alocada e que possui maior número de vagas é criada com seis possíveis horários para alocação ordenados crescentemente pelo custo de alocação, calculado durante cada simulação. Em seguida, é formada a LRC, com apenas três horários, supondo o fator de aleatoriedade $\alpha = 0,5$. Dentre esses três horários é feita uma escolha aleatória para definir em qual horário a oferta é alocada. Essa alocação é representada na Tabela 11 com a alocação da oferta cujo identificador é 0 (destaque em laranja).

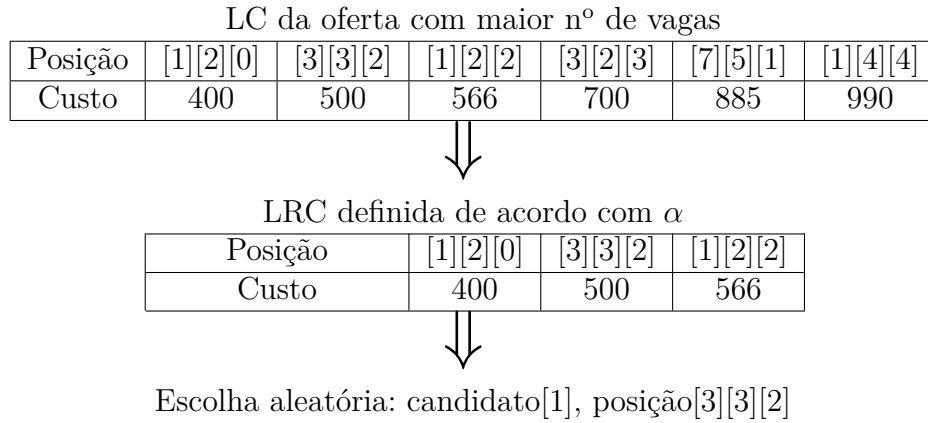


Figura 1 – Exemplo de uma iteração do construtor de soluções

3.3.2 Busca Local

O objetivo da busca local é melhorar a solução inicialmente construída. A estratégia de busca local utilizada foi a meta-heurística *Simulated Annealing*, pois ela se mostrou satisfatória em outros trabalhos, como de Rocha (2013), Mariano (2014) e Carvalho et al. (2016).

	Sala X				
Horário	Segunda	Terça	Quarta	Quinta	Sexta
07:00-08:00	-1	-1	-1	-1	-1
08:00-09:00	-1	-1	-1	-1	-1
09:00-10:00	-1	-1	-1	-1	-1
10:00-11:00	-1	-1	0	-1	-1
11:00-12:00	-1	-1	0	-1	-1
13:00-14:00	-1	-1	-1	-1	-1
14:00-15:00	-1	-1	-1	-1	-1
15:00-16:00	-1	-1	-1	-1	-1
16:00-17:00	-1	-1	-1	-1	-1
18:00-19:00	-1	-1	-1	-1	-1
19:00-20:00	-1	-1	-1	-1	-1
20:00-21:00	-1	-1	-1	-1	-1
21:00-22:00	-1	-1	-1	-1	-1
22:00-23:00	-1	-1	-1	-1	-1

Tabela 11 – Tabela-horário depois de uma iteração do procedimento de criar soluções.

3.3.2.1 *Simulated Annealing*

A meta-heurística *Simulated Annealing* tem como inspiração o processo da metalurgia conhecido como recozimento, em que derrete-se um metal e ele é esfriado lentamente até a solidificação, de forma controlada, para que sua estrutura se mantenha organizada e equilibrada (KIRKPATRICK; GELATT; VECCHI, 1983).

Os cinco principais parâmetros que o *Simulated Annealing* possui são: a solução inicial (S), a temperatura inicial (T_i), a temperatura final (T_f), a taxa de resfriamento (β) e o número de soluções vizinhas (N_v) que deverão ser geradas a cada iteração.

O algoritmo começa com uma temperatura inicial e vai sendo resfriada até chegar na temperatura final. Em cada temperatura (iteração), começando com a temperatura inicial T_i , são gerados N_v vizinhos. Se o vizinho gerado tem uma FO menor que a da solução atual, esta é atualizada. Se o vizinho tiver uma FO maior, ele pode ser aceito com uma probabilidade $e^{-\Delta f/T}$, em que Δf é a diferença entre os valores das FO's da solução vizinha e da solução atual, e T é a temperatura atual. Quanto maior Δf , e menor a temperatura atual T , menor será a chance de aceitar uma solução vizinha. Tipicamente, o algoritmo aceita grande diversificação de soluções no início, quando a temperatura está alta. À medida que a temperatura decresce, poucas soluções piores são aceitas e assim é intensificada a busca em uma determinada vizinhança. Ao final de cada iteração, a temperatura T é multiplicada pelo valor β , sendo $\beta \in [0, 1]$ a taxa de resfriamento. O Algoritmo termina quando $T < T_f$.

O Algoritmo 2 apresenta o pseudocódigo da meta-heurística *Simulated Annealing* aplicada na fase da busca local.

Algoritmo 2: Algoritmo do SA adaptado de Rocha (2013)

Entrada: Solução S , T_i , T_f , β , N_v
Saída: Solução S^{Melhor}

```

1 início
2    $T \leftarrow T_i$ ;
3    $S^{Melhor} \leftarrow S$ ;
4   enquanto  $T > T_f$  faça
5     para  $i \leftarrow 1$  até  $MaxIter$  faça
6        $S^{Vizinho} \leftarrow GeraVizinho(S^{Atual})$ ;
7        $\Delta f \leftarrow f(S^{Vizinho}) - f(S^{Atual})$ ;
8       se  $\Delta f < 0$  então
9          $S^{Atual} \leftarrow S^{Vizinho}$ ;
10      senão
11        Gera um número aleatório  $\rho \in [0, 1]$ ;
12        se  $\rho < e^{-\Delta f/T}$  então
13           $S^{Atual} \leftarrow S^{Vizinho}$ ;
14        fim se
15      fim se
16      se  $f(S^{Vizinho}) < f(S^{Melhor})$  então
17         $S^{Melhor} \leftarrow S^{Vizinho}$ ;
18      fim se
19    fim para
20     $T \leftarrow T * \beta$ ;
21  fim enquanto
22 fim

```

Para gerar novas soluções vizinhas da solução atual, foram utilizados três movimentos. Os movimentos clássicos *Move* e *Swap* implementados por Moreira et al. (2016), e o movimento *Lecture Move*, baseado neles e proposto por Müller (2009).

3.3.2.2 Lecture Move

Uma vizinhança é definida como um movimento ou um conjunto de movimentos realizados que geram um subconjunto de soluções no espaço de soluções. De acordo com Schaerf (1999), os clássicos *Move* e *Swap* são os movimentos de vizinhança que estão entre os mais utilizados para problemas de tabela-horário.

O movimento *Move* consiste em mover uma oferta escolhida aleatoriamente para uma posição não ocupada na tabela-horário, também escolhida aleatoriamente. Na Figura 2 ocorre a realização de um movimento do tipo *Move*. Na Figura 2a é apresentada uma tabela-horário antes da execução do movimento, e na Figura 2b, a tabela-horário após a execução do movimento. A oferta de código 92 previamente alocada na Terça-feira, no horário de 10:00-12:00 (Figura 2a) é movida para Segunda-feira no horário de 08:00-10:00 (Figura 2b). Nota-se que no *Move*, a oferta selecionada pode ser movida para qualquer

umas das células (ofertas) vazias, desde que essa mudança não cause uma inviabilidade na solução, ou seja, desde que não infrinja nenhuma restrição forte.

	Sala X			
Horário	Segunda	Terça	Quarta	...
07:00-08:00	-1	-1	-2	...
08:00-09:00	-1	-1	27	...
09:00-10:00	-1	-1	27	...
10:00-11:00	-1	92	-2	...
11:00-12:00	-1	92	-2	...
⋮	⋮	⋮	⋮	

(a) Antes do *Move*

	Sala X			
Horário	Segunda	Terça	Quarta	...
07:00-08:00	-1	-1	-2	...
08:00-09:00	92	-1	27	...
09:00-10:00	92	-1	27	...
10:00-11:00	-1	-1	-2	...
11:00-12:00	-1	-1	-2	...
⋮	⋮	⋮	⋮	

(b) Depois do *Move*Figura 2 – Exemplo de um movimento do tipo *Move*

O movimento *Swap* consiste na troca entre duas ofertas escolhidas aleatoriamente na tabela-horário. Na Figura 3 é apresentado um exemplo de aplicação do *Swap*. Na Figura 3a está apresentada uma tabela-horário antes da execução do movimento, e na Figura 3b, a tabela-horário após a execução do movimento. A oferta de código **92** previamente alocada na Terça-feira, no horário de 10:00-12:00 (Figura 3a) é movida para Quarta-feira no horário de 08:00-10:00, enquanto a oferta de código **27**, previamente alocada nessa posição passa a ocupar o lugar da oferta **92** (Figura 3b). Nota-se que no *Swap*, as ofertas selecionadas podem ser movidas para qualquer umas das células (ofertas) compatíveis, desde que essa mudança não cause uma inviabilidade na solução, ou seja, desde que não infrinja nenhuma restrição forte.

	Sala X			
Horário	Segunda	Terça	Quarta	...
07:00-08:00	-1	-1	-2	...
08:00-09:00	-1	-1	27	...
09:00-10:00	-1	-1	27	...
10:00-11:00	-1	92	-2	...
11:00-12:00	-1	92	-2	...
⋮	⋮	⋮	⋮	

(a) Antes do *Swap*

	Sala X			
Horário	Segunda	Terça	Quarta	...
07:00-08:00	-1	-1	-2	...
08:00-09:00	-1	-1	92	...
09:00-10:00	-1	-1	92	...
10:00-11:00	-1	27	-2	...
11:00-12:00	-1	27	-2	...
⋮	⋮	⋮	⋮	

(b) Depois do *Swap*Figura 3 – Exemplo de um movimento do tipo *Swap*

Os movimentos *Move* e *Swap* foram implementados no trabalho de [Moreira et al. \(2016\)](#). Neste trabalho, utilizando como base os movimentos *Move* e *Swap* implementados por [Moreira et al. \(2016\)](#), foi implementado um novo movimento, denominado *Lecture Move*. O movimento *Lecture Move* consiste na troca entre duas ofertas ou entre uma oferta e um horário vazio. Em outras palavras, o *Lecture Move* corresponde a uma combinação de *Move* e *Swap*.

No Algoritmo 3 é apresentado o pseudocódigo do movimento *Lecture Move*.

Algoritmo 3: Algoritmo do *Lecture Move*

Entrada: Solução S

Saída: Solução $S^{Vizinha}$

```

1  início
2      Trocou  $\leftarrow false$ ;
3      enquanto não Trocou faça
4          Escolhe aleatoriamente posição 1 ( $p1$ );
5          Escolhe aleatoriamente posição 2 ( $p2$ );
6          se  $p1$  possui oferta alocada E  $p2$  possui oferta alocada então
7               $S^{Vizinha} \leftarrow Swap(S, p1, p2)$ ;
8              Trocou  $\leftarrow true$ ;
9          senão se  $p1$  possui oferta alocada E  $p2$  NÃO possui oferta alocada então
10              $S^{Vizinha} \leftarrow Move(S, p1, p2)$ ;
11             Trocou  $\leftarrow true$ ;
12          senão se  $p1$  NÃO possui oferta alocada E  $p2$  possui oferta alocada então
13              $S^{Vizinha} \leftarrow Move(S, p2, p1)$ ;
14             Trocou  $\leftarrow true$ ;
15      fim enquanto
16 fim

```

4 Resultados Computacionais

Este capítulo apresenta os experimentos computacionais e os resultados alcançados neste trabalho, constando as análises comparativas com outro trabalho, bem como as soluções manuais elaboradas pelos coordenadores de curso do DCOMP-CCENS-UFES. Mostra também as escolhas dos parâmetros do GRASP, os parâmetros específicos da busca local *Simulated Annealing* e os movimentos utilizados para a geração de soluções vizinhas.

4.1 Escolha de Parâmetros

O algoritmo GRASP possui dois parâmetros: o número máximo de iterações Max_{Iter} e o valor α , que define a forma como a construção da solução inicial será conduzida. Se o parâmetro α tem seu valor muito próximo a 0 (zero), o algoritmo de construção inicial tem comportamento predominantemente guloso, e produz soluções de boa qualidade, porém pouco diversificadas. Caso α é mais próximo de 1 (um), as soluções possuem características mais aleatórias, porém com a desvantagem de possuir um valor de função objetivo mais alto. Para o fator de aleatoriedade α foram escolhidos 5 valores distintos: 0, 0.25, 0.5, 0.75 e 1, para serem usados nos experimentos. Além desses valores, em uma das execuções do GRASP o valor usado não foi fixo, ou seja, em cada iteração um valor aleatório de α no intervalo $[0,1]$ é escolhido e usado. Para efeitos de comparação com o trabalho de [Moreira et al. \(2016\)](#), o valor de $\alpha = 0.15$ também foi testado. E da mesma forma também que [Moreira et al. \(2016\)](#), o parâmetro Max_{Iter} foi definido pelo tempo de execução: 500 segundos.

Para o *Simulated Annealing*, os parâmetros foram calibrados de forma a permitir um certo grau de diversificação no início da busca e maior intensificação no final do processo. Para cada valor de α e para cada instância (2013/2 e 2016/1) o SA foi testado com dois conjuntos de movimentos: *Move* e *Swap*, estes, com 50% de probabilidade cada e *Lecture Move*.

Na tabela 12 são apresentados os parâmetros definidos durante a calibração para o algoritmo SA.

Parâmetro	Descrição	Valor
T_i	Temperatura inicial para o SA	1000
T_f	Temperatura de congelamento para o SA	0.01
β	Taxa de resfriamento para o SA	0.975
N_v	Número máximo de iterações para o SA	500

Tabela 12 – Parâmetros e valores dos algoritmos

Para comparações dos resultados, os valores das penalizações (vetor ω) das restrições fortes e fracas da função objetivo, usadas neste trabalho, foram as mesmas adotadas por [Moreira et al. \(2016\)](#).

Penalização	Valor
ω_1	5000
ω_2	5000
ω_3	5000
ω_4	5000
ω_5	5000
ω_6	300
ω_7	5000
ω_8	10
ω_9	20
ω_{10}	4
ω_{11}	600
ω_{12}	10
ω_{13}	10
ω_{14}	10
ω_{15}	500
ω_{16}	200
ω_{17}	150

Tabela 13 – Penalizações e valores do PTHU

4.2 Detalhes de Implementação

Os algoritmos descritos neste trabalho foram implementados na linguagem C++ e os testes computacionais foram realizados em uma máquina com CPU Intel Core i5-7200U CPU @ 2.50GHz com 12,0 GB de memória RAM e sistema operacional Microsoft Windows 10.

As ofertas das disciplinas *Trabalho de Conclusão de Curso I*, *Trabalho de Conclusão de Curso II* e *Estágio em Informática* não foram consideradas para o problema, visto que são disciplinas que não necessitam serem consideradas no horário e não requerem espaço físico.

4.3 Análise dos Resultados

O algoritmo foi executado 10 vezes para cada instância (2013/2 e 2016/1), para cada valor de α (0, 0.15, 0.25, 0.5, 0.75, 1 e Aleatório) e para cada conjunto de movimentos da busca local (*Move/Swap* e *Lecture Move*). Ao final das execuções, todas as soluções obtidas apresentaram-se viáveis, ou seja, não violaram nenhuma restrição forte para o

problema. É importante notar que, apesar da execução ocorrer até Max_{Iter} , os tempos exibidos são os tempos em que foi alcançada a melhor solução.

Nas Tabelas 14 e 15, pode-se verificar que, mantendo $\alpha = 0.15$, mesmo valor usado por Moreira et al. (2016), para ambos os semestres, a solução com o movimento *Lecture Move* apresentou melhores resultados. Para 2013/2, a média dos resultados foi 18,37% melhor do que a solução com os movimentos *Move/Swap*, enquanto para o semestre 2016/1, a melhora foi de 20,08%.

A melhoria da solução com o *Lecture Move* em relação ao *Move/Swap* pode ser explicada pela eliminação da escolha aleatória de qual movimento seria realizado. Com a introdução do *Lecture Move*, primeiro são escolhidos os horários, se os dois horários escolhidos já possuem aulas alocadas, tenta-se realizar o *Swap*, mas se somente um horário já está alocado, tenta-se fazer o *Move*.

	Move/Swap	Lecture Move
Melhor Solução	406	306
Média	463,6	331,4
Tempo	267,012	255,710

Tabela 14 – Melhores Soluções, médias das soluções e tempo para 2013/2 com $\alpha = 0.15$

	Move/Swap	Lecture Move
Melhor Solução	278	208
Média	298,8	238,0
Tempo	231,525	301,577

Tabela 15 – Melhores Soluções, médias das soluções e tempo para 2016/1 com $\alpha = 0.15$

Na Tabela 16 são mostrados as melhores resultados, a média dos valores e a média de tempo utilizando o movimento de busca local *Lecture Move* e variando o valor de α . Pode-se observar que a melhor média de soluções para o semestre 2013/2 foi obtida com $\alpha = 0$, com a média dos resultados de 320,8, enquanto que para 2016/1 foi com $\alpha = 0.75$, com a média dos resultados de 233,4.

Para o semestre 2013/2, a melhor solução encontrada foi com $\alpha = 0$, isso significa que uma estratégia totalmente gulosa foi melhor aproveitada para essa instância. Já para o semestre 2016/1, com a melhor solução possuindo $\alpha = 0.75$, percebe-se que uma maior aleatoriedade foi mais benéfica para essa instância.

Na Tabela 17 pode-se verificar a melhor solução, os tempos médios de execução e a média da FO obtidas neste trabalho, bem como da solução proposta por Moreira et al. (2016) e o valor da FO da solução construída manualmente pelos coordenadores de curso.

Para o semestre de 2013/2 a solução manual obtida pelo DCOMP resultou em uma FO = 760, já para o trabalho de Moreira et al. (2016) a melhor solução encontrada foi FO

		0	0.15	0.25	0.5	0.75	1.0	Aleatório
2013/2 <i>Lecture Move</i>	Melhor Solução	268	306	324	306	296	302	282
	Média	320,8	331,4	351,8	347,8	347,0	350,0	337,8
	Tempo	183,860	255,710	261,245	231,892	363,247	197,625	336,832
2016/1 <i>Lecture Move</i>	Melhor Solução	214	208	200	220	198	214	222
	Média	247,4	238,0	234,4	242,8	233,4	243,8	234,8
	Tempo	342,737	301,577	215,537	271,353	185,639	261,776	285,848

Tabela 16 – Melhores soluções, média das soluções e tempo para o Movimento Lecture Move com variação de alfa

= 466. Assim é possível verificar que a solução apresentada por este trabalho alcançou uma melhora percentual de 57,79% em relação à solução manual 2013/2 e uma melhora de 20.98% quando comparada à solução de [Moreira et al. \(2016\)](#).

Já para o semestre de 2016/1, a solução manual obtida pelo DCOMP teve sua FO calculada em 2374. Assim, pode-se verificar que a solução obtida por este trabalho apresenta uma melhora de 90.17% comparada com a solução manual e de 21.63% quando comparada à solução de [Moreira et al. \(2016\)](#).

		Manual	Moreira et al. (2016)	GRASP+SA+LM
2013/2	Melhor Solução	760	406	268
	Média	-	463,6	320,8
	Tempo	-	267,012	206,482
2016/1	Melhor Solução	2374	278	198
	Média	-	297,8	233,4
	Tempo	-	231,525	185,639

Tabela 17 – Melhores soluções, média das soluções e tempo para o Movimento Lecture Move

5 Conclusões

Este trabalho teve como objetivo o estudo do impacto da aleatoriedade e de movimentos na meta-heurística GRASP com *Simulated Annealing* para a resolução do Problema de Tabela-Horários de Universidades, considerando o Departamento de Computação do Centro de Ciências Exatas, Naturais e da Saúde da Universidade Federal do Espírito Santo como caso de estudo.

Neste trabalho foi implementado o movimento *Lecture Move*, que é a combinação dos clássicos movimentos *Move* e *Swap*, sendo executado para diferentes fatores de aleatoriedade (α).

Os resultados obtidos foram comparados com outro trabalho (MOREIRA et al., 2016), bem como as soluções manuais elaboradas pelos coordenadores de curso do DCOMP-CCENS-UFES, apresentando soluções melhores para ambos os casos.

Portanto, os resultados obtidos por este trabalho mostraram-se ser mais eficazes que os existentes, podendo ser aplicado para a geração de Tabela-Horários em semestres futuros, e assim obter tabelas de horários aproximadamente 78,2% melhores, ou seja, com menos violações de restrições, do que as soluções atuais.

Como trabalhos futuros, sugere-se a implementação de novos movimentos para problemas de tabela-horário, como exemplo, o clássico *Cadeia de Kempe*, além de movimentos específicos para restrições fracas, assim como Müller (2009) fez para o ITC-2007. Além disso, outros trabalhos futuros poderiam realizar a implementação de outras meta-heurísticas como Busca Tabu e *Iterated Local Search*, que também podem usar o *SA* como busca local.

Todo o código fonte deste trabalho, bem como os resultados e instâncias utilizadas, se encontram disponíveis através do link <https://github.com/mmtulio/TCC2>.

Referências

- BURKE, E. K.; ELLIMAN, D.; WEARE, R. A genetic algorithm based university timetabling system. In: *Proceedings of the 2nd east-west international conference on computer technologies in education*. [S.l.: s.n.], 1994. v. 1, p. 35–40. Citado na página 15.
- BURKE, E. K.; NEWALL, J. P.; WEARE, R. F. A memetic algorithm for university exam timetabling. In: SPRINGER. *international conference on the practice and theory of automated timetabling*. [S.l.], 1995. p. 241–250. Citado na página 15.
- ČANGALOVIĆ, M.; SCHREUDER, J. A. Exact coloring algorithms for weighed graphs applied to timetabling problems with lectures of different lengths. *European Journal of Operational Research*, v. 51, p. 248–258, 1991. Citado na página 15.
- CARVALHO, A. S. et al. Simulated Annealing aplicado ao problema de programação de horários do CCA-UFES. *Simpósio de Pesquisa Operacional e Logística da Marinha - Publicação Online*, v. 2, n. 1, p. 341 – 352, 2016. ISSN 2175-6295. Disponível em: <www.proceedings.blucher.com.br/article-details/22705>. Citado 2 vezes nas páginas 16 e 24.
- CESCHIA, S.; GASPERO, L. D.; SCHAERF, A. Design, Engineering, and Experimental Analysis of a Simulated Annealing Approach to the Post-Enrolment Course Timetabling Problem. *Comput. Oper. Res.*, Elsevier Science Ltd., GBR, v. 39, n. 7, p. 1615–1624, jul. 2012. ISSN 0305-0548. Disponível em: <<https://doi.org/10.1016/j.cor.2011.09.014>>. Citado na página 16.
- ELMOHAMED, M. S.; CODDINGTON, P.; FOX, G. A comparison of annealing techniques for academic course scheduling. In: SPRINGER. *International Conference on the Practice and Theory of Automated Timetabling*. [S.l.], 1997. p. 92–112. Citado na página 16.
- FEO, T. A.; RESENDE, M. G. A probabilistic heuristic for a computationally difficult set covering problem. *Operations research letters*, Elsevier, v. 8, n. 2, p. 67–71, 1989. Citado 3 vezes nas páginas 9, 22 e 23.
- FEO, T. A.; RESENDE, M. G. Greedy randomized adaptive search procedures. *Journal of global optimization*, Springer, v. 6, n. 2, p. 109–133, 1995. Citado na página 23.
- FEO, T. A.; RESENDE, M. G. C.; SMITH, S. H. A greedy randomized adaptive search procedure for maximum independent set. *Operations Research*, INFORMS, v. 42, n. 5, p. 860–878, 1994. Citado na página 22.
- FERLAND, J. A.; ROY, S. Timetabling problem for university as assignment of activities to resources. *Computers & operations research*, Elsevier, v. 12, n. 2, p. 207–218, 1985. Citado na página 15.
- GOTLIEB, C. The construction of class-teacher time-tables. In: ASSOC COMPUTING MACHINERY 1515 BROADWAY, NEW YORK, NY 10036. *Communications of the ACM*. [S.l.], 1962. v. 5, n. 6, p. 312–313. Citado na página 15.

- KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by Simulated Annealing. *Science*, American association for the advancement of science, v. 220, n. 4598, p. 671–680, 1983. Citado na página 25.
- KOSTUCH, P. The university course timetabling problem with a three-phase approach. In: SPRINGER. *International Conference on the Practice and Theory of Automated Timetabling*. [S.l.], 2004. p. 109–125. Citado na página 15.
- LEWIS, R. A survey of metaheuristic-based techniques for university timetabling problems. *OR spectrum*, Springer, v. 30, n. 1, p. 167–190, 2008. Citado na página 15.
- LI, Y.; PARDALOS, P. M.; RESENDE, M. C. G. A greedy randomized adaptive search procedure for the quadratic assignment problem. *Quadratic Assignment and Related Problems, DIMACS Series on Discrete Mathematics and Theoretical Computer Science*, v. 16, p. 237–261, 1994. Citado na página 22.
- MACHADO, A. M.; BOERES, M. C. S. Uma proposta de formulação do problema de programação de tabela-horário de exames de toronto via coloração de grafos e sua resolução pelo algoritmo de busca tabu. *XLI SBPO-Pesquisa Operacional na Gestão do Conhecimento*, 2009. Citado na página 15.
- MARIANO, G. P. Resolução do problema de programação de horários de disciplinas do CCA-UFES utilizando a meta-heurística ALNS. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Universidade Federal do Espírito Santo, 2014. Citado 6 vezes nas páginas 13, 15, 16, 17, 21 e 24.
- MOREIRA, L. V. et al. Meta-heurística GRASP para o Problema de Tabela-horário de Disciplinas do Departamento de Computação do CCA-UFES. In: *Anais do XLVIII Simpósio Brasileiro de Pesquisa Operacional*. Rio de Janeiro, RJ, Brasil: SOBRAPO, 2016. p. 2171–2182. Citado 14 vezes nas páginas 13, 14, 15, 16, 17, 21, 22, 26, 27, 29, 30, 31, 32 e 33.
- MÜLLER, T. ITC2007 solver description: a hybrid approach. *Annals of Operations Research*, Springer, v. 172, n. 1, p. 429, 2009. Citado 3 vezes nas páginas 15, 26 e 33.
- NEUFELD, G.; TARTAR, J. Generalized graph colorations. *SIAM Journal on Applied Mathematics*, SIAM, v. 29, n. 1, p. 91–98, 1975. Citado na página 15.
- OSTERMANN, R.; WERRA, D. de. Some experiments with a timetabling system. *Operations-Research-Spektrum*, Springer, v. 3, n. 4, p. 199–204, 1982. Citado na página 15.
- RESENDE, M. G.; FEO, T. A. A GRASP for satisfiability. In: CITESEER. *Cliques, Coloring, and Satisfiability: The second DIMACS implementation challenge, Volume 26 of DIMACS Series on discrete mathematics and theoretical computer science*. [S.l.], 1996. p. 499–520. Citado na página 22.
- RESENDE, M. G.; RIBEIRO, C. C. A GRASP for graph planarization. *Networks: An International Journal*, Wiley Online Library, v. 29, n. 3, p. 173–189, 1997. Citado na página 22.
- RESENDE, M. G.; RIBEIRO, C. C. A GRASP with path-relinking for private virtual circuit routing. *Networks: An International Journal*, Wiley Online Library, v. 41, n. 2, p. 104–114, 2003. Citado na página 22.

- ROCHA, W. *Algoritmo GRASP para o problema de tabela-horário de universidades*. Dissertação (Mestrado) — Mestrado em Informática, Universidade Federal do Espírito Santo, 2013. Citado 5 vezes nas páginas 9, 15, 22, 24 e 26.
- SANTOS, H. G.; SOUZA, M. J. F. Programação de horários em instituições educacionais: formulações e algoritmos. *XXXIX SBPO-Simpósio Brasileiro de Pesquisa Operacional*, n. 1, p. 2827–2882, 2007. Citado na página 12.
- SCHAERF, A. A survey of automated timetabling. *Artificial intelligence review*, Springer, v. 13, n. 2, p. 87–127, 1999. Citado 3 vezes nas páginas 12, 15 e 26.
- SCHMIDT, G.; STRÖHLEIN, T. Timetable construction—an annotated bibliography. *The Computer Journal*, Oxford University Press, v. 23, n. 4, p. 307–316, 1980. Citado na página 15.
- SEGATTO, E. de A. et al. *Um Algoritmo GRASP com Cadeia de Kempe Aplicado ao Problema de Tabela-horário para Universidades*. Dissertação (Mestrado) — Mestrado em Informática, Universidade Federal do Espírito Santo, 2017. Citado na página 15.
- SOUZA, M. J. F. *Programação de horários em escolas: uma aproximação por metaheurísticas*. Tese (Doutorado) — Doutorado em Ciências em Engenharia de Sistemas e Computação - Universidade Federal do Rio de Janeiro, 2000. Citado na página 15.
- TRIPATHY, A. School timetabling—a case in large binary integer linear programming. *Management science*, INFORMS, v. 30, n. 12, p. 1473–1489, 1984. Citado na página 15.
- WOOD, D. C. A technique for colouring a graph applicable to large scale timetabling problems. *The Computer Journal*, Oxford University Press, v. 12, n. 4, p. 317–319, 1969. Citado na página 15.
- WREN, A. Scheduling, timetabling and rostering - a special relationship? In: *Selected Papers from the First International Conference on Practice and Theory of Automated Timetabling*. Berlin, Heidelberg: Springer-Verlag, 1995. p. 46–75. ISBN 3540617949. Citado na página 12.

APÊNDICE A – Lista de dados do CCENS-UFES

A seguir, são apresentadas as listas de todos os dados para os períodos de 2013/2 e 2016/1 do DCOMP.

A.1 Dados 2013/2

Id	Prédio	Tipo	Número	Capacidade
0	Prédio Central	Sala	9	40
1	Prédio Central	Sala	4	55
2	Prédio Novo	Sala	3	90
3	Prédio Novo	Sala	1	55
4	Prédio Novo	Sala	12	65
5	Prédio Novo	Sala	9	90
6	Prédio Antigo	Sala	3	70
7	Prédio Central	Sala	0	80
8	ChiChiu	Laboratório	3	40
9	ChiChiu	Laboratório	2	40
10	ChiChiu	Laboratório	1	40
11	Reuni	Laboratório	7	20
12	Reuni	Laboratório	5	20
13	Reuni	Laboratório	6	20

Tabela 18 – Lista de locais disponíveis

Id	Nome
0	Antonio Almeida De Barros Junior
1	Bruno Vilela Oliveira
2	Clayton Vieira Fraga Filho
3	Edmar Hell Kampke
4	Geraldo Regis Mauri
5	Helder De Amorim Mendes
6	Jacson Rodrigues Correia Da Silva
7	Juliana Pinheiro Campos
8	Larice Nogueira De Andrade
9	Paulo Roberto Nunes De Souza
10	Rodrigo Freitas Silva
11	Simone Dornelas Costa
12	Thiago Meireles Paixao
13	Valeria Alves Da Silva
14	Alexandre Rosa
15	Tharso
16	Bernado
17	Atila
18	Ronald
19	Maristela
20	Clovis
21	Paulo Henrique Souza
22	Eleonesio Strey
23	Aline De Menezes

Tabela 19 – Lista de professores

Id	Código	Nome	Nível
0	COM06842	Programação I	1
1	COM06850	Introducao a Ciencia da Computacao	0
2	COM06851	Matemática Discreta	1
3	COM10076	Arquitetura de Computadores	0
4	COM06992	Estrutura de Dados I	1
5	COM10078	Estrutura de Dados II	1
6	COM06853	Lógica Computacional	1
7	COM10080	Lógica Computacional II	1
8	COM10081	Metodologia de Pesquisa em Informática	1
9	COM10082	Programação II	1
10	COM10275	Engenharia de Requisitos de Software	1
11	COM10392	Linguagens de Programação	1
12	COM10393	Métodos de Otimização	1
13	COM10394	Redes de Computadores	0
14	COM10395	Teoria da Computação	0
15	COM10733	Gerência de Projeto de Software	1
16	ENG10791	Compiladores	1
17	ENG10792	Inteligência Artificial	0
18	ENG10793	Trabalho de Conclusão de Curso em Ciencia da Computacao I	0
19	COM11063	Trabalho de Conclusão de Curso em Ciencia da Computacao II	0
20	COM10396	Desenvolvimento de Sistemas Para Web	0
21	COM11014	Gerenciamento de Banco de Dados	0

22	COM05207	Informática	0
23	COM06039	Lógica e Técnica de Programação	0
24	COM06847	Introdução à Informática	0
25	COM06852	Introdução aos Sistemas de Informação	0
26	COM06996	Informática e Sociedade	0
27	COM10014	Computabilidade e Complexidade	1
28	COM10015	Engenharia de Software	1
29	COM10016	Sistema de Apoio à Decisão	0
30	COM10128	Algoritmos Numéricos	1
31	COM10129	Banco de Dados	1
32	COM10131	Otimização Linear	1
33	COM10132	Sistemas Operacionais	1
34	COM10399	Processamento Digital de Imagens	0
35	COM10507	Interface Humano-computador	0
36	COM10508	Projeto de Sistemas de Software	1
37	COM10607	Computação Forense	0
38	COM10612	Tópicos Especiais em Informática I	0
39	COM10616	Sistemas Distribuídos	1
40	COM11259	Sistemas de Software Livre	0
41	COM11260	Estágio em Informática	0
42	COM11261	Trabalho de Conclusão de Curso em Sistemas de Informação II	0
43	ENG11006	Comércio Eletrônico	0
44	ENG11007	Segurança e Auditoria de Sistemas	0
45	MPA06839	Cálculo A	1
46	ENG06849	Inglês Instrumental	0
47	MPA06840	Vetores e Geometria Analítica	1
48	MPA	Cálculo C	1
49	DQF	Fund Física II	1
50	ENG	Estatística	0
51	CFM10426	Administração e Economia	0
52	ENG06854	Português Instrumental	0
53	MPA06855	Álgebra Linear	1
54	VET10127	Libras	0

Tabela 20 – Lista de disciplinas.

Id	Descrição
0	Sala
1	Laboratório

Tabela 21 – Lista de tipos de salas

Id	Curso	Período	Turno Preferencial
0	Ciência da Computação	1	1
1	Ciência da Computação	3	1
2	Ciência da Computação	5	1
3	Ciência da Computação	7	1
4	Sistemas de Informação	1	2
5	Sistemas de Informação	3	2
6	Sistemas de Informação	5	2
7	Sistemas de Informação	7	2
8	Sistemas de Informação	9	2
9	Matemática	5	2
10	Agronomia	1	1
11	Engenharia Industrial Madeireira	3	1
12	Engenharia de Alimentos	3	1
13	Geologia	3	1
14	Ciências Biológicas	2	1
15	Nutrição	2	1
16	Engenharia Florestal	2	1
17	Matemática	1	2
18	Engenharia Química	1	1
19	Sistemas de Informação	1	2
20	Matemática	1	2
21	Ciência da Computação	1	1

Tabela 22 – Lista de turmas

Id	Horário Inicial	Horário Final	Duração
0	07h00	08h00	60
1	08h00	09h00	60
2	09h00	10h00	60
3	10h00	11h00	60
4	11h00	12h00	60
5	13h00	14h00	60
6	14h00	15h00	60
7	15h00	16h00	60
8	16h00	17h00	60
9	18h00	19h00	60
10	19h00	20h00	60
11	20h00	21h00	60
12	21h00	22h00	60
13	22h00	23h00	60

Tabela 23 – Lista de horários

Id	Disciplina	Nº de Turmas	Turmas[]	Vagas	Turno	Professor	Tipo de Sala	Ch
0	0	2	0 18	37	1	12	1	2
1	0	2	0 18	37	1	12	1	2
2	1	2	0 21	40	1	1	0	2
3	1	2	0 21	40	1	1	0	2
4	2	2	0 21	40	1	3	0	2
5	2	2	0 21	40	1	3	0	2
6	3	1	1	30	1	13	0	2

7	3	1	1	30	1	13	0	2
8	5	1	1	30	1	5	0	2
9	5	1	1	30	1	5	1	2
10	7	1	1	40	1	7	0	2
11	7	1	1	40	1	7	0	2
12	8	1	1	30	1	8	1	2
13	9	1	1	30	1	2	1	2
14	9	1	1	30	1	2	1	2
15	10	1	2	30	1	2	1	2
16	10	1	2	30	1	2	0	2
17	11	1	2	20	1	10	0	2
18	11	1	2	20	1	10	0	2
19	12	1	2	20	1	4	1	2
20	12	1	2	20	1	4	1	2
21	13	1	2	20	1	10	1	2
22	13	1	2	20	1	10	1	2
23	14	1	2	30	1	7	0	2
24	14	1	2	30	1	7	0	2
25	15	1	3	20	1	2	1	2
26	15	1	3	20	1	2	1	2
27	16	1	3	20	1	10	0	2
28	16	1	3	20	1	10	0	2
29	17	1	3	20	1	6	1	2
30	17	1	3	20	1	6	1	2
31	20	2	2 3	20	1	1	1	2
32	20	2	2 3	20	1	1	1	2
33	21	1	3	20	1	0	1	2
34	21	1	3	20	1	0	1	2
35	24	1	4	30	2	13	1	2
36	24	1	4	30	2	13	1	2
37	25	2	4 19	70	2	3	0	2
38	6	2	4 19	70	2	6	0	2
39	6	2	4 19	70	2	6	0	2
40	0	1	4	30	2	3	1	2
41	0	1	4	30	2	3	1	2
42	4	1	5	30	2	7	1	2
43	4	1	5	30	2	7	1	2
44	27	1	5	40	2	7	0	2
45	27	1	5	40	2	7	0	2
46	28	1	5	30	2	1	0	2
47	28	1	5	30	2	1	0	2
48	29	1	5	40	2	11	0	2
49	29	1	5	40	2	11	0	2
50	31	1	6	30	2	0	1	2
51	31	1	6	30	2	0	1	2

52	32	1	6	20	2	4	0	2
53	32	1	6	20	2	4	1	2
54	33	1	6	20	2	6	1	2
55	33	1	6	20	2	6	0	2
56	36	1	6	30	2	2	0	2
57	36	1	6	30	2	2	1	2
58	35	1	6	20	2	11	0	2
59	35	1	6	20	2	11	1	2
60	26	1	7	30	2	8	0	2
61	39	1	7	20	2	5	0	2
62	39	1	7	20	2	5	0	2
63	43	1	7	20	2	11	0	2
64	43	1	7	20	2	11	1	2
65	44	1	7	20	2	11	0	2
66	44	1	7	20	2	11	1	2
67	34	1	7	20	2	12	1	1
68	34	1	7	20	2	12	1	3
69	40	1	8	20	2	6	0	2
70	37	1	8	20	2	6	1	2
71	37	1	8	20	2	6	1	2
72	38	1	8	20	2	13	0	2
73	38	1	8	20	2	13	0	2
74	41	1	8	10	2	8	0	2
75	42	1	8	10	2	3	0	2
76	22	3	14 15 16	30	1	8	1	2
77	22	1	10	30	1	8	1	2
78	30	1	9	15	2	12	0	2
79	30	1	9	15	2	12	0	2
80	23	3	11 12 13	37	1	1	1	2
81	23	3	11 12 13	37	1	1	1	2
82	0	1	21	30	1	3	1	2
83	0	1	21	30	1	3	1	2
84	0	1	19	30	2	12	1	2
85	0	1	19	30	2	12	1	2
86	24	1	19	30	2	8	1	2
87	24	1	19	30	2	8	1	2
88	24	1	17	35	2	13	1	2
89	24	1	17	35	2	13	1	2
90	24	1	20	35	2	1	1	2
91	24	1	20	35	2	1	1	2
92	18	1	3	10	1	3	0	2
93	45	2	0 21	30	1	15	0	2
94	45	2	0 21	30	1	15	0	2
95	45	2	0 21	30	1	15	0	2
96	46	2	0 21	30	1	14	0	2

97	47	2	0 21	30	1	16	0	2
98	47	2	0 21	30	1	16	0	2
99	48	1	1	30	1	17	0	2
100	48	1	1	30	1	17	0	2
101	49	1	1	30	1	18	0	2
102	49	1	1	30	1	18	0	2
103	50	1	2	30	1	19	0	2
104	50	1	2	30	1	19	0	2
105	51	1	3	30	1	20	0	4
106	52	2	4 19	70	2	14	0	2
107	47	2	4 19	70	2	21	0	2
108	47	2	4 19	70	2	21	0	2
109	53	1	5	45	2	22	0	2
110	53	1	5	45	2	22	0	2
111	51	1	7	40	2	20	0	4
112	54	1	8	20	2	23	0	2
113	54	1	8	20	2	23	0	2

Tabela 24 – Lista de ofertas.

A.2 Dados 2016/1

Id	Prédio	Tipo	Número	Capacidade
0	Prédio Central	0	9	40
1	Prédio Central	0	4	55
2	Prédio Novo	0	1	55
3	Prédio Novo	0	12	65
4	Maracanã	0	0	80
5	ChiChiu	1	1	40
6	ChiChiu	1	2	40
7	ChiChiu	1	3	40
8	Reuni	1	6	20
9	Reuni	1	7	20

Tabela 25 – Lista de locais disponíveis

Id	Nome
0	Leandro
1	Bruno Vilela
2	André
3	Dayan
4	Edmar Hell
5	Geraldo Mauri
6	Helder Mendes
7	Jacson Rodrigues
8	Juliana Campos
9	Larice Nogueira
10	Marcelo
11	Rômulo Louzada
12	Rodrigo Freitas
13	Simone Dornelas
14	Valeria Alves

Tabela 26 – Lista de professores

Id	Código	Nome	Nível
0	COM06842	Programação I	1
1	COM06850	Introdução a Ciência da Computação	0
2	COM06851	Matemática Discreta	1
3	COM10076	Arquitetura de Computadores	0
4	COM06992	Estrutura de Dados I	1
5	COM10078	Estrutura de Dados II	1
6	COM06853	Lógica Computacional	1
7	COM10080	Lógica Computacional II	1
8	COM10081	Metodologia de Pesquisa em Informática	1
9	COM10082	Programação II	1
10	COM10275	Engenharia de Requisitos de Software	1
11	COM10392	Linguagens de Programação	1
12	COM10393	Métodos de Otimização	1
13	COM10394	Redes de Computadores	0

14	COM10395	Teoria da Computação	0
15	COM10733	Gerência de Projeto de Software	1
16	ENG10791	Compiladores	1
17	ENG10792	Inteligência Artificial	0
18	ENG10793	Trabalho de Conclusão de Curso em Ciência da Computação I	0
19	COM11063	Trabalho de Conclusão de Curso em Ciência da Computação II	0
20	COM10396	Desenvolvimento de Sistemas para WEB	0
21	COM11014	Gerenciamento de Banco de Dados	0
22	COM05207	Informática	0
23	COM06039	Lógica e Técnica de Programação	0
24	COM06847	Introdução a informática	0
25	COM06852	Introdução aos Sistemas de Informação	0
26	COM06996	Informática e Sociedade	0
27	COM10014	Computabilidade e Complexidade	0
28	COM10015	Engenharia de Software	1
29	COM10016	Sistema de Apoio a Decisão	0
30	COM10128	Algoritmos Numéricos	1
31	COM10129	Banco de Dados	1
32	COM10131	Otimização Linear	1
33	COM10132	Sistemas Operacionais	1
34	COM10399	Processamento Digital de Imagens	0
35	COM10507	Interface Humano-Computador	0
36	COM10508	Projeto de Sistemas de Software	1
37	COM10607	Computação Forense	0
38	COM10612	Tópicos Especiais em Informática I	0
39	COM10616	Sistemas Distribuídos	1
40	COM11259	Sistemas de Software Livre	0
41	COM11260	Estágio em Informática	0
42	COM11261	Trabalho de Conclusão de Curso em Sistemas de Informação II	0
43	ENG11006	Comércio Eletrônico	0
44	ENG11007	Segurança e Auditoria de Sistemas	0
55	COM06999	Circuitos Digitais	0
57	COM10130	Linguagens Formais e Autômatos	1
58	COM10133	Teoria dos Grafos	1
59	COM10602	Análise e Projeto de Algoritmos	1
60	COM10604	Computação Gráfica	0
61	COM10603	Direito e Legislação	0
62	COM11013	Tópicos Especiais em Programação	0
64	COM11608	Tópicos Especiais em Inteligência Artificial	0
65	COM11013	Tópicos Especiais em Otimização	0
66	COM06985	Teoria Geral de Sistemas	0
67	COM06984	Fundamentos de Programação Web	0
68	COM10608	Gerenciamento e Administração de Redes	0
69	COM11211	Gestão da Qualidade de Software	1
70	ENG10792	Segurança em Redes	0

71	COM11212	Trabalho de Conclusão de Curso I (SI)	0
----	----------	---------------------------------------	---

Tabela 27 – Lista de disciplinas.

Id	Descrição
0	Sala
1	Lab

Tabela 28 – Lista de tipos de salas

Id	Curso	Período	Turno Preferencial
0	Ciência da Computação	2	1
1	Ciência da Computação	3	1
2	Ciência da Computação	6	1
3	Ciência da Computação	8	1
4	Sistemas de Informação	2	2
5	Sistemas de Informação	4	2
6	Sistemas de Informação	6	2
7	Sistemas de Informação	8	2
8	Matemática	1	2
9	Matemática	5	2
10	Agronomia	1	1
11	Engenharia de Alimentos	3	1
12	Geologia	7	1
13	Engenharia Florestal	1	1
14	Sistemas de Informação	1	2

Tabela 29 – Lista de turmas

Id	Horário Inicial	Horário Final	Duração
0	07h00	08h00	60
1	08h00	09h00	60
2	09h00	10h00	60
3	10h00	11h00	60
4	11h00	12h00	60
5	13h00	14h00	60
6	14h00	15h00	60
7	15h00	16h00	60
8	16h00	17h00	60
9	18h00	19h00	60
10	19h00	20h00	60
11	20h00	21h00	60
12	21h00	22h00	60
13	22h00	23h00	60

Tabela 30 – Lista de horários

Id	Disciplina	Nº de Turmas	Turmas[]	Vagas	Turno	Professor	Tipo de Sala	Ch
0	55	1	0	30	1	15	0	2

1	55	1	0	30	1	12	0	2
2	4	1	0	40	1	11	1	2
3	4	1	0	40	1	11	1	2
4	26	1	0	30	1	10	0	2
5	6	1	0	30	1	9	0	2
6	6	1	0	30	1	9	0	2
7	30	1	1	30	1	23	0	2
8	30	1	1	30	1	23	0	2
9	31	1	1	30	1	0	0	2
10	31	1	1	30	1	0	0	2
11	28	1	1	30	1	2	0	2
12	28	1	1	30	1	2	0	2
13	57	1	1	30	1	9	1	2
14	57	1	1	30	1	9	1	2
15	32	1	1	30	1	6	0	2
16	32	1	1	30	1	6	0	2
17	33	1	1	20	1	5	0	2
18	33	1	1	20	1	5	0	2
19	58	1	1	30	1	4	0	2
20	58	1	1	30	1	4	0	2
21	59	1	2	30	1	13	0	2
22	59	1	2	30	1	13	0	2
23	60	1	2	30	1	5	1	2
24	60	1	2	30	1	5	1	2
25	61	1	2	30	1	10	0	2
26	35	1	2	20	1	14	0	2
27	35	1	2	20	1	14	1	2
28	36	1	2	30	1	2	1	2
29	36	1	2	30	1	2	1	2
30	39	1	2	20	1	5	1	2
31	39	1	2	20	1	5	1	2
32	62	1	2	30	1	8	1	2
33	62	1	2	30	1	8	1	2
34	64	1	3	30	1	8	1	2
35	64	1	3	30	1	8	1	2
36	65	1	3	30	1	3	1	2
37	65	1	3	30	1	3	1	2
38	67	1	4	40	2	1	0	2
39	67	1	4	40	2	1	0	2
40	2	1	4	40	2	4	0	2
41	2	1	4	40	2	4	0	2
42	66	1	4	40	2	14	0	2
43	66	1	4	40	2	14	0	2
44	3	1	5	50	2	15	0	2
45	3	1	5	50	2	15	0	2

46	10	1	5	40	2	2	1	2
47	10	1	5	40	2	2	1	2
48	5	1	5	30	2	11	1	2
49	5	1	5	30	2	11	1	2
50	9	1	5	30	2	1	1	2
51	9	1	5	30	2	1	1	2
52	61	1	6	40	2	10	0	2
53	15	1	6	20	2	2	1	2
54	15	1	6	20	2	2	1	2
55	21	1	6	20	2	0	1	2
56	21	1	6	20	2	0	1	2
57	8	1	6	40	2	10	0	2
58	12	1	6	20	2	6	1	2
59	12	1	6	20	2	6	1	2
60	13	1	6	20	2	13	1	2
61	13	1	6	20	2	13	1	2
62	20	1	7	20	2	8	1	2
63	20	1	7	20	2	8	1	2
64	68	1	7	20	2	7	1	2
65	68	1	7	20	2	7	1	2
66	69	1	7	20	2	2	1	2
67	69	1	7	20	2	2	1	2
68	70	1	7	20	2	7	1	2
69	70	1	7	20	2	7	1	2
70	22	1	9	40	1	12	1	3
71	22	1	12	40	1	12	1	3
72	0	1	8	28	2	12	1	2
73	0	1	8	28	2	12	1	2
74	0	1	13	30	2	12	1	2
75	0	1	13	30	2	12	1	2
76	23	1	10	40	1	0	1	2
77	23	1	10	40	1	0	1	2
78	23	1	11	40	1	0	1	2
79	23	1	11	40	1	0	1	2
80	56	1	0	30	1	19	0	2
81	56	1	0	30	1	19	0	2
82	53	1	0	21	1	18	0	2
83	53	1	0	21	1	18	0	2
84	45	1	4	40	2	21	0	2
85	45	1	4	40	2	21	0	2
86	45	1	4	40	2	21	0	2
87	50	1	5	40	2	20	0	2
88	50	1	5	40	2	20	0	2
89	63	1	7	40	2	17	0	2
90	63	1	3	30	1	17	0	2

91	46	1	4	40	2	22	0	2
92	52	1	0	30	1	16	0	2

Tabela 31 – Lista de ofertas.

APÊNDICE B – Tabela de horários obtidas pelo GRASP para Salas do período 2013/2

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira	
07h00						
08h00	Lógica Computacional II					
09h00	Lógica Computacional II					
10h00	Arquitetura de Computadores				Introdução a Ciência da Computação	
11h00	Arquitetura de Computadores				Introdução a Ciência da Computação	
13h00					Teoria da Computação	
14h00					Teoria da Computação	
15h00	Compiladores					
16h00	Compiladores					
18h00						
19h00						
20h00						
21h00						
22h00						
Disciplina		Curso	Período	Professor	Capacidade	Número de Vagas
Lógica Computacional II		CC	3	Juliana Pinheiro Campos	40	40
Arquitetura de Computadores		CC	3	Valeria Alves Da Silva	40	30
Compiladores		CC	7	Rodrigo Freitas Silva	40	20
Introdução a Ciência da Computação		CC	1	Bruno Vilela Oliveira	40	40
Teoria da Computação		CC	5	Juliana Pinheiro Campos	40	30

Tabela 32 – Prédio Central - Sala 09

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00	Compiladores			Administração e Economia	
14h00	Compiladores			Administração e Economia	
15h00	Cálculo C		Cálculo C	Administração e Economia	
16h00	Cálculo C		Cálculo C	Administração e Economia	
18h00		Álgebra linear			
19h00		Álgebra linear			
20h00				Álgebra linear	
21h00				Álgebra linear	
22h00					
Disciplina	Curso	Período	Professor	Capacidade	Número de Vagas
Compiladores	CC	7	Rodrigo Freitas Silva	55	20
Cálculo C	CC	3	Átila	55	30
Álgebra linear	SI	3	Eleonésio Strey	55	45
Administração e Economia	CC	7	Clóvis	55	30

Tabela 33 – Prédio Central - Sala 04

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00		Linguagens de Programação			
14h00		Linguagens de Programação			
15h00					
16h00					
18h00					
19h00					
20h00					
21h00					
22h00					
Disciplina	Curso	Período	Professor	Capacidade	Número de Vagas
Linguagens de Programação	CC	5	Rodrigo Freitas Silva	90	20

Tabela 34 – Prédio Novo - Sala 03

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira	
07h00						
08h00	Estatística		Estatística			
09h00	Estatística		Estatística			
10h00	Teoria da Computação		Fundamentos de Física III			
11h00	Teoria da Computação		Fundamentos de Física III			
13h00		Lógica Computacional II	Arquitetura de Computadores	Engenharia de Requisitos de Software		
14h00		Lógica Computacional II	Arquitetura de Computadores	Engenharia de Requisitos de Software		
15h00	Inglês Instrumental				Fundamentos de Física III	
16h00	Inglês Instrumental				Fundamentos de Física III	
18h00	Sistemas Distribuídos		Engenharia de Software	Administração e Economia		
19h00	Sistemas Distribuídos		Engenharia de Software	Administração e Economia		
20h00	Sistema de Apoio a Decisão		Sistema de Apoio a Decisão	Administração e Economia	Sistemas Operacionais	
21h00	Sistema de Apoio a Decisão		Sistema de Apoio a Decisão	Administração e Economia	Sistemas Operacionais	
22h00						
Disciplina		Curso	Período	Professor	Capacidade	Número de Vagas
Arquitetura de Computadores		CC	3	Valeria Alves Da Silva	55	30
Lógica Computacional II		CC	3	Juliana Pinheiro Campos	55	40
Engenharia de Requisitos de Software		CC	5	Clayton Vieira Fraga Filho	55	30
Teoria da Computação		CC	5	Juliana Pinheiro Campos	55	30
Engenharia de Software		SI	3	Bruno Vilela Oliveira	55	30
Sistema de Apoio a Decisão		SI	3	Simone Dornelas Costa	55	40
Sistemas Operacionais		SI	5	Jacson Rodrigues Correia Da Silva	55	20
Sistemas Distribuídos		SI	7	Helder De Amorim Mendes	55	20
Inglês Instrumental		CC	1	Alexandre Rosa	55	30
Fundamentos de Física III		CC	3	Ronald	55	30
Estatística		CC	5	Maristela	55	30
Administração e Economia		SI	7	Clóvis	55	40

Tabela 35 – Prédio Novo - Sala 01

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira	
07h00						
08h00						
09h00						
10h00						
11h00						
13h00						
14h00						
15h00						
16h00						
18h00	Computabilidade e Complexidade	Tópicos Especiais em Informática I		Tópicos Especiais em Informática I	Segurança e Auditoria de Sistemas	
19h00	Computabilidade e Complexidade	Tópicos Especiais em Informática I		Tópicos Especiais em Informática I	Segurança e Auditoria de Sistemas	
20h00	Otimização Linear	Comércio Eletrônico	Sistemas de Software Livre			
21h00	Otimização Linear	Comércio Eletrônico	Sistemas de Software Livre			
22h00						
Disciplina		Curso	Período	Professor	Capacidade	Número de Vagas
Computabilidade e Complexidade		SI	3	Juliana Pinheiro Campos	65	40
Otimização Linear		SI	5	Geraldo Regis Mauri	65	20
Comércio Eletrônico		SI	7	Simone Dornelas Costa	65	20
Segurança e Auditoria de Sistemas		SI	7	Simone Dornelas Costa	65	20
Sistemas de Software Livre		SI	9	Jacson Rodrigues Correia Da Silva	65	20
Tópicos Especiais em Informática I		SI	9	Valeria Alves Da Silva	65	20

Tabela 36 – Prédio Novo - Sala 12

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira	
07h00						
08h00					Matemática Discreta	
09h00					Matemática Discreta	
10h00						
11h00						
13h00	Cálculo A	Vetores e Geometria Analítica	Cálculo A	Vetores e Geometria Analítica	Cálculo A	
14h00	Cálculo A	Vetores e Geometria Analítica	Cálculo A	Vetores e Geometria Analítica	Cálculo A	
15h00				Introdução a Ciência da Computação		
16h00				Introdução a Ciência da Computação		
18h00						
19h00						
20h00						
21h00						
22h00						
Disciplina		Curso	Período	Professor	Capacidade	Número de Vagas
Introdução a Ciência da Computação		CC	1	Bruno Vilela Oliveira	90	40
Matemática Discreta		CC	1	Edmar Hell Kampke	90	40
Cálculo A		CC	1	Tharso	90	30
Vetores e Geometria Analítica		CC	1	Bernado	90	30

Tabela 37 – Prédio Novo - Sala 09

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira	
07h00						
08h00						
09h00						
10h00						
11h00						
13h00						
14h00						
15h00						
16h00						
18h00	Algoritmos Numéricos	Sistemas Distribuídos	Libras	Computabilidade e Complexidade	Libras	
19h00	Algoritmos Numéricos	Sistemas Distribuídos	Libras	Computabilidade e Complexidade	Libras	
20h00			Informática e Sociedade			
21h00			Informática e Sociedade			
22h00						
Disciplina		Curso	Período	Professor	Capacidade	Número de Vagas
Computabilidade e Complexidade		SI	3	Juliana Pinheiro Campos	70	40
Informática e Sociedade		SI	7	Larice Nogueira De Andrade	70	30
Sistemas Distribuídos		SI	7	Helder De Amorim Mendes	70	20
Algoritmos Numéricos		MA	5	Thiago Meireles Paixao	70	15
Libras		SI	9	Aline De Menezes	70	20

Tabela 38 – Prédio Antigo - Sala 03

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira	
07h00						
08h00						
09h00						
10h00						
11h00						
13h00						
14h00						
15h00						
16h00						
18h00	Português Instrumental	Vetores e Geometria Analítica	Lógica Computacional	Vetores e Geometria Analítica	Lógica Computacional	
19h00	Português Instrumental	Vetores e Geometria Analítica	Lógica Computacional	Vetores e Geometria Analítica	Lógica Computacional	
20h00	Introdução aos Sistemas de Informação	Projeto de Sistemas de Software		Interface Humano-Computador		
21h00	Introdução aos Sistemas de Informação	Projeto de Sistemas de Software		Interface Humano-Computador		
22h00						
Disciplina		Curso	Período	Professor	Capacidade	Número de Vagas
Introdução aos Sistemas de Informação		SI	1	Edmar Hell Kampke	80	70
Lógica Computacional		SI	1	Jacson Rodrigues Correia Da Silva	80	70
Projeto de Sistemas de Software		SI	5	Clayton Vieira Fraga Filho	80	30
Interface Humano-Computador		SI	5	Simone Dornelas Costa	80	20
Português Instrumental		SI	1	Alexandre Rosa	80	70
Vetores e Geometria Analítica		SI	1	Paulo Henrique Souza	80	70

Tabela 39 – Prédio Central - Sala 0

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00			Gerência de Projeto de Software		
11h00			Gerência de Projeto de Software		
13h00	Metodologia de Pesquisa em Informática		Inteligência Artificial		Programação I
14h00	Metodologia de Pesquisa em Informática		Inteligência Artificial		Programação I
15h00	Desenvolvimento de Sistemas para WEB		Gerenciamento de Banco de Dados		
16h00	Desenvolvimento de Sistemas para WEB		Gerenciamento de Banco de Dados		
18h00	Introdução a Informática			Programação I	Estrutura de Dados I
19h00	Introdução a Informática			Programação I	Estrutura de Dados I
20h00			Introdução a Informática		
21h00			Introdução a Informática		
22h00					
Disciplina	Curso	Período	Professor	Capacidade	Número de Vagas
Metodologia de Pesquisa em Informática	CC	3	Larice Nogueira De Andrade	40	30
Gerência de Projeto de Software	CC	7	Clayton Vieira Fraga Filho	40	20
Inteligência Artificial	CC	7	Jacson Rodrigues Correia Da Silva	40	20
Desenvolvimento de Sistemas para WEB	CC	7	Bruno Vilela Oliveira	40	20
Gerenciamento de Banco de Dados	CC	7	Antonio Almeida De Barros Junior	40	20
Programação I	SI	1	Thiago Meireles Paixao	40	30
Introdução a Informática	MA	1	Bruno Vilela Oliveira	40	35
Estrutura de Dados I	SI	3	Juliana Pinheiro Campos	40	30

Tabela 40 – ChiChiu - Laboratório 03

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira	
07h00						
08h00		Engenharia de Requisitos de Software				
09h00		Engenharia de Requisitos de Software				
10h00		Programação I				
11h00		Programação I				
13h00	Lógica e Técnicas de Programação	Gerência de Projeto de Software	Redes de Computadores			
14h00	Lógica e Técnicas de Programação	Gerência de Projeto de Software	Redes de Computadores			
15h00			Desenvolvimento de Sistemas para WEB			
16h00			Desenvolvimento de Sistemas para WEB			
18h00	Introdução a Informática	Introdução a Informática	Otimização Linear			
19h00	Introdução a Informática	Introdução a Informática	Otimização Linear			
20h00		Introdução a Informática		Computação Forense		
21h00		Introdução a Informática		Computação Forense		
22h00						
Disciplina		Curso	Período	Professor	Capacidade	Número de Vagas
Programação I		CC	1	Thiago Meireles Paixao	40	37
Engenharia de Requisitos de Software		CC	5	Clayton Vieira Fraga Filho	40	30
Redes de Computadores		CC	5	Rodrigo Freitas Silva	40	20
Gerência de Projeto de Software		CC	7	Clayton Vieira Fraga Filho	40	20
Desenvolvimento de Sistemas para WEB		CC	5	Bruno Vilela Oliveira	40	20
Introdução a Informática		SI	1	Valeria Alves Da Silva	40	30
Otimização Linear		SI	5	Geraldo Regis Mauri	40	20
Computação Forense		SI	9	Jacson Rodrigues Correia Da Silva	40	20
Lógica e Técnicas de Programação		EM	3	Bruno Vilela Oliveira	40	37
Introdução a Informática		SI	1	Larice Nogueira De Andrade	40	30

Tabela 41 – ChiChiu - Laboratório 02

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00				Programação II	
09h00				Programação II	
10h00		Redes de Computadores	Métodos de Otimização		
11h00		Redes de Computadores	Métodos de Otimização		
13h00	Métodos de Otimização	Programação I			Inteligência Artificial
14h00	Métodos de Otimização	Programação I			Inteligência Artificial
15h00	Informática	Lógica e Técnicas de Programação			
16h00	Informática	Lógica e Técnicas de Programação			
18h00	Interface Humano-Computador	Introdução a Informática	Programação I	Projeto de Sistemas de Software	Banco de Dados
19h00	Interface Humano-Computador	Introdução a Informática	Programação I	Projeto de Sistemas de Software	Banco de Dados
20h00				Introdução a Informática	Comércio Eletrônico
21h00				Introdução a Informática	Comércio Eletrônico
22h00					Processamento Digital de Imagens
Disciplina	Curso	Período	Professor	Capacidade	Número de Vagas
Programação II	CC	3	Clayton Vieira Fraga Filho	40	30
Métodos de Otimização	CC	5	Geraldo Regis Mauri	40	20
Redes de Computadores	CC	5	Rodrigo Freitas Silva	40	20
Inteligência Artificial	CC	7	Jacson Rodrigues Correia Da Silva	40	20
Introdução a Informática	SI	1	Valeria Alves Da Silva	40	30
Banco de Dados	SI	5	Antonio Almeida De Barros Junior	40	30
Projeto de Sistemas de Software	SI	5	Clayton Vieira Fraga Filho	40	30
Interface Humano-Computador	SI	5	Simone Dornelas Costa	40	20
Comércio Eletrônico	SI	7	Simone Dornelas Costa	40	20
Processamento Digital de Imagens	SI	7	Thiago Meireles Paixao	40	20
Informática	AG	1	Larice Nogueira De Andrade	40	30
Lógica e Técnicas de Programação	EM	3	Bruno Vilela Oliveira	40	37
Programação I	SI	1	Thiago Meireles Paixao	40	30
Introdução a Informática	MA	1	Bruno Vilela Oliveira	40	35

Tabela 42 – ChiChiu - Laboratório 01

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00					
14h00					
15h00					Gerenciamento de Banco de Dados
16h00					Gerenciamento de Banco de Dados
18h00					
19h00					
20h00					
21h00					
22h00					
Disciplina	Curso	Período	Professor	Capacidade	Número de Vagas
Gerenciamento de Banco de Dados	CC	7	Antonio Almeida De Barros Junior	20	20

Tabela 43 – Reuni - Laboratório 07

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00					
14h00					
15h00					
16h00					
18h00	Computação Forense	Sistemas Operacionais	Segurança e Auditoria de Sistemas		
19h00	Computação Forense	Sistemas Operacionais	Segurança e Auditoria de Sistemas		
20h00	Processamento Digital de Imagens				
21h00	Processamento Digital de Imagens				
22h00	Processamento Digital de Imagens				
Disciplina	Curso	Período	Professor	Capacidade	Número de Vagas
Computação Forense	SI	9	Jacson Rodrigues Correia Da Silva	20	20
Sistemas Operacionais	SI	5	Jacson Rodrigues Correia Da Silva	20	20
Segurança e Auditoria de Sistemas	SI	7	Simone Dornelas Costa	20	20
Processamento Digital de Imagens	SI	7	Thiago Meireles Paixão	20	20

Tabela 44 – Reuni - Laboratório 05

APÊNDICE C – Tabela de horários obtidas pelo GRASP para Turmas do período 2016/1

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00		Estrutura De Dados I			
09h00		Estrutura De Dados I			
10h00	Estrutura De Dados I	Informática e Sociedade			
11h00	Estrutura De Dados I	Informática e Sociedade			
13h00	Português Instrumental	Cálculo B	Lógica Computacional	Cálculo B	Lógica Computacional
14h00	Português Instrumental	Cálculo B	Lógica Computacional	Cálculo B	Lógica Computacional
15h00		Álgebra Linear	Circuitos Digitais	Álgebra Linear	Circuitos Digitais
16h00		Álgebra Linear	Circuitos Digitais	Álgebra Linear	Circuitos Digitais
18h00					
19h00					
20h00					
21h00					
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
COM06992	Estrutura De Dados I		CC-3	Marcelo	40
ENG06854	Português Instrumental		PC-4	Alexandre Rosa	30
COM06996	Informática e Sociedade		PN-1	Larice Nogueira De Andrade	30
MPA06979	Cálculo B		PC-4	Ana Clara	30
MPA06855	Álgebra Linear		PC-4	Gabriel	21
COM06853	Lógica Computacional		PC-9	Juliana Pinheiro Campos	30
COM06999	Circuitos Digitais		PC-9	Valeria Alves Da Silva	30

Tabela 45 – Ciência da Computação - 2º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00				Sistemas Operacionais	Sistemas Operacionais
09h00				Sistemas Operacionais	Sistemas Operacionais
10h00				Algoritmos Numéricos	Algoritmos Numéricos
11h00				Algoritmos Numéricos	Algoritmos Numéricos
13h00	Engenharia de Software	Teoria dos Grafos	Teoria dos Grafos	Otimização Linear	Banco de Dados
14h00	Engenharia de Software	Teoria dos Grafos	Teoria dos Grafos	Otimização Linear	Banco de Dados
15h00					
16h00					
18h00					
19h00					
20h00					
21h00					
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
COM10132	Sistemas Operacionais		PN-1/PC-9	Fabício	20
COM10128	Algoritmos Numéricos		PN-1/PC-9	Prof. Algoritmos Numéricos	30
COM10015	Engenharia de Software		CC-2	André	30
COM10133	Teoria dos Grafos		PN-1/PC-4	Edmar Hell Kampke	30
COM10131	Otimização Linear		CC-1	Geraldo Régis Mauri	30
COM10129	Banco de Dados		CC-2	Leandro	30

Tabela 46 – Ciência da Computação - 4º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00				Projeto de Sistemas de Software	
09h00				Projeto de Sistemas de Software	
10h00				Computação Gráfica	Sistemas Distribuídos
11h00				Computação Gráfica	Sistemas Distribuídos
13h00	Análise e Projeto de Algoritmos	Projeto de Sistemas de Software	Análise e Projeto de Algoritmos	Sistemas Distribuídos	Computação Gráfica
14h00	Análise e Projeto de Algoritmos	Projeto de Sistemas de Software	Análise e Projeto de Algoritmos	Sistemas Distribuídos	Computação Gráfica
15h00	Direito e Legislação	Interface Humano-Computador	Tópicos Especiais em Programação	Interface Humano-Computador	Tópicos Especiais em Programação
16h00	Direito e Legislação	Interface Humano-Computador	Tópicos Especiais em Programação	Interface Humano-Computador	Tópicos Especiais em Programação
18h00					
19h00					
20h00					
21h00					
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
COM10507	Interface Humano-Computador		RN-6	Simone Dornelas Costa	20
COM10508	Projeto de Sistemas de Software		CC-2/CC-3	André	30
COM10602	Análise e Projeto de Algoritmos		PN-1/PC-9	Rodrigo Freitas Silva	30
COM10603	Direito Legislação		PN-1	Larice Nogueira De Andrade	30
COM10604	Computação Gráfica		CC-1/CC-3	Fabício	30
COM10616	Sistemas Distribuídos		RN-7	Fabício	20
COM11013	Tópicos Especiais em Programação		CC-3	Jacson Rodrigues Correia Da Silva	30

Tabela 47 – Ciência da Computação - 6º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00		Tópicos Especiais em Otimização	Tópicos Especiais em Inteligência Artificial		
14h00		Tópicos Especiais em Otimização	Tópicos Especiais em Inteligência Artificial		
15h00	Empreendedorismo		Tópicos Especiais em Otimização	Tópicos Especiais em Inteligência Artificial	
16h00	Empreendedorismo		Tópicos Especiais em Otimização	Tópicos Especiais em Inteligência Artificial	
18h00					
19h00					
20h00					
21h00					
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
CFM11061	Empreendedorismo		PC-9	Wendel	30
COM11013	Tópicos Especiais em Otimização		CC-1/CC-2	Dayan	30
COM11608	Tópicos Especiais em Inteligência Artificial		CC-1/CC-3	Jacson Rodrigues Correia Da Silva	30

Tabela 48 – Ciência da Computação - 8º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00					
14h00					
15h00					
16h00					
18h00	Cálculo A	Matemática Discreta	Cálculo A	Matemática Discreta	Cálculo A
19h00	Cálculo A	Matemática Discreta	Cálculo A	Matemática Discreta	Cálculo A
20h00	Inglês Instrumental	Teoria Geral Sistemas	Fundamentos de Programação Web	Fundamentos de Programação Web	Teoria Geral Sistemas
21h00	Inglês Instrumental	Teoria Geral Sistemas	Fundamentos de Programação Web	Fundamentos de Programação Web	Teoria Geral Sistemas
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
COM06851	Matemática Discreta		MA-0/PN-1	Edmar Hell Kampke	40
COM06984	Fundamentos de Programação Web		CC-1/CC-3	Bruno Vilela Oliveira	40
COM06985	Teoria Geral Sistemas		MA-0	Simone Dornelas Costa	40
ENG06849	Inglês Instrumental		MA-0	Prof Inglês Instrumental	40
MPA06839	Cálculo A		MA-0	Prof Calculo A	40

Tabela 49 – Sistemas de Informação - 2º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00					
14h00					
15h00					
16h00					
18h00	Engenharia de Requisitos de Software	Estatística	Estrutura de Dados II	Estatística	Programação II
19h00	Engenharia de Requisitos de Software	Estatística	Estrutura de Dados II	Estatística	Programação II
20h00	Arquitetura de Computadores	Estrutura de Dados II	Arquitetura de Computadores	Engenharia de Requisitos de Software	
21h00	Arquitetura de Computadores	Estrutura de Dados II	Arquitetura de Computadores	Engenharia de Requisitos de Software	
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
COM10076	Arquitetura de Computadores		PN-1/PN-12	Valéria Alves Da Silva	50
COM10078	Estrutura de Dados II		CC-1/CC-3	Marcelo	30
COM10082	Programação II		CC-1	Bruno Vilela Oliveira	30
COM10275	Engenharia de Requisitos de Software		CC-2/CC-3	André	40
ENG05510	Estatística		PN-12	Prof Estatística Básica	40

Tabela 50 – Sistemas de Informação - 4º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00					
14h00					
15h00					
16h00					
18h00	Metodologia de Pesquisa em Informática	Métodos de Otimização	Redes de Computadores	Métodos de Otimização	Gerência de Projeto de Software
19h00	Metodologia de Pesquisa em Informática	Métodos de Otimização	Redes de Computadores	Métodos de Otimização	Gerência de Projeto de Software
20h00	Redes de Computadores	Direito Legislação		Gerenciamento de Banco de Dados	Gerenciamento de Banco de Dados
21h00	Redes de Computadores	Direito Legislação		Gerenciamento de Banco de Dados	Gerenciamento de Banco de Dados
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
COM10081	Metodologia de Pesquisa em Informática		PN-12	Larice Nogueira De Andrade	40
COM10393	Métodos de Otimização		CC-1/CC-3	Geraldo Régis Mauri	20
COM10394	Redes de Computadores		CC-1/RN-7	Rodrigo Freitas Silva	20
COM10603	Direito Legislação		PN-12	Larice Nogueira De Andrade	40
COM10733	Gerência de Projeto de Software		RN-6	André	20
COM11014	Gerenciamento de Banco de Dados		CC-2/RN-6	Leandro	20

Tabela 51 – Sistemas de Informação - 6º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00					
14h00					
15h00					
16h00					
18h00	Segurança em Redes	Gestão Qualidade de Software	Gestão Qualidade de Software	Empreendedorismo	Desenvolvimento de Sistemas para WEB
19h00	Segurança em Redes	Gestão Qualidade de Software	Gestão Qualidade de Software	Empreendedorismo	Desenvolvimento de Sistemas para WEB
20h00	Gerenciamento Administração Redes	Gerenciamento Administração Redes	Segurança em Redes	Desenvolvimento de Sistemas para WEB	
21h00	Gerenciamento Administração Redes	Gerenciamento Administração Redes	Segurança em Redes	Desenvolvimento de Sistemas para WEB	
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
CFM11061	Empreendedorismo		PN-1	Wendel	40
COM10396	Desenvolvimento de Sistemas para WEB		CC-2/RN-7	Jacson Rodrigues Correia Da Silva	20
COM10608	Gerenciamento Administração Redes		CC-3/RN-7	Helder De Amorim Mendes	20
COM11211	Gestão Qualidade de Software		RN-6	André	20
ENG10792	Segurança em Redes		CC-3/RN-7	Helder De Amorim Mendes	20

Tabela 52 – Sistemas de Informação - 8º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00					
14h00					
15h00					
16h00					
18h00		Programação I	Programação I		
19h00		Programação I	Programação I		
20h00					
21h00					
22h00					
Código da disciplina		Disciplina	Sala	Professor	Vagas
COM06842		Programação I	CC-1/CC-3	Rômulo	28

Tabela 53 – Matemática - 1º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					Informática
08h00					Informática
09h00					Informática
10h00					
11h00					
13h00					
14h00					
15h00					
16h00					
18h00					
19h00					
20h00					
21h00					
22h00					
Código da disciplina		Disciplina	Sala	Professor	Vagas
COM05207		Informática	CC-3	Rômulo	40

Tabela 54 – Agronomia - 1º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00			Lógica e Técnicas de Programação	Lógica e Técnicas de Programação	
14h00			Lógica e Técnicas de Programação	Lógica e Técnicas de Programação	
15h00					
16h00					
18h00					
19h00					
20h00					
21h00					
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
COM06039	Lógica e Técnicas de Programação		CC-2/CC-3	Leandro	40

Tabela 55 – Engenharia de Alimentos - 3º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00					
14h00			Lógica e Técnicas de Programação	Lógica e Técnicas de Programação	
15h00			Lógica e Técnicas de Programação	Lógica e Técnicas de Programação	
16h00					
18h00					
19h00					
20h00					
21h00					
22h00					
Código da disciplina	Disciplina		Sala	Professor	Vagas
COM06039	Lógica e Técnicas de Programação		CC-1/CC-2	Leandro	40

Tabela 56 – Geologia - 7º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00			Informática		
08h00			Informática		
09h00			Informática		
10h00					
11h00					
13h00					
14h00					
15h00					
16h00					
18h00					
19h00					
20h00					
21h00					
22h00					
Código da disciplina		Disciplina	Sala	Professor	Vagas
COM05207		Informática	CC-1	Rômulo	40

Tabela 57 – Engenharia Florestal - 1º Período

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07h00					
08h00					
09h00					
10h00					
11h00					
13h00					
14h00					
15h00					
16h00					
18h00				Programação I	Programação I
19h00				Programação I	Programação I
20h00					
21h00					
22h00					
Código da disciplina		Disciplina	Sala	Professor	Vagas
COM06842		Programação I	CC-2/CC-3	Rômulo	30

Tabela 58 – Sistemas de Informação - 1º Período