UNIVERSIDADE FEDERAL DO PAMPA

MILENA LIMA PEREIRA

TEORIA DOS GRAFOS E O PROBLEMA DE FLUXO EM REDES: APLICAÇÃO AO SALÃO INTERNACIONAL DE ENSINO, PESQUISA E EXTENSÃO (SIEPE) REALIZADO NA UNIVERSIDADE FEDERAL DO PAMPA

Bagé

2019

MILENA LIMA PEREIRA

TEORIA DOS GRAFOS E O PROBLEMA DE FLUXO EM REDES: APLICAÇÃO AO SALÃO INTERNACIONAL DE ENSINO, PESQUISA E EXTENSÃO (SIEPE) REALIZADO NA UNIVERSIDADE FEDERAL DO PAMPA

Trabalho de Conclusão de Curso apresentado ao Curso de Licenciatura em Matemática da Universidade Federal do Pampa, como requisito parcial para obtenção do título de Licenciada em Matemática.

Orientadora: Profa. Dra. Elizangela Dias Pereira.

Bagé

2019

Ficha catalográfica elaborada automaticamente com os dados fornecidos pela autora através do Módulo de Biblioteca do Sistema GURI (Gestão Unificada de Recursos Institucionais).

P436t Pereira, Milena Lima
Teoria do grafos e o problema de fluxos em redes: aplicação ao Salão Internacional de Ensino, Pesquisa e Extensão (SIEPE) realizado na Universidade Federal do Pampa / Milena Lima Pereira.
60 p.

Trabalho de Conclusão de Curso(Graduação)-- Universidade Federal do Pampa, MATEMÁTICA, 2019.
"Orientação: Elizangela Dias Pereira".

1. Grafos. 2. Fluxo em Redes. 3. Árvore Geradora Mínima. I. Título.

MILENA LIMA PEREIRA

TEORIA DOS GRAFOS E O PROBLEMA DE FLUXO EM REDES: APLICAÇÃO AO SALÃO INTERNACIONAL DE ENSINO, PESQUISA E EXTENSÃO (SIEPE) REALIZADO NA UNIVERSIDADE FEDERAL DO PAMPA

Trabalho de Conclusão de Curso apresentado ao Curso de Matemática da Universidade Federal do Pampa, como requisito parcial para obtenção do título de Licenciada em Matemática.

Trabalho de Conclusão de Curso defendido e aprovado em: 04/12/2019.

Banca examinadora:

Elizangela Dias Pereira

Profa. Dra. Elizangela Dias Pereira
Orientadora
UNIPAMPA

Anderson Bihain

Prof. Dr. Anderson Luís Jeske Bihain
UNIPAMPA

Sandra Dutra Piovesan

Profa. Dra. Sandra Dutra Piovesan
UNIPAMPA

Dedico este trabalho aos meus pais, pelo apoio incondicional e aos meus irmãos pela paciência e fé em mim.

“As grandes ideias surgem da observação dos pequenos detalhes”.

Augusto Cury

RESUMO

Este trabalho tem como objetivo o estudo e aplicação da Teoria dos Grafos, ramo da matemática discreta. A metodologia apresentada neste trabalho permite o aprofundamento da pesquisa qualitativa através de estudo de caso, a qual realiza análises baseadas em modelagem matemática, com o objetivo de mapear problemas a serem solucionados de forma otimizada. Assim, o caso definido corresponde em analisar e determinar qual dos dez campi UNIPAMPA que ao sediar o evento SIEPE (Salão Internacional de Ensino, Pesquisa e Extensão) resultaria em um menor custo para a instituição em relação a logística. Para realizar o estudo foram considerados os seguintes fatores: infraestrutura dos campi UNIPAMPA, o número total de alunos de cada campus, o número de inscritos no evento a partir da 7º edição, a frota de transportes disponível da UNIPAMPA e as distâncias entre todos os campi. Desta forma, tendo o conhecimento de Problema de Fluxo de Redes foi possível observar que através do método de Árvores Geradoras Mínimas foi viável obter uma solução adequada para o problema. Para esta solução foi realizada a implementação dos Algoritmos de Prim e Kruskal no qual pode-se obter dois dos campi UNIPAMPA que melhor se adequaram aos critérios propostos, sendo possível encontrar o trajeto mais apropriado de acordo com as frotas para o deslocamento dos alunos ao evento.

Palavras-Chave: Grafos. Fluxo em redes. Árvore geradora mínima.

ABSTRACT

The aim of this work is the study and application of the Graph Theory, an area of discrete mathematics. The methodology presented in this work permits the further development of the qualitative research through a case study, which performs analyses based on mathematical modeling, with the purpose of mapping problems to be solved in an optimized manner. Therefore, the defined case corresponds to analyze and determine which of the ten UNIPAMPA campuses can host the event SIEPE (International Fair of Teaching Institutions of Pampa) would result in a lower cost to the institution in relation to logistics. To perform the study, the following factors were considered: the infrastructure of the campuses of UNIPAMPA, the total number of students in each campus, the number of the registered people in the event from the 7th edition, the transport fleet of UNIPAMPA and the distances between all the campuses. Thus, with the knowledge of Network Flow Problem it was possible to observe that through the Minimum Generating Trees method it was feasible to obtain an appropriate solution to the problem. For this solution, it was realized analysis and implementation of Prim's and Kruskal's algorithms in which it was possible to obtain two of the UNIPAMPA campuses which best fit the proposed standards, being possible to find the shortest route for the movement of the students to the event.

Keywords: Graphs. Network flows. Minimum spanning tree.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

Dia Diagram Editor – Programa para criação de diagramas e grafos

GCC – GNU Compiler Collection

Gephi – Software para gráficos e redes

Graphpar – Software para particionar grafos

MST – Árvore Geradora Mínima

PMST – Problema da Árvore Geradora Mínima Probabilística

PostgreSQL – Sistema gerenciador de banco de dados

LISTA DE SIGLAS

CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico

COPEL – Companhia Paranaense de Energia Elétrica

e-SIC – Sistema eletrônico do serviço de informação ao cidadão

FAPERGS – Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul

SIEPE – Salão Internacional de Ensino, Pesquisa e Extensão

UNIPAMPA – Universidade Federal do Pampa

SUMÁRIO

1. INTRODUÇÃO

A teoria dos Grafos é um ramo da matemática discreta que compreende o estudo dos grafos, o qual é utilizado em aplicações do mundo moderno. De acordo com Mota [c.a 2013], a teoria dos Grafos surgiu no ano de 1736 através dos estudos do matemático suíço Leonhard Euler com o desafio das Pontes de Königsberg, localizada na cidade de Königsberg, atual Kaliningrado. Este estudo consistia em uma cidade cortada pelo rio Pregel que dificultava o transporte de mercadorias e pessoas através de barcos. A partir disto, foram construídas sete pontes que auxiliariam o deslocamento entre as ilhas e as duas margens. Logo, começaram a existir perguntas se era possível sair de casa, passar por todas as pontes apenas uma vez e voltar para casa em segurança. Esse desafio, portanto, culminou no desenvolvimento teórico e prático de Grafos.

Neste trabalho, foi verificado a localização do campus UNIPAMPA com menor custo logístico para a realização do evento SIEPE, informando o melhor caminho para o deslocamento dos transportes. Os fatores levados em consideração foram escolhidos para que houvesse uma análise mais detalhada sobre cada campus, assim como o número de alunos dos campi e número de inscritos no SIEPE, no qual minimiza o número de transportes ao evento.

Dentre as opções levadas em consideração para a pesquisa, foram realizados estudos para que a modelagem matemática representasse devidamente o modelo para o problema estudado. Por isso, visto que a modelagem por Caminhos Mínimos não apresentou uma única solução, devido a existência de um vértice de origem e não de chegada como proposto no modelo, foi recorrido à modelagem por Árvore Geradora Mínima, também parte da Teoria dos Grafos. Desta forma, foram estudados novos algoritmos no qual resultaram em uma solução mais adequada ao problema, que poderá ser aplicada na escolha da sede para o evento SIEPE da Universidade Federal do Pampa.

A metodologia utilizada foi fundamentada em Pesquisa Qualitativa com estudo de caso, a qual explora suas características gerando possibilidades para resolver o problema, encontrando a fonte do teste e definindo a ferramenta mais adequada para solucionar o mesmo. Este trabalho conta com uma revisão bibliográfica sobre a Teoria dos Grafos, o problema clássico de Fluxos de Redes e a aplicação dos Algoritmos no problema real.

1.1 Justificativa

Através do estudo de Teoria dos Grafos e suas ramificações, houve a necessidade de uma aplicação real do contexto estudado para melhor compreensão do conteúdo. A partir disso, ocorreu a escolha do problema, no qual envolvia aplicar a Teoria dos Grafos e seus conceitos na localização do campus UNIPAMPA para se tornar sede do evento SIEPE. Assim foi realizada a construção do grafo que representa o problema, com seus determinados vértices (cidades) e arestas (quilômetros entre um campus e outro), levando em consideração apenas a quilometragem entre as cidades. No início deste estudo, considerou-se que o problema fosse solucionado com os Algoritmos de Dijkstra e Floyd, porém ocorreu uma falha, pois os dois algoritmos exigem um vértice de saída que levaria a outros vértices, e o problema proposto envolve que todos os vértices cheguem apenas em um deles, no caso, a sede do evento.

Logo, a modelagem matemática foi analisada novamente e foi refeita para uma melhor solução, assim levando em consideração além das distâncias percorridas entre os campi, também, o número total de alunos, a infraestrutura física para contemplar o evento e informações em relação a gestão de frotas da UNIPAMPA. Portanto, a partir disso foi aprofundado o estudo na Teoria dos Grafos no qual encontrou-se outros dois algoritmos conhecidos como, Algoritmos de Prim e Kruskal, no qual geram uma Árvore Geradora Mínima que satisfazem o problema proposto.

1.2 Problema de Pesquisa

É possível selecionar um dos campi UNIPAMPA para o evento SIEPE, definindo o trajeto com menor custo logístico, selecionando a saída dos transportes a partir da disponibilidade de cada campus, utilizando um algoritmo que gera a árvore geradora mínima?

1.3 Objetivos

O presente trabalho tem como objetivo o estudo e aplicação da Teoria dos Grafos, para determinar qual campus da UNIPAMPA pode se tornar sede fixa para abrigar o evento SIEPE. O objetivo é encontrar o menor deslocamento acumulado entre todos os campi UNIPAMPA e consequentemente com o menor custo de transporte para universidade, enfatizando que será considerado a infraestrutura física dos campi, o número de alunos de cada campus e a frota de transportes da Universidade.

Como objetivos específicos pode-se citar:

- Revisão Bibliográfica da Teoria dos Grafos e o Problema de Fluxo de Redes;
- Modelo conceitual e matemático;
- Processo de busca de dados;
- Escrita do Trabalho de Conclusão de Curso;
- Construções dos Grafos;
- Estudo dos Algoritmos;
- Aplicação dos algoritmos no problema;
- Apresentação do método para solução do problema;

2. REFERÊNCIAL TEÓRICO

Na seção 2.1 será apresentado a história da Teoria dos Grafos e seu desenvolvimento, já na seção 2.2 encontram-se os conceitos fundamentais da Teoria dos Grafos, assim como os principais conceitos utilizados no problema de pesquisa.

2.1 Teoria dos Grafos

Segundo Mota [ca. 2013], a Teoria dos Grafos teve seu início no século XVII pelo matemático suíço Leonhard Euler (1707-1783) na cidade de Königsberg, através de um problema que, de acordo com os historiadores, os habitantes tinham como passatempo tentar encontrar um caminho de modo a passar por cada uma das pontes uma única vez e retornar ao ponto de partida, entretanto nenhuma tentativa obteve êxito, porém, foi apenas em 1730 que o problema conhecido como O Problema das Pontes de Königsberg, tornou-se um problema matemático, no qual foi comprovado ser impossível encontrar tal percurso.

Conforme Mota [ca. 2013], em 1735 Leonhard Euler, apresentou um artigo aos seus colegas, com a solução de um problema relacionado à geometria de posição: era o problema das Pontes de Königsberg. Neste documento ele também apresentou uma generalização do problema. Assim em 1736, Leonhard Euler escreveu um artigo – *Solutio Problematis ad Geometriam Situs Pertinentis* – em que resolve este problema em particular, e, além disso, atribui um método para a resolução de problemas semelhantes. Este artigo foi de muita importância para a Teoria de Grafos, dado que marcou o início do desenvolvimento da Matemática como um todo.

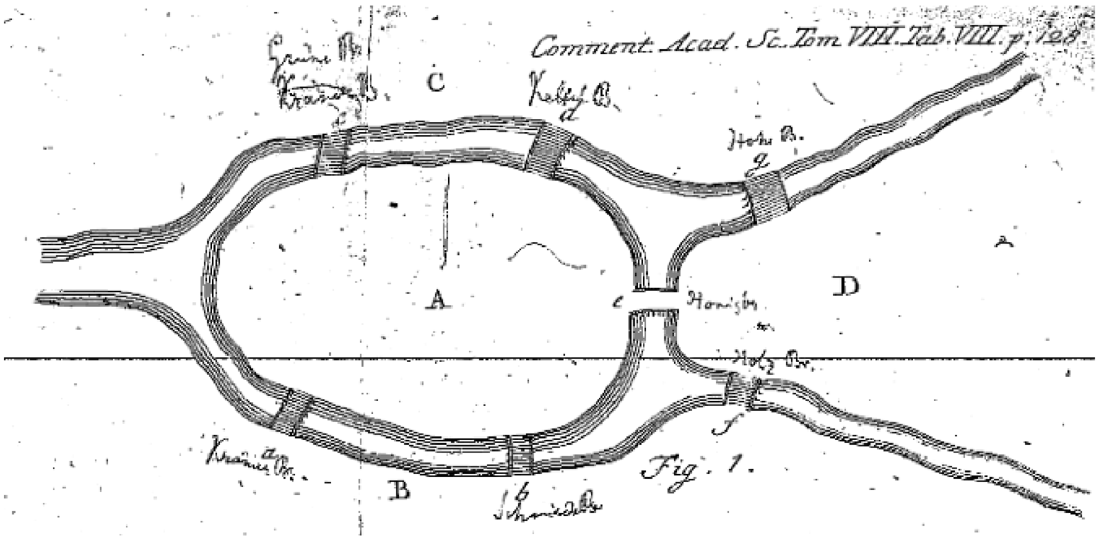
Segundo Mota [ca. 2013], Euler não referência as palavras “grafo”, “vértice”, “caminho” ou “grau”, utilizando apenas terminologias proveniente da Geografia, como pontes ou porções de terreno. Este artigo foi publicado apenas em 1741 na *Commentarii Academiae Scientiarum Imperialis Petropolitana*, e foi mais tarde reeditado numa outra edição da mesma publicação (*Novi Acta Commentarii*) em 1752.

Contudo não era claro os tipos de problemas e métodos que poderiam ser utilizados para este tipo de proposta. Dessa forma, não havia como o Problema das Sete Pontes de Königsberg pertencer ao domínio da Geometria de Posição, dado que tinha forma geométrica e não envolvia cálculos de distâncias. (MOTA [ca. 2013])

Artigo de Euler de 1736

O primeiro artigo que traz os princípios da Teoria dos Grafos foi o de Euler a respeito do problema das pontes de Königsberg; Mota [ca.2013] mostra que Euler produziu o esquema apresentado na Figura 1 e representou a ilha pela letra A e as restantes áreas pelas letras B, C e D. As pontes foram representadas pelas letras a, b, c, d, e, f e g.

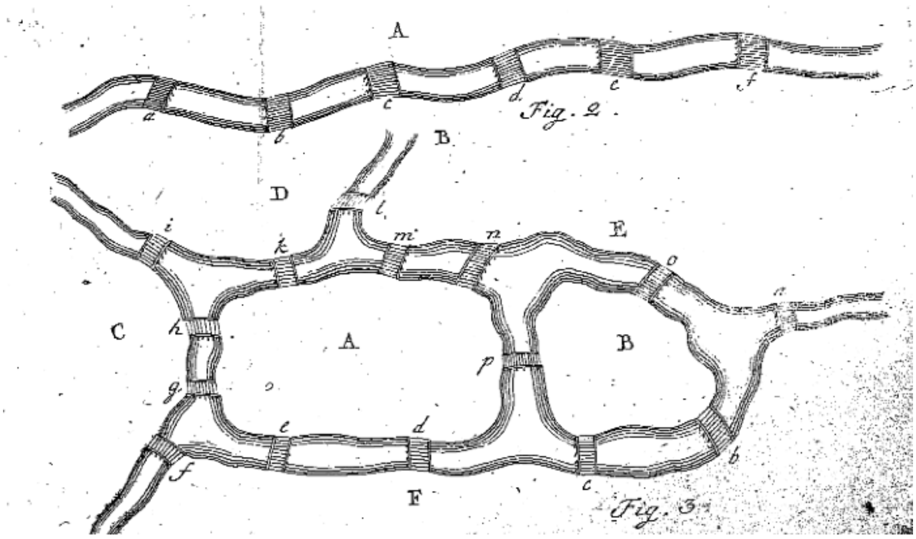
Figura 1 - Esquema do Problema das Sete Pontes de Königsberg.



Fonte: (Euler, 1741, p.128).

Euler menciona que uma lista com todos os percursos consumiria muito tempo e seria impossível ser replicada caso o número de pontes e áreas de terreno fosse aumentada, assim, apresentou as seguintes notações que seriam utilizadas. Primeiramente, explicou que uma sequência da forma $ABCD$ simboliza um trajeto que percorre as quatro áreas pela ordem apresentada, independentemente das pontes utilizadas em cada uma das três travessias. Relatou que a solução do problema solicitava da existência de uma sequência de oito letras dado que existem apenas sete pontes. Nesta sequência as letras A e B, por exemplo, devem estar adjacentes por duas vezes, dado que existem duas pontes unindo estas duas regiões. Posto isto, concluiu-se que o problema se reduzia a provar se existia, ou não, uma sequência de oito letras formada pelas letras A, B, C e D, na qual os seus pares aparecessem apenas uma vez. Euler não se limitou a encontrar a solução particular do problema das Sete Pontes de Königsberg, e assim começou por considerar apenas uma área de terreno A, com número indeterminado de pontes (Figura 2), que deveriam ser todas atravessadas uma única vez. Mota [ca.2013]

Figura 2 – Representação de Euler para seis pontes em uma única área



Fonte: (Euler, 1741, p. 128)

Euler descreveu o número de vezes que a letra A surgiu numa sequência, independentemente do percurso se iniciar, ou não na zona A, no caso do número de pontes, n, ser um número ímpar:

Tabela 1- Pontes de Königsberg

Número de Pontes	Número de vezes em que a letra A surge na sequência.
1	1
3	2
5	3
$\begin{pmatrix}\vdots\end{pmatrix}$	$\begin{pmatrix}\vdots\end{pmatrix}$
n	$\frac{n+1}{2}$

Fonte: Mota [ca.2013]

Adaptando para o problema das Pontes de Königsberg:

Tabela 2 - Pontes de Königsberg Ajustada

Zona	Números de pontes	Número de vezes que a letra tem sequência
A	5	3
B	3	2
C	3	2
D	3	2
Total		9

Fonte: Mota [ca.2013]

Dado que o número total de letras da sequência do problema é oito, Euler comprovou que o percurso pretendido era impossível utilizando as sete pontes. No caso do número de pontes, n, ser par, é necessário identificar o ponto de partida, dado que o número de vezes que a letra A aparece depende de o ponto de partida ser, ou não, o ponto A. (MOTA. [ca. 2013])

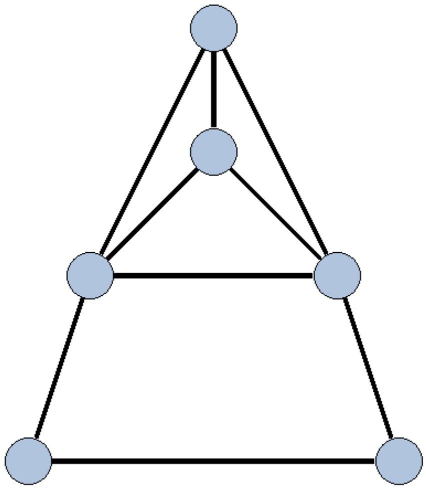
Portanto, segundo Mota [ca.2013], Euler afirma que se o número total de ocorrências das letras que denotam as diferentes áreas for igual ao número total de pontes mais 1, o percurso é possível se for iniciado em uma área com número ímpar de pontes. Por outro lado, se o número total de ocorrências das letras coincidir com o número de pontes, o percurso é possível se o iniciarmos numa área com um número par de pontes. Concluindo, Euler apresenta que S é somatório do número de pontes de cada área, podendo-se concluir que S é um número par, dado que cada ponte é contabilizada duas vezes, ou seja, se todas as áreas tiverem um número par de pontes, o percurso será sempre possível, dado que o número de letras da sequência corresponde ao número de pontes.

2.2 Conceitos Fundamentais

De acordo com Scheeren, Vanessa *et al.* (2017), um grafo é uma estrutura de um conjunto discreto e ordenado de pontos denominados Vértices e um conjunto de linhas denominadas Arestas, no qual cada aresta está ligada a pelo menos um vértice, assim como um grafo $G(V,E)$ é constituído de um conjunto V, finito e não vazio de n vértices, e um conjunto E de m arestas que são pares ordenados de elementos de V.

Graficamente um Grafo é representado por uma figura com **Vértices** (Nós), ligados por um traço denominado **Arestas.**

Figura 3 – Estrutura básica de um Grafo



Fonte: Autora – Software Grafos v.1.3.5

INCIDÊNCIA E GRAU:

Em um grafo, uma aresta que liga dois vértices é incidente de cada vértice, logo um vértice é incidente a uma aresta a ele conectada, sendo assim o grau de um vértice é dado pelo número de arestas incidentes a ele (PINTO, 1999, p.182).

ADJACÊNCIA E CAMINHO:

Os vértices adjacentes são vértices unidos por uma aresta, logo se chama caminho todos os vértices e arestas distintos, exceto primeiro e o último, que são iguais (JURKIEWICZ, 2008).

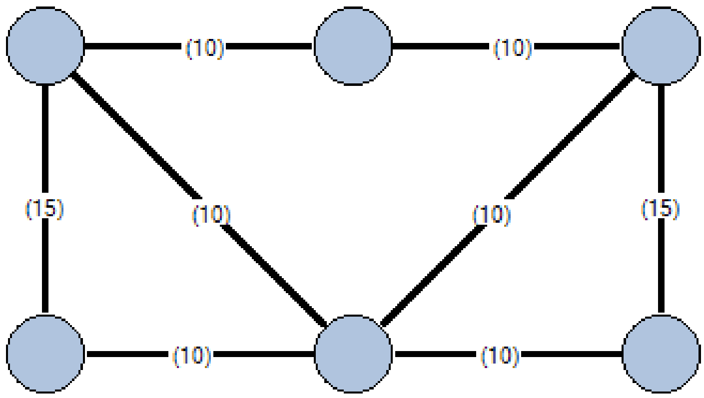
GRAFO SIMPLES E REGULAR:

São grafos que possuem apenas uma aresta entre os dois vértices. Em outras palavras, é dito regular se todo vértices tem o mesmo grau (BUENO, 2012. p. 17).

GRAFO CONEXO:

Um grafo é conexo se existe um caminho entre quaisquer dois dos seus vértices, como mostra a Figura 4 (LOESCH; HEIN, 1999, p. 192).

Figura 4 - Estrutura básica de um Grafo conexo e valorado.

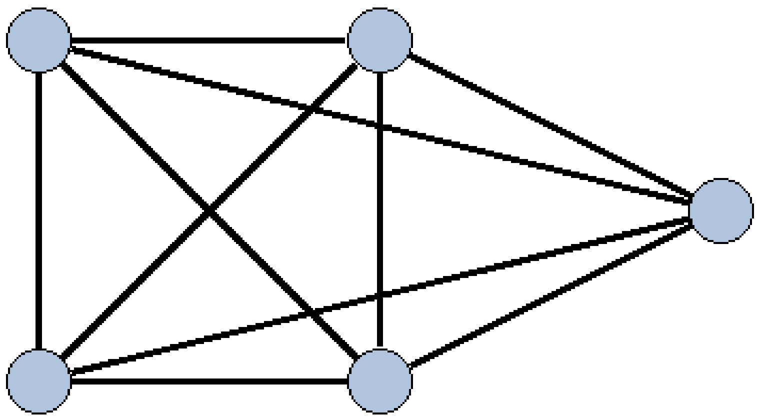


Fonte: Autora – Software Grafos v.1.3.5

GRAFO COMPLETO:

Um grafo completo ocorre quando todos os vértices se ligam à todos os outros vértices, ou seja todo vértice é adjacente aos demais, como mostra na Figura 5 (BUENO, 2012. p. 22).

Figura 5 - Grafo Completo



Fonte: Autora – Software Grafos v.1.3.5

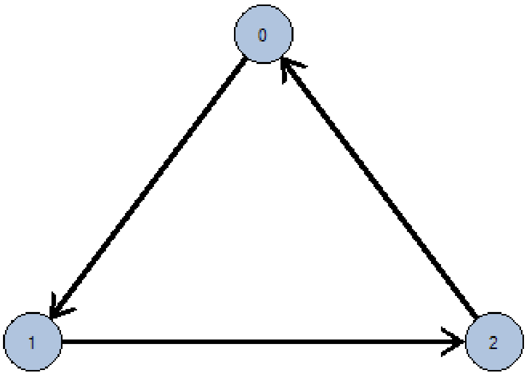
GRAFO VALORADO:

É a existência de uma ou mais funções relacionadas ao vértice ou aresta com um conjunto de números, como mostrado na Figura 4 (LOESCH; HEIN, 1999, p. 192).

GRAFO DIRECIONADO:

É um grafo com direção, ou seja, é a atribuição de um sentido para cada vértice, como mostra a Figura 6 (LOESCH; HEIN, 1999, p. 190).

Figura 6 - Estrutura básica de uma Grafo Direcionado



Fonte: Autora – Software Grafos v.1.3.5

ORDEM E DIMENSÃO:

Ordem são números de vértices de cada grafo e Dimensão é o número de arestas de cada grafo (BUENO, 2012. p. 16).

CICLO E ACÍCLICO:

Ciclo é um caminho de extremos idênticos, ou seja, o primeiro e o último vértice coincidem e grafo acíclico é aquele que não possui ciclos (LOESCH; HEIN, 1999, p. 192).

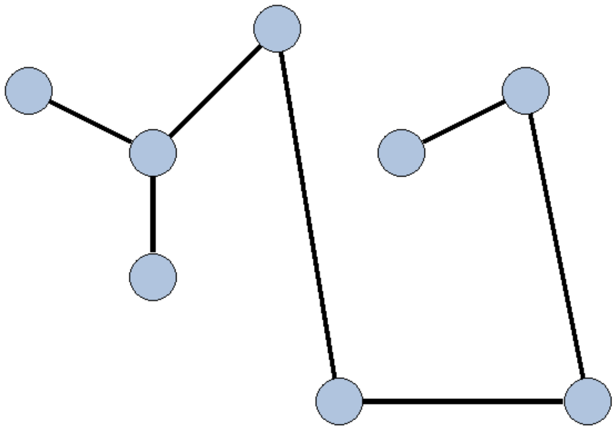
FONTE E DESTINO:

Fonte é o primeiro vértice de uma aresta direcionada de um grafo e o segundo diz-se destino (LOESCH; HEIN, 1999. p. 191).

ÁRVORE:

É um grafo conexo que não possui ciclos como subgrafos, como mostra na Figura 7 (JURKIEWICZ, 2008. p. 21).

Figura 7 - Árvore

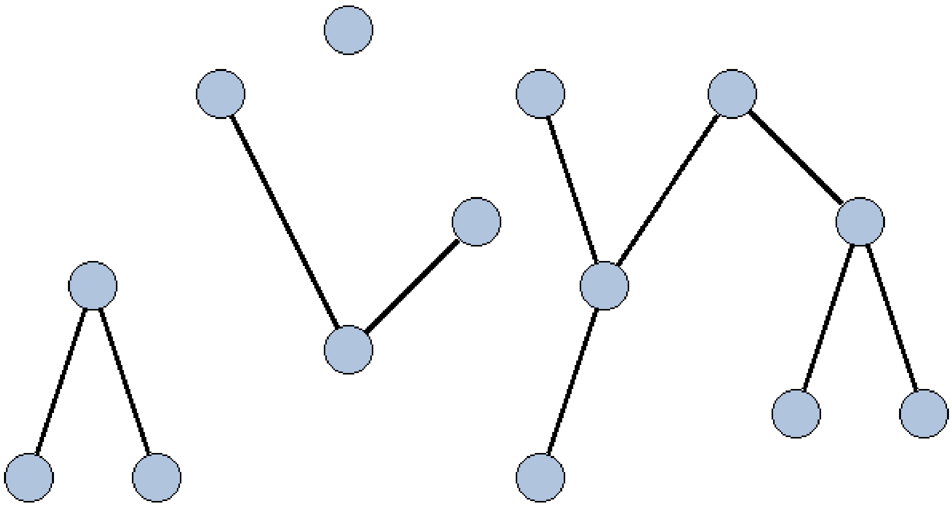


Fonte: Autora - Software Grafos v.1.3.5

FLORESTA:

Uma floresta é um grafo acíclico, desconexo composto pela união disjunta de árvores, conforme a Figura 8 (JURKIEWICZ, 2008. p. 69).

Figura 8 - Floresta



Fonte: Autora - Software Grafos v.1.3.5

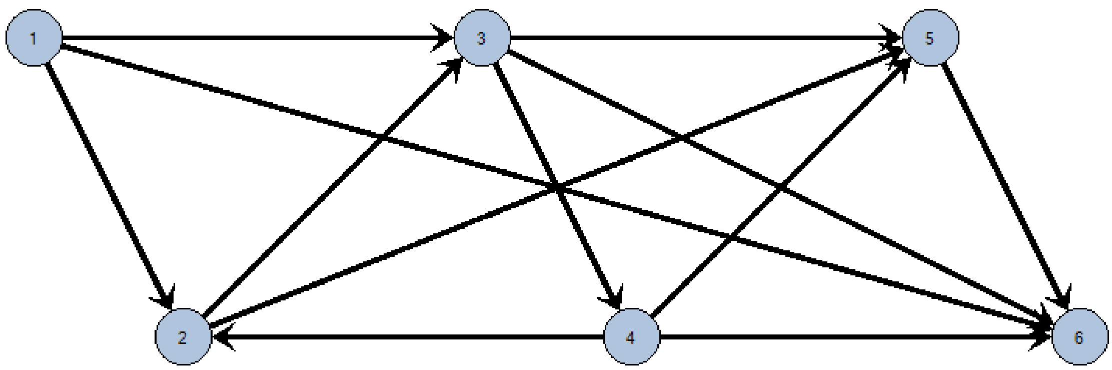
2.3 Fluxo de Redes

Taha (2008, p.105) afirma que uma rede se constitui de um conjunto de nós conectados por ramos (arcos). Essa rede é definida como (N, A), na qual N é o conjunto de nós e A é o conjunto de ramos. Relacionada à cada rede está um fluxo que em geral é limitado pela capacidade de seus ramos, que pode ser finita ou infinita.

Como exemplo, a rede da Figura 8 mostra que:

$$N = \{1,2,3,4,5,6\}$$
$$A = \{(1,2),(1,3),(1,6),(2,3),(2,5),(3,4),(3,5),(3,6),(4,2),(4,5),(4,6),(5,6)\}$$

Figura 9 - Estrutura básica de uma Rede (N, A).



Fonte: Autora – Software Grafos v.1.3.5

2.4 Software Grafos v.1.3.5

Existem vários softwares disponíveis no meio eletrônico para o problema que envolvem a Teoria dos Grafos, como por exemplo o *Graphpar, Dia Diagram Editor, Gephi,* entre outros. Contudo para gerar, testar e analisar o caminho mínimo e a árvore geradora mínima foram utilizados algoritmos construídos com a Linguagem de Programação C, juntamente com o compilador GCC (GNU *Compiler Collection*) e também o software Grafos na versão v.1.3.5. Assim a partir dos testes foi possível verificar que ambos geram os mesmos resultados, validando assim os resultados obtidos com o software Grafos. Diante disso, se optou pela utilização do software Grafos, pela simplicidade da interface, facilitando os testes realizados.

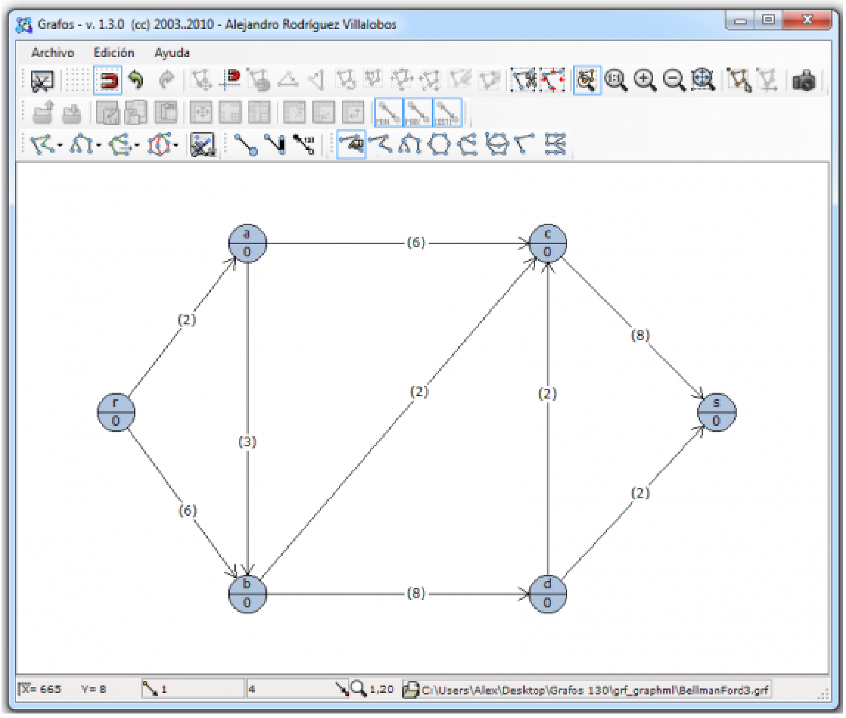
De acordo o site para download, Villalobos (2012) informa que o software Grafos é indicado para o ensino e aprendizagem da teoria dos grafos e outras disciplinas relacionadas, como engenharia industrial, logística e transporte, pesquisa operacional, projeto de rede, etc. Os gráficos podem ser usados perfeitamente para modelar e resolver problemas reais de um determinado tamanho e complexidade.

Um grafo representa um modelo de realidade comercial na forma de uma rede. Este modelo pode ser analisado sob diferentes pontos de vista, graças aos algoritmos e funções incorporados no software Grafos, que foi criado em 2003 por Alejandro Rodríguez Villalobos, como parte de um projeto de pesquisa e desenvolvimento de aplicativos de computador de design modular orientados para o ensino, pesquisa e trabalho profissional de engenharia da organização industrial.

Segundo o Villalobos (2012) no qual é disponível para download do programa, o Grafos tem filosofia *"desenhar, modelar, resolver e analisar".* Isso tem como objetivo oferecer ao usuário total liberdade para lidar e resolver problemas gráficos. Podendo desenhar livremente o gráfico sem se preocupar com a análise ou o algoritmo que usará posteriormente. Os gráficos o notificam em caso de inviabilidade ou qualquer outro requisito para uma análise específica.

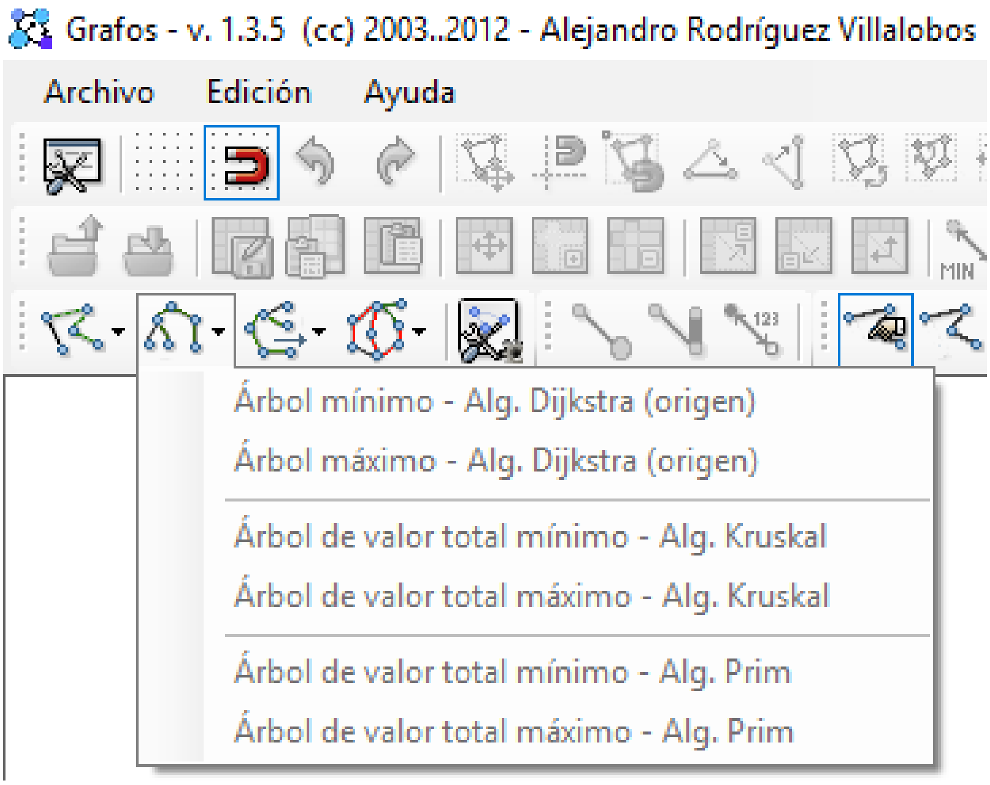
Este software possui várias funções, como a criação de um grafo simples, como apresentado na Figura 9 assim como a implementação dos Algoritmos de Dijkstra, Kruskal, Prim e Floyd conforme pode-se observar nas Figuras 10 e 11, respectivamente

Figura 10 - Software Grafos v.1.3.5



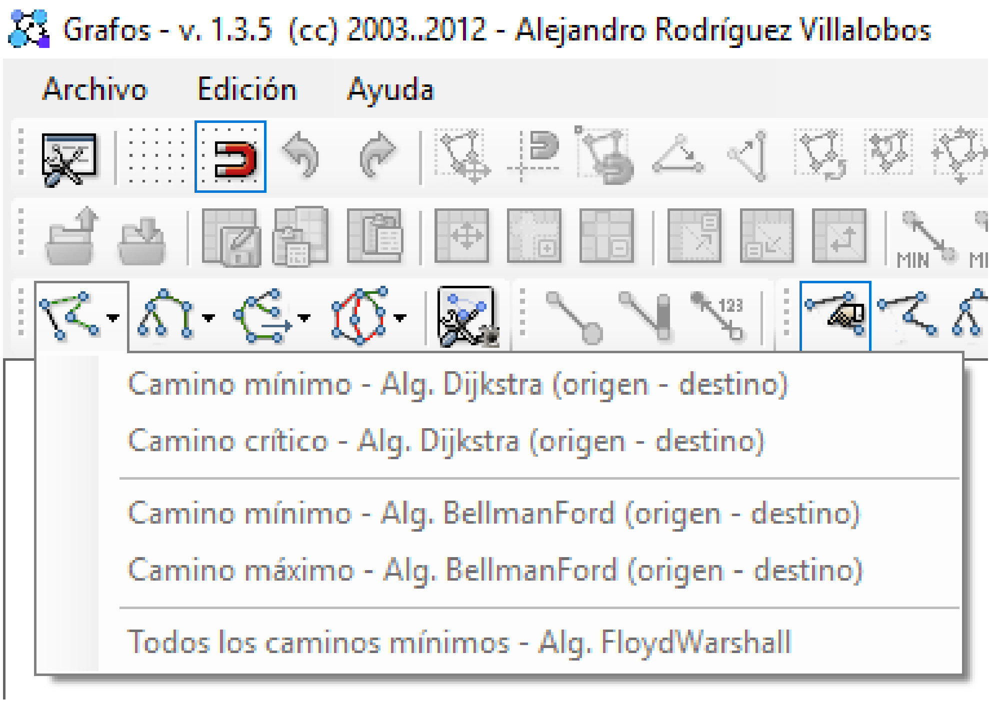
Fonte: Site do Software Grafos v.1.3.5.

Figura 11 - Algoritmos do Software Grafos v.1.3.5



Fonte: Autora (2019) - Software Grafos v.1.3.5.

Figura 12 - Algoritmos do Software Grafos v.1.3.5



Fonte: Autora (2019) - Software Grafos v.1.3.5

3. ALGORITMOS PARA SOLUÇÃO

Segundo Taha (2008, p.105), problemas como este são determinados por algoritmos de otimização de redes na quais podem ser classificados como: i) árvore geradora mínima: conecta todos os nós de uma rede, usando o comprimento total mais curto de ramos conectores; ii) algoritmo de caminho mais curto: determina o caminho mais curto entre um destino e uma origem de uma rede; iii) algoritmo de fluxo máximo: algoritmo que se baseia em achar rotas de passagem que maximizam a quantidade total de fluxo de sua origem, até um escoadouro (onde se termina o fluxo); iv) algoritmo de caminho crítico: fornece meios analíticos para programar determinada atividade, onde se definem os tempos máximos e mínimos das atividades.

Deste modo, nesta seção serão apresentados os algoritmos para solução do problema proposto, dividido neste trabalho na seção 3.1 com algoritmos para solução dos caminhos mínimos e 3.2 com algoritmos para solução com árvores geradoras mínimas.

3.1 Algoritmos para solução dos Caminhos Mínimos

Na resolução de problemas que envolvem a determinação do caminho mais curto, foi analisado dados e recorrido a pesquisas onde destacam-se o Problema do Caixeiro Viajante, por ser conhecido como o principal problema de programação matemática em problemas de otimização linear, no qual envolve roteamento de pontos de demanda e a busca em encontrar um grafo hamiltoniano de menor custo. (CAIXEIRO, 2009, p.1). Neste caso é válido lembrar que o problema do Caixeiro Viajante não é viável como solução, pois tem-se como solução apenas um transporte para a realização do deslocamento e para o problema proposto será utilizado mais de um transporte para os deslocamentos dos alunos à sede fixa do evento. Outra, a ser destacada é de Selong e Kripka (2009) no qual se tem o estudo de roteiros para a distribuição de ferro, de uma determinada empresa, localizada na cidade de Passo Fundo/RS, com o intuito de determinar o roteiro de menor distância entre as cidades a serem visitadas em cada distribuição, assim as análises foram feitas através das quatro principais rotas semanais da empresa. O resultado obtido, foi representado pelo modelo matemático de otimização linear com a ajuda computacional conhecida como Solver do Excel e pode ser concluído que todas as rotas investigadas eram boas, pois representavam rotas melhores que as utilizadas pelos funcionários da empresa. Assim, no primeiro roteiro otimizado obteve-se uma redução na distância total percorrida de aproximadamente $7{,}75\%$, no segundo $15{,}32\%$, no terceiro $10{,}88\%$ e no quarto $12{,}14\%$, o que, em média, demonstrou uma redução aproximada de $11{,}52\%$ que neste caso, a otimização pode trazer benefícios tanto para a empresa distribuidora, quanto para as lojas que vão receber a mercadoria. (SELONG; KRIPKA, 2009, p. 17-18).

Explorando as aplicações realizadas com o Algoritmo de Dijkstra para a busca de menor caminho, a pesquisa de Rizzi e Vicente (2011) teve destaque por demandar de dois objetivos, contidos em um projeto agrícola para uma propriedade rural, situada no noroeste do Paraná. O primeiro consiste em definir por qual caminho deveria passar uma rede de tubos, destinada à ligação de inúmeros pontos em uma área de cultivo, a fim de minimizar a quantidade de tubos. O segundo objetivo constitui-se em encontrar o menor caminho para a passagem de um veículo, que deve visitar esses pontos citados. Após o uso dos recursos em teoria dos grafos, os objetivos denominados pelos autores como problemas, foram resolvidos, o primeiro, por meio de uma árvore geradora mínima que se obteve por meio do Algoritmo de Dijkstra e, o segundo, por meio de um algoritmo para o problema do caixeiro viajante. As soluções destes problemas mostraram que o produtor poderia ter usado 139,4 metros a menos de tubos, em relação ao sistema já implantado, bem como um caminho com 65,3 metros a menos, em relação ao caminho ora utilizado. (RIZZI. VICENTE. 2011. p. 208).

Para o Algoritmo de Floyd foi encontrado o trabalho de STEINER, Maria *et al.* (2006) em que se trata da otimização da logística de atendimento aos usuários de uma rede de distribuição de energia elétrica, aplicada à agência do Portão, localizada em Curitiba, PR, de responsabilidade da Companhia Paranaense de Energia Elétrica (COPEL). Através de técnicas da área de Pesquisa Operacional, o Algoritmo de Floyd, determina de forma otimizada o dimensionamento de equipes de atendimento para a referida agência e o despacho otimizado das mesmas aos locais das ocorrências, visando um atendimento satisfatório aos usuários. Como conclusão, o algoritmo de Floyd, definiu as matrizes de custos e de trajetos entre um conjunto de nós, considerando as ligações entre os mesmos. Desta forma, tem-se como despachar de forma ótima, as equipes às ocorrências. Pode-se, inclusive, obter as matrizes de custos e trajetos preliminarmente aos despachos, considerando diversos horários do dia, assim, dada uma determinada ocorrência, faz-se a leitura das matrizes de Floyd correspondentes ao horário em questão e o despacho ocorre em sequência, de acordo com a prioridade de cada ocorrência. STEINER, Maria *et al.* (2006, p. 1-15)

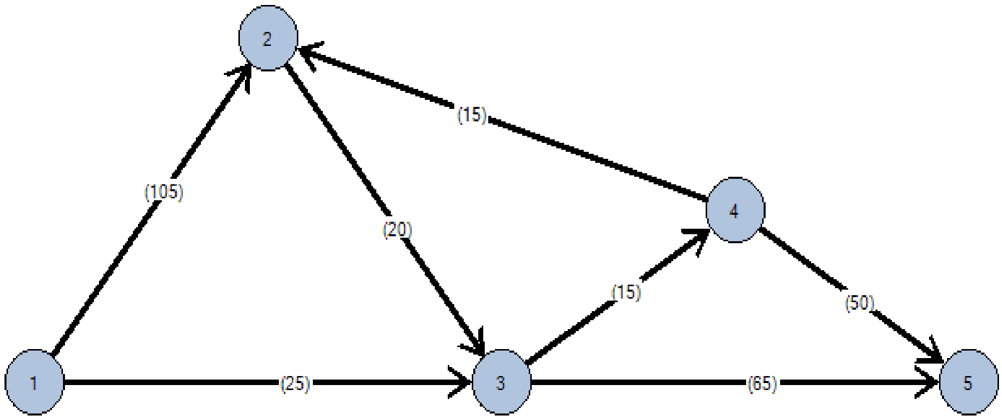
3.1.1 Algoritmo de Dijkstra

Segundo Méndez e Guardia (2008), o algoritmo de Dijkstra encontra o menor caminho de um nó fonte ou origem s até os outros nós de uma rede orientada para o caso em que todos os pesos dos arcos sejam não negativos, dividindo os nós em dois grupos: rotulados permanentes e rotulados temporários. A distância rotulada dos nós permanentes representa a menor distância do nó fonte até um outro nó. No caso dos nós temporários, esta distância demonstra o limite superior da distância do caminho mais curto até o nó. Assim, em cada iteração o rótulo do nó i é o menor caminho desde o nó fonte ao longo de um caminho que contém outros nós intermediários. O algoritmo escolhe o nó i com a mínima distância rotulada temporária para fazê-la permanente e visita os outros nós, ou seja, examina os arcos de $A(i)$ e atualiza as distâncias rotuladas dos nós adjacentes. O algoritmo termina quando todos os nós são designados como permanentes. (MÉNDEZ; GUARDIA. 2008, p. 7)

Segundo Taha (2008, p. 111) seja u_i a menor distância do nó de origem 1 ao nó i, e defina-se $d_{ij}(\geq 0)$ como o comprimento do arco (i,j). Logo, o algoritmo define o rótulo para um nó imediatamente posterior, j, como $[u_j, i] = [u_i + d_{ij}, i], d_{ij} \geq 0$.

Considerando a rede representada na Figura 13, com um conjunto de 5 nós e 7 arestas, é possível encontrar o menor caminho entre o nó 1 e qualquer outro nó da rede aplicando o Algoritmo de Dijkstra manualmente.

Figura 13 - Exemplo de um Grafo para encontrar o caminho mínimo de Dijkstra.



Fonte: Autora – Software Grafos v.1.3.5

- Iteração 0 – Designe o rótulo *permanente* ao nó 1.

- Iteração 1 – Os nós 2 e 3 podem ser alcançados com base no nó 1, assim mostra a tabela abaixo:

Tabela 3 - Lista de nós1

Nó	Rótulo	Status
1		Permanente
2	$[0 + 105{,}1] = [105{,}1]$	Temporário
3	$[0 + 25{,}1] = 25{,}1$	Temporário

Fonte: Autora (2019).

Para os dois rótulos temporários $[105{,}1]$e $[25{,}1]$, o nó 3 resulta na menor distância$(u_3 = 25)$. Assim, o status do nó 3 muda para permanente.

- Iteração 2 – Os nós 4 e 5 podem ser alcançados com base no nó 3, e a lista de nós fica como nos mostra a Tabela 4:

Tabela 4 - Lista de nós 2

Nó	Rótulo	Status
1		Permanente
2	$[105,1]$	Temporário
3	$[25,1]$	Permanente
4	$[25 + 15,3] = [40,3]$	Temporário
5	$[25 + 65,3] = [90,3]$	Temporário

Fonte: Autora (2019).

- Iteração 3 – Nós 2 e 5 podem ser alcançados com base no nó 4. Assim, a lista de nós rotulados é atualizada:

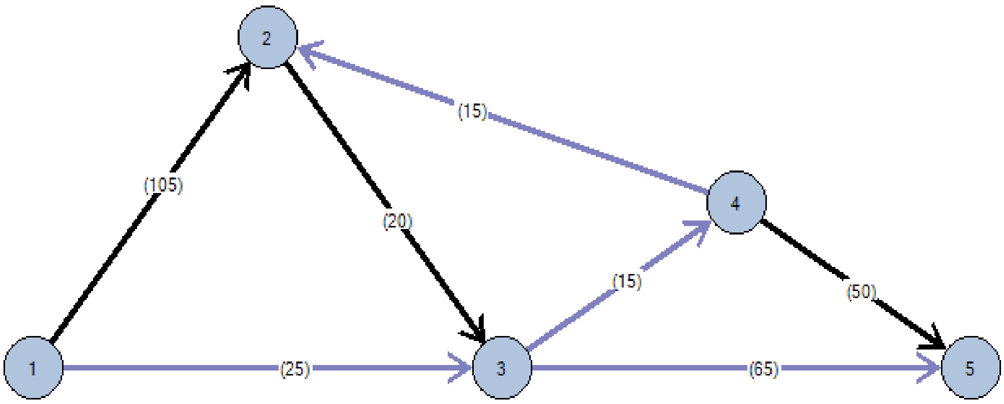
Tabela 5 - Lista de nós 3

Nó	Rótulo	Status
1		Permanente
2	$[40 + 15,4 = 55,4]$	Temporário
3	$[30,1]$	Permanente
4	$[40,3]$	Permanente
5	$[90,3] \rightarrow [40 + 50,4] = [90,4]$	Temporário

Fonte: Autora (2019).

- Iteração 4 – Só o nó 3 pode será obtido com base no nó 2. Apesar disso, o nó 3 tem um rótulo permanente e não pode ser rotulado novamente, fazendo com que a nova lista de rótulo permaneça a mesma da iteração 3, exceto que o rótulo no nó 2 agora é permanente. Como o nó 5 não leva a outros nós, seu status é convertido em permanente e o processo termina, como mostra na Figura 13.

Figura 14 - Exemplo de caminho mínimo usando o Algoritmo de Dijkstra



Fonte: Autora - Software Grafos v.1.3.5

Como citado anteriormente o procedimento do algoritmo continua até que se tenham todos os nós rotulados como permanente e assim é finalizado. Desta forma, a rota finalizada foi $1 \rightarrow 3 \rightarrow 4 \rightarrow 2$ com um comprimento total de 55 unidades de comprimento.

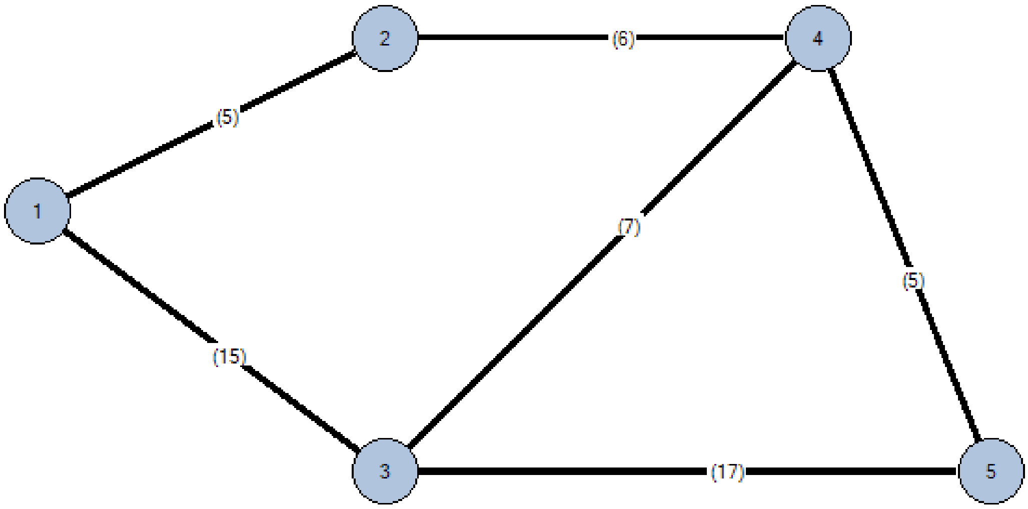
3.1.2 Algoritmo de Floyd

Segundo SOUZA, Rafael et.al (2009, p. 5) o algoritmo de Floyd ajuda a solucionar problemas de caminhos mínimos entre vértices de um grafo orientado, em que principalmente recebe como entrada uma matriz de adjacência e em seguida, é calculado para cada par de vértices o menor caminho entre estes.

O Algoritmo de Floyd é mais abrangente e permite a determinação do caminho mais curto entre quaisquer dois nós da rede. O Algoritmo representa uma rede de n nós, uma matriz quadrada com $n$linhas e n colunas. A entrada (i, j) da matriz dá a distância d_{ij}, do nó $i$ao nó j, que é finita se i estiver ligado diretamente a j, caso contrário é infinita (TAHA, 2008, p.113).

Para demonstrar algumas iterações do Algoritmo de Floyd, foi construída uma rede hipotética, com 5 nós e 6 arestas valoradas, sendo possível encontrar a solução pelo Algoritmo de Floyd de forma manual. A rede está representada na Figura 14.

Figura 15 – Exemplo de um Grafo para encontrar o caminho mínimo de Floyd.



Fonte: Autora – Software Grafos v.1.3.5

- Iteração 0 – As matrizes D_0 e S_0 dão a representação inicial da rede D_0 que são simétricas, exceto por $d_{53} = \infty$, porque nenhum tráfego é permitido do nó 5 ao nó 3.

Tabela 6 - Matriz D_0 e Matriz S_0

	1	2	3	4	5
1	–	5	15	∞	∞
2	5	–	∞	6	∞
3	15	∞	–	7	17
4	∞	6	7	–	5
5	∞	∞	∞	5	–

	1	2	3	4	5
1	–	2	3	4	5
2	1	–	3	4	5
3	1	2	–	4	5
4	1	2	3	–	5
5	1	2	3	4	–

Fonte: Autora (2019).

- Iteração 1 – Determine $k = 1$. A linha pivô e a coluna pivô são mostradas pela primeira linha e pela primeira coluna sombreadas em tom mais claro na matriz D_0. As células sombreadas em toma mais escuro, d_{23} e d_{32}, são as únicas que podem sem melhoradas pela operação tripla. Assim, D_1 e S_1 são obtidas de D_0 e S_0 da seguinte forma:

1. Substitua d_{23} por $d_{21} + d_{13} = 5 + 15 = 20$ e determina $S_{23} = 1$.
2. Substitua d_{32} por $d_{31} + d_{12} = 15 + 5 = 20$ e determina $S_{32} = 1$.

Tabela 7 - Matriz D_1 e Matriz S_1

	1	2	3	4	5
1	−	5	15	∞	∞
2	5	−	**20**	6	∞
3	15	**20**	−	7	17
4	∞	6	7	−	5
5	∞	∞	∞	5	−

	1	2	3	4	5
1	−	2	3	4	5
2	1	−	**1**	4	5
3	1	**1**	−	4	5
4	1	2	3	−	5
5	1	2	3	4	−

Fonte: Autora (2019).

- Iteração 2 – Determine $k = 2$, como mostram a linha e a coluna sombreada em tom mais calor em D_1. A operação tripla é aplicada às células sombreadas em tom mais escuro em D_1e S_1. As mudanças resultantes são representadas em negrito em D_2 e S_2

Tabela 8 - Matriz D_2 e Matriz S_2

	1	2	3	4	5
1	−	5	15	**11**	∞
2	5	−	20	6	∞
3	15	20	−	7	17
4	**11**	6	7	−	5
5	∞	∞	∞	5	−

	1	2	3	4	5
1	−	2	3	**2**	5
2	1	−	**1**	4	5
3	1	**1**	−	4	5
4	**2**	2	3	−	5
5	1	2	3	4	−

Fonte: Autora (2019).

- Iteração 3 – Determine $k = 3$, como mostram a linha e a coluna sobreadas em D_2. As novas matrizes são dadas por D_3e S_3

Tabela 9 - Matriz D_3 e Matriz S_3

	1	2	3	4	5
1	–	5	15	11	**16**
2	5	–	**13**	6	**27**
3	15	∞	–	7	17
4	11	6	7	–	5
5	∞	∞	∞	5	–

	1	2	3	4	5
1	–	2	3	2	3
2	1	–	1	4	3
3	1	1	–	4	5
4	2	2	3	–	5
5	1	2	3	4	–

Fonte: Autora (2019).

- Iteração 4 – Dado um $k = 4$, como mostram a linha e a coluna sobreadas em D_3. As novas matrizes são dadas por D_4 e S_4.

Tabela 10 - Matriz D_4 e Matriz S_4

	1	2	3	4	5
1	–	5	15	11	**16**
2	5	–	**13**	6	**11**
3	15	13	–	7	12
4	11	6	7	–	5
5	**15**	**10**	**12**	5	–

	1	2	3	4	5
1	–	2	3	**2**	**4**
2	1	–	**4**	4	**4**
3	1	**4**	–	4	**4**
4	2	2	3	–	5
5	**4**	**4**	**4**	4	–

Fonte: Autora (2019).

- Iteração 5 – Determine $k = 5$. Nenhuma melhoria adicional é possível nessa iteração.

O algoritmo continua com iterações até que nenhuma melhoria adicional seja possível, nesse caso, até a iteração 5. Por fim, como $S_{12} = 2$, $S_{24} = 4$ e $S_{45} = 5$, nenhuma outra "dissecção" é necessária e $1 \rightarrow 2 \rightarrow 4 \rightarrow 5$ define a rota mais curta.

Para o problema proposto neste trabalho, observou-se que os Algoritmos de Dijkstra e Floyd não indicam uma sede de chegada ao evento SIEPE, e sim determinam um campus como ponto de saída para os demais, não contemplando o objetivo da pesquisa. Logo, o modelo do problema foi remodelado como sendo uma árvore geradora mínima cujo algoritmos de solução são apresentados a seguir.

3.2 Algoritmos para problemas de Árvore Geradora Mínima

Segundo Pina (2011), o problema de Árvores Geradoras Mínimas também conhecida como “Árvore Geradora de Custo Mínimo” possui várias aplicações na atualidade, como por exemplo, projeto de redes de telecomunicação, fibra-ótica, redes de telefonia, projeto de rodovias, ferrovias, entre outros. Essas aplicações consistem na construção de grafos e resolução com algoritmos, como o Algoritmo de Prim e o Algoritmo de Kruskal.

Em pesquisas relacionadas ao problema de árvore geradora mínima, pode-se destacar Souza (2008, p.1-74) no qual apresentam um trabalho que descreve e analisa heurísticas para obter boas soluções viáveis para o problema da árvore geradora mínima probabilística (PMST). O PMST é uma generalização natural do problema clássico da árvore geradora mínima (MST) e é frequentemente um modelo mais realista. Assim, é possível utilizar heuristicamente a probabilidade dos nós para reduzir o custo ativo esperado da árvore geradora. Através da análise dos custos das árvores medidos e através de simulações, este recurso pode ser empregado como forma de medir o custo das árvores geradoras, alternativamente ao uso das equações para cálculo de custo ativo esperado, evitando-se assim problemas numéricos.

Em relação ao Algoritmo de Prim, destaca-se a pesquisa de Silva, Gomes, Frigieri (2016, p.1-6) no qual tem como objetivo minimizar a potência total de uma rede de sensores sem fio, propondo uma abordagem baseada nesse algoritmo, para calcular a potência de transmissão para cada nó, mantendo todos os nós conectados. Assim, foram avaliados diferentes cenários considerando o tempo computacional e a redução da potência total, onde foi possível comparar o desempenho do algoritmo para diferentes densidades de nós e áreas. Por meio dessa análise concluíram que o algoritmo proposto atingiu reduções na potência total acima de 65%, no qual foi possível identificar um impacto positivo no desemprenho em relação à redução de potência ao aumentar a densidade de nós da rede.

Já o Algoritmo de Kruskal, Guttoski (2006, p. 1-105) mostra sua aplicação no processo de otimização de consultas do PostgreSQL (sistema gerenciador de banco de dados, ou seja, código aberto), por sua adequação à definição de processos de otimização de consultas, assim seu espaço de busca é reduzido trazendo apenas uma árvore de execução é gerada como solução ótima. Os resultados demonstram que o algoritmo de Kruskal é um algoritmo viável para a técnica de otimização de consultas, por isso, quanto maior o número de relações que compõem a consulta, o processo de construção do plano será mais rápido.

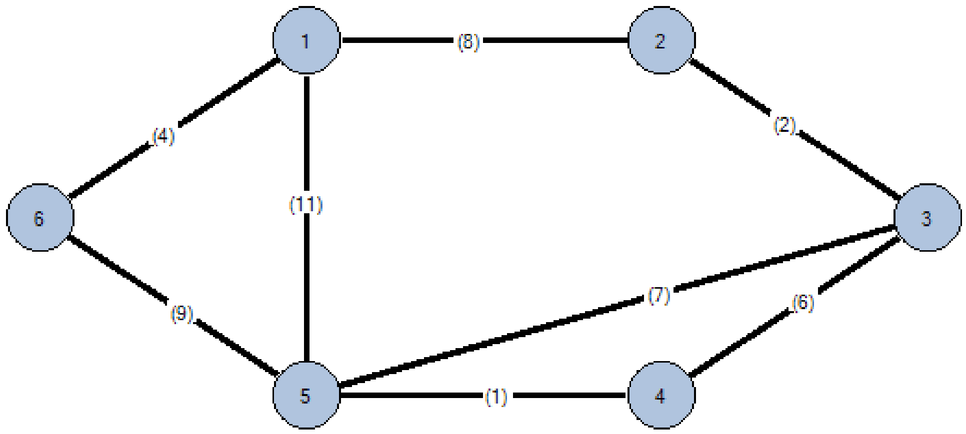
3.2.1 Algoritmo de Prim

Segundo Pina (2011), o algoritmo de Prim é usado para encontrar uma árvore geradora de custo mínimo em um grafo dado como G. A árvore geradora de um grafo G é um subgrafo de G formado pelo menor número de arestas que mantém o subgrafo ainda conexo. Já, a árvore geradora de custo mínimo é formada pelas arestas que, somando seus pesos, dará o menor custo total. Em primeiro lugar é escolhido um vértice j que constrói parcialmente uma árvore T. Após, une-se a T a aresta do grafo cujo peso é mínimo entre as demais com um vértice pertencente a T e outro vértice não pertencente a T. Logo, esse processo é repetido enquanto existam vértices que possam ser inseridos em T.

A execução do algoritmo de Prim é restrita a grafos conexos (GOODRICH; TAMASSIA, 2004, p. 368).

Exemplo do Algoritmo de Prim na Figura 15 e 16.

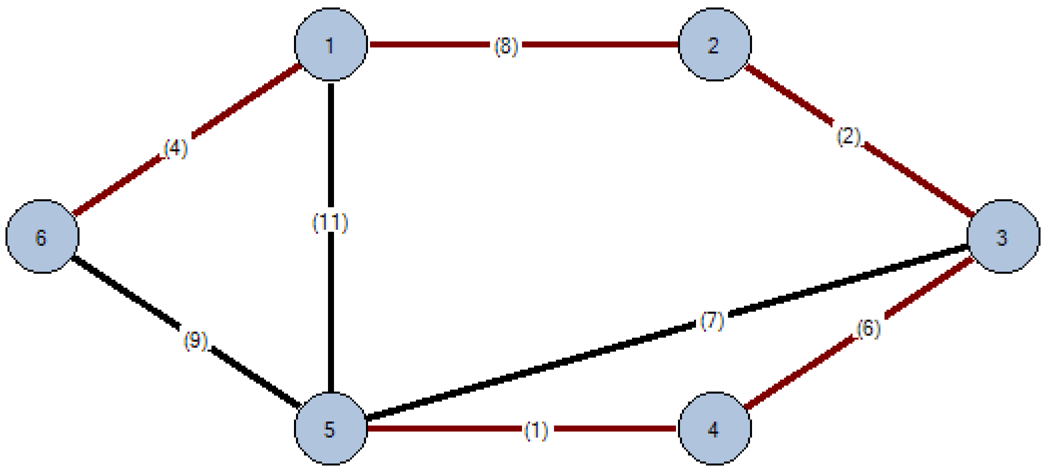
Figura 16 - Exemplo para Algoritmo de Prim



Fonte: Autora - Software Grafos v.1.3.5

Assim, através do Software Grafos v.1.3.5 que possui o Algoritmo de Prim, pode-se perceber a árvore geradora mínima destacada na cor bordo na Figura 16.

Figura 17 - Resultado após à aplicação do Algoritmo de Prim



Fonte: Autora - Software Grafos v.1.3.5

Neste caso, o algoritmo mantém uma única árvore que cresce a cada iteração com a inserção de um vértice.

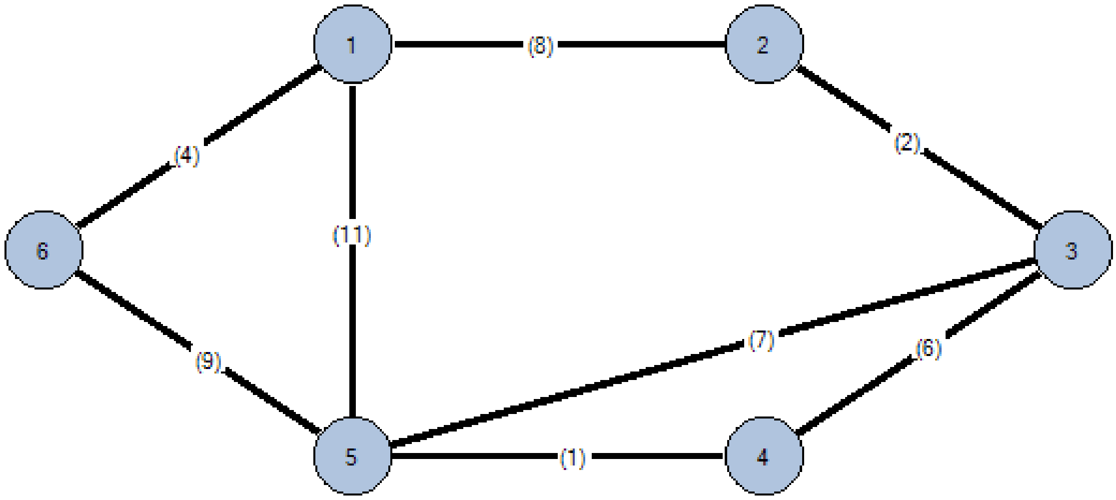
3.2.2 Algoritmo de Kruskal

Segundo Pina (2011) o algoritmo de Kruskal é utilizado para gerar árvores geradoras de custo mínimo em grafos desconexos. Inicialmente cada componente de um grafo é composta por apenas um vértice, na qual faz crescer uma floresta geradora se tornando conexa. O resultado é um conjunto de uma ou mais árvores.

Em uma melhor forma de compreender, segundo Quitzau (2003), o algoritmo de Kruskal começa com uma floresta de árvores formadas por apenas um vértice e procura, a cada passo, uma aresta que, além de conectar duas árvores distintas da floresta, possua peso mínimo.

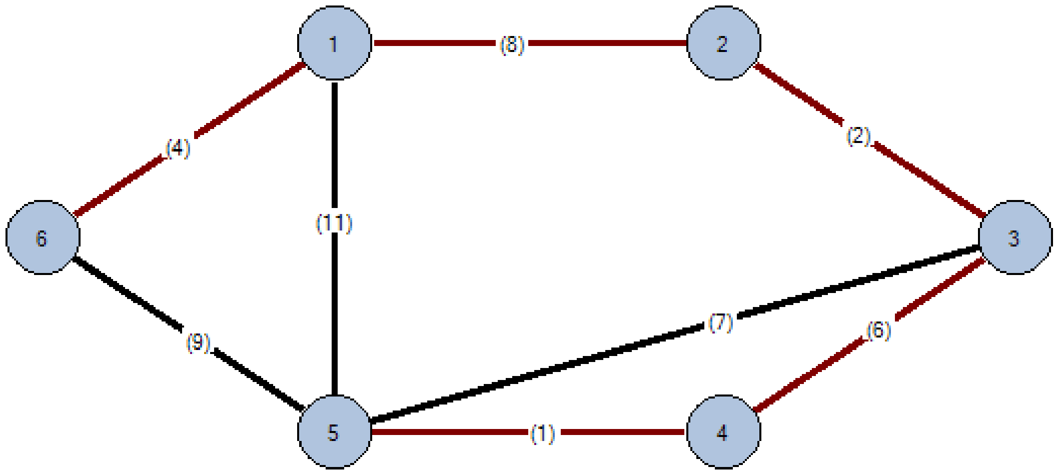
Utilizando o mesmo exemplo visto no Algoritmo de Prim, podemos notar que para Kruskal o resultado mostra-se igual ao anterior, como mostra na Figura 18 e na Figura 19 no qual seu resultado é mostrado na cor bordo.

Figura 18 - Exemplo para Algoritmo de Kruskal



Fonte: Autora – Software Grafos v.1.3.5

Figura 19 - Resultado após aplicação do Algoritmo de Kruskal



Fonte: Autora - Software Grafos v.1.3.5

Segundo Quitzau (2003), a diferença entre os dois está na forma como cada um escolhe uma aresta segura. No algoritmo de Kruskal, o conjunto A é uma floresta cujas componentes conexas vão se unindo a cada iteração. Já no algoritmo de Prim, A nunca deixa de ser uma árvore, uma vez que a aresta segura incluída a cada passo une um vértice que está fora da árvore a um vértice que já pertence a ela.

4. METODOLOGIA

A metodologia utilizada neste trabalho é baseada em Pesquisa Qualitativa, na qual, segundo Günther (2006), ao invés de utilizar instrumentos padronizados, a pesquisa qualitativa considera cada problema objeto de uma pesquisa específica para a qual são necessários instrumentos e procedimentos específicos. Tal postura requer, portanto, maior cuidado na descrição de todos os passos da pesquisa: a) delineamento, b) coleta de dados, c) transcrição e d) preparação dos mesmos para sua análise específica. (GÜNTHER, 2006, p. 204).

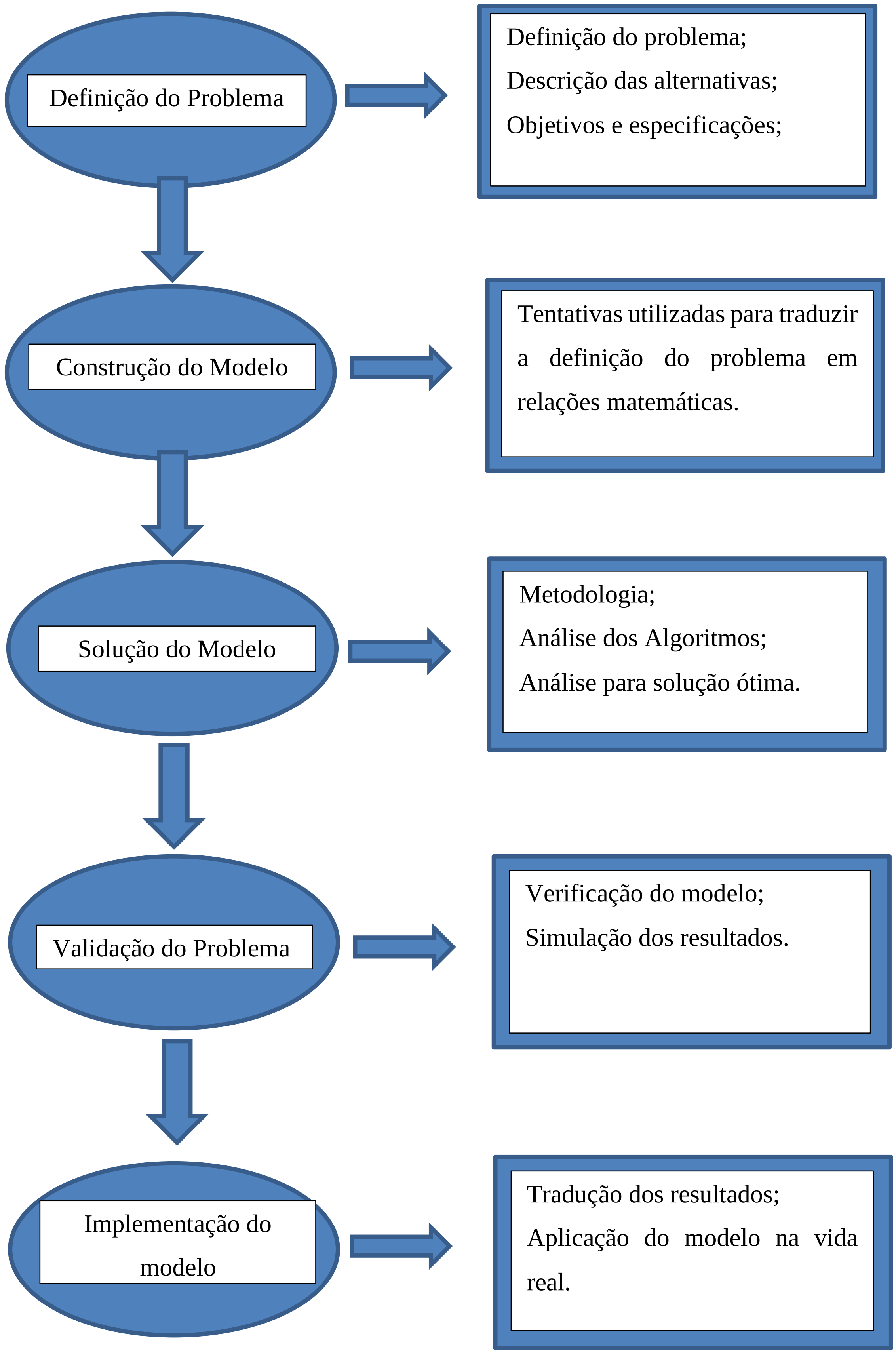
Desta forma, a Pesquisa Qualitativa possui três amplos caminhos do conhecimento conhecidos como: Pesquisa documental, Estudo de caso e Etnografia. Neste caso, o método de Estudo de caso, melhor se adequou a pesquisa estudada.

> O estudo de caso tem se tornado a estratégia preferida quando os pesquisadores procuram responder às questões "como" e "por quê" certos fenômenos ocorrem, quando há pouca possibilidade de controle sobre os eventos estudados e quando o foco de interesse é sobre fenômenos atuais, que só poderão ser analisados dentro de algum contexto de vida real. (GODOY,1995. p. 25)

Segundo Godoy (1995.p. 25), no estudo de caso, o pesquisador geralmente utiliza uma variedade de dados coletados em diferentes momentos, por meio de variadas fontes de informação. Produz relatórios que apresentam um estilo mais informal, narrativo, ilustrado com citações, exemplos e descrições fornecidos pelos sujeitos, podendo ainda utilizar fotos, desenhos, colagens ou qualquer outro tipo de material que o auxilie na transmissão do caso.

Os procedimentos adotados para o estudo de caso aqui proposto foram baseados nas premissas da Pesquisa Operacional (PO), que, segundo Taha (2008, p.4), é uma ciência que concretiza processos baseados na modelagem matemática de problemas que necessitam de um conjunto de alternativas e ações, e para isso faz-se previsões e comparações de valores, custos e eficiência. Tais procedimentos são divididos nas seguintes fases, como: Definição do Problema, Construção do modelo, Solução do modelo, Validação do modelo e Implementação da solução nos quais se tem as seguintes características apresentadas na Figura 20:

Figura 20 - Fluxograma



Fonte: Autora (2019).

Os modelos matemáticos de PO possuem como estrutura uma função objetiva e restrita, nas quais estão ancorados na matemática e estatística, possuindo uma vasta aplicabilidade em especial na Teoria dos Grafos no ramo de Fluxo de Redes, devido ao sucesso das fases que resultam na solução do modelo matemático depende em grande parte da criatividade e da experiência da equipe operacional.

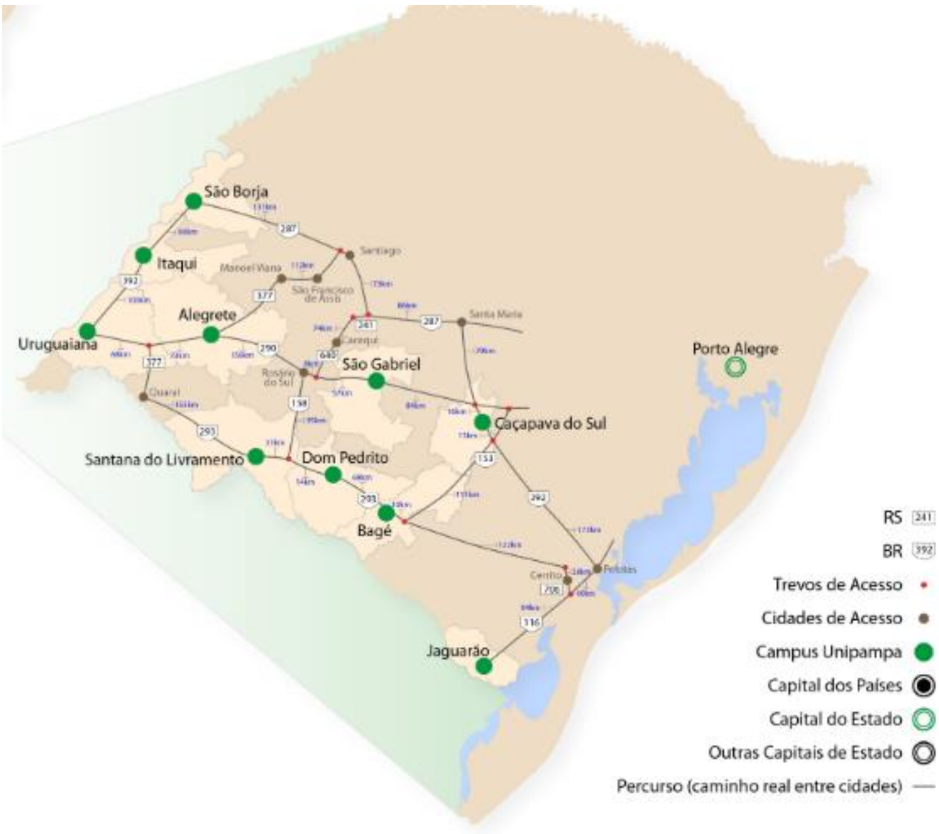
Isto quer dizer que para este trabalho, primeiramente foi estudado todos os conceitos fundamentais e realizada a modelagem matemática do problema através dos fatores que iam ser considerados no problema, logo em seguida obteve-se o referencial teórico no qual pode-se fazer a construção dos grafos para análises do suposto resultado.

Portanto, tendo em vista o problema proposto, a suposição será constituída do estudo e da produção de todos os grafos possíveis para verificação de qual será realmente o melhor caminho que envolva todos os campi e que não exija nenhum percurso complicado que possa a vir interromper o trajeto dos discentes ao evento.

5. DESENVOLVIMENTO DA PESQUISA

Segundo informações no site eletrônico da UNIPAMPA, a Universidade Federal do Pampa foi criada pelo governo federal por meio da lei nº 11.640, de 11/01/2008, para minimizar o processo de estagnação econômica onde está inserida, pois a educação viabiliza o desenvolvimento regional, procurando ser um operador da definitiva incorporação da região ao mapa do desenvolvimento do Rio Grande do Sul. Desta forma, a UNIPAMPA é constituída de dez campi no qual totalizam 69 cursos de graduação, dez mestrados acadêmicos, oito mestrados profissionais, quatro doutorados e 35 especializações. Os campi podem ser visualizados no mapa fornecido pelo site assim como mostra a Figura 21.

Figura 21 - Localização dos campi UNIPAMPA



Fonte: Site UNIPAMPA

Em 2008 a Universidade Federal do Pampa deu início ao evento SIEPE no qual teve sua primeira versão na cidade de Uruguaiana e é de grande importância para a universidade, pois através deste, ocorre várias exposições de trabalhos, seminários, palestras e minicursos. O SIEPE atende uma exigência do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul (FAPERGS), para que os bolsistas vinculados aos programas e projetos da universidade, tenham oportunidade de apresentar e divulgar à comunidade acadêmica suas pesquisas.

Segundo Marco Antônio Fontoura Hansen, reitor da UNIPAMPA, o evento ocorre, pela terceira vez na cidade de Santana do Livramento/RS, por ser um campus que com satisfação integra o lado uruguaio ao evento, desenvolvendo-se para a internacionalização. O evento é promovido pela Universidade Federal do Pampa (Unipampa) em parceria com *Universidad de La República Uruguay* (Udelar), Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense (IFSul) e *Universidad Tecnológica del Uruguay* (Utec).

Houve a necessidade de saber o número de inscritos no evento, desta forma foi enviado ao e-SIC (Sistema eletrônico do serviço de informação ao cidadão) perguntas sobre o mesmo, no qual obteve-se resposta com dados a partir da 7º edição do evento, considerando alunos de Graduação, Pós-Graduação e alunos de outras instituições, como mostra abaixo na Figura 22.

Figura 22 - Quantidade alunos no 7º, 8º, 9º e 10º SIEPE

7º SIEPE
Alegrete – 350 alunos
Bagé – 237 alunos
Caçapava – 120 alunos
Dom Pedrito – 83 alunos
Itaqui – 246 alunos
Jaguarão – 67 alunos
Santana do Livramento – 68 alunos
São Borja – 89 alunos
São Gabriel – 164 alunos
Uruguaiana – 384 alunos

8º SIEPE
Alegrete – 217 alunos
Bagé – 207 alunos
Caçapava – 88 alunos
Dom Pedrito – 81 alunos
Itaqui – 253 alunos
Jaguarão – 44 alunos
Santana do Livramento – 57 alunos
São Borja – 78 alunos
São Gabriel – 119 alunos
Uruguaiana – 558 alunos

9º SIEPE
Alegrete – 160 alunos
Bagé – 268 alunos
Caçapava – 92 alunos
Dom Pedrito – 107 alunos
Itaqui – 156 alunos
Jaguarão – 67 alunos
Santana do Livramento – 143 alunos
São Borja – 74 alunos
São Gabriel – 88 alunos
Uruguaiana – 283 alunos

10º SIEPE
Alegrete – 191 alunos
Bagé – 319 alunos
Caçapava – 164 alunos
Dom Pedrito – 75 alunos
Itaqui – 200 alunos
Jaguarão – 89 alunos
Santana do Livramento – 204 alunos
São Borja – 176 alunos
São Gabriel – 96 alunos
Uruguaiana – 417 alunos

Fonte: e-SIC

Com as pesquisas realizadas no site da Universidade Federal do Pampa foi encontrado também o PDI 2019-2023 que informam dados sobre as infraestruturas físicas dos campi, da mesma maneira como o número de alunos de cada campus e comparações dos anos de 2014 e 2018 em relação às melhorias. Esses documentos estão disponíveis para o público em geral.

Desta maneira em análise a esse documento e à todas as planilhas fornecidas pode-se observar a infraestrutura física e números de alunos na Figura 23, no qual mostra a evolução dos campi UNIPAMPA.

Figura 23 - Evolução Área Acadêmica - PDI 2014/2018

Campus/ Unidade	Até 2013	2014 - 2018	Aumento Área Construída (m^2)	Nº Alunos 2018	m^2 x Discentes	N º alunos 2014	m^2 x Discentes
Alegrete	4.902,25	240,000	5.142,25	1.211	4,25	1216	4,03
Bagé	8.557,50	500,80	9.058,30	1.468	6,17	1495	5,72
Caçapava do Sul	2.288,95	2.586,70	4.875,65	501	9,73	557	4,11
Dom Pedrito	2.830,52	408,64	3.239,16	740	4,38	514	5,51
Itaqui	3.696,61	341,10	4.037,71	1.067	3,78	844	4,38
Jaguarão	2.813,27	0,00	2.813,27	481	5,85	579	4,86
Santana do Livramento	2.308,63	0,00	2,308,63	926	2,49	834	2,77
São Borja	4.136,28	0,00	4.136,28	779	5,31	758	5,46
São Gabriel	2.143,68	2.867,68	5.011,36	510	9,83	527	4,07
Uruguaiana	11.378,05	0,00	11.378,05	1.542	7,38	1361	8,36
TOTAL	45.055,73	6.944,92	52.000,65	9.225	5,64	8685	5,19

Fonte: PDI UNIPAMPA 2019-2023

Porém para que este evento ocorra é necessário que haja transportes suficiente para levar os alunos de um campus ao outro, para isso, achou-se necessário trazer como informação a relação da frota de ônibus dos campi, o que se pode observar na Tabela 11 a seguir.

Tabela 11 - Frotas dos campi

Veículos	
Bagé	Micro-ônibus
Caçapava do Sul	Ônibus
Dom Pedrito	Micro-ônibus
Itaqui	Micro-ônibus
Jaguarão	Micro-ônibus
Santana do Livramento	Micro-ônibus
São Borja	Micro-ônibus
São Gabriel	Micro-ônibus
Uruguaiana	Micro-ônibus
Reitoria	Van / Ônibus

Fonte: Autora (2019)

Para estes dados não é disponibilizado o valor quantitativo de transportes de cada campus UNIPAMPA. A partir disso, associando todos os valores e priorizando o campus que poderá comportar o evento, em relação ao número de alunos, número de alunos inscritos a partir da 7º edição e a infraestrutura dos campi, pode-se observar que qualquer um dos campi poderiam ser sede deste evento, porém os campi Uruguaiana e Bagé possuem maior estrutura física para comportar o evento SIEPE e também dispõem de um número maior tanto de alunos matriculados como de inscritos no evento, nos quais favorece um número menor de transportes para se deslocar, dessa maneira os dois campi serão melhor analisados com os Algoritmos de Prim e Kruskal.

6. APRESENTAÇÃO E ANÁLISE DE RESULTADOS

Nesta seção serão apresentados os resultados e análises da aplicação dos Algoritmos de Prim e Kruskal, nos quais será subdivido em 6.1 Campus Bagé e 6.2 Campus Uruguaiana.

6.1 Campus Bagé

O Campus Bagé da Universidade Federal do Pampa é localizado na Av. Maria Anunciação Gomes de Godoy, 2650 – Bairro Malafaia, composta pelo total de 1468 alunos, 11 cursos de graduação, três especializações e cinco mestrados. Por conta de sua estrutura física, número de alunos no campus e números de alunos inscritos no evento, foi um dos escolhidos para ser a possível sede do evento SIEPE, como mostra na Tabela 12 e Figura 24.

Tabela 12 - Campus Bagé

INFRAESTRUTURA (m^2)	NÚMERO DE ALUNOS	TOTAL DE ALUNOS A PARTIR DA 7º EDIÇÃO
9.058,30	1468	1031

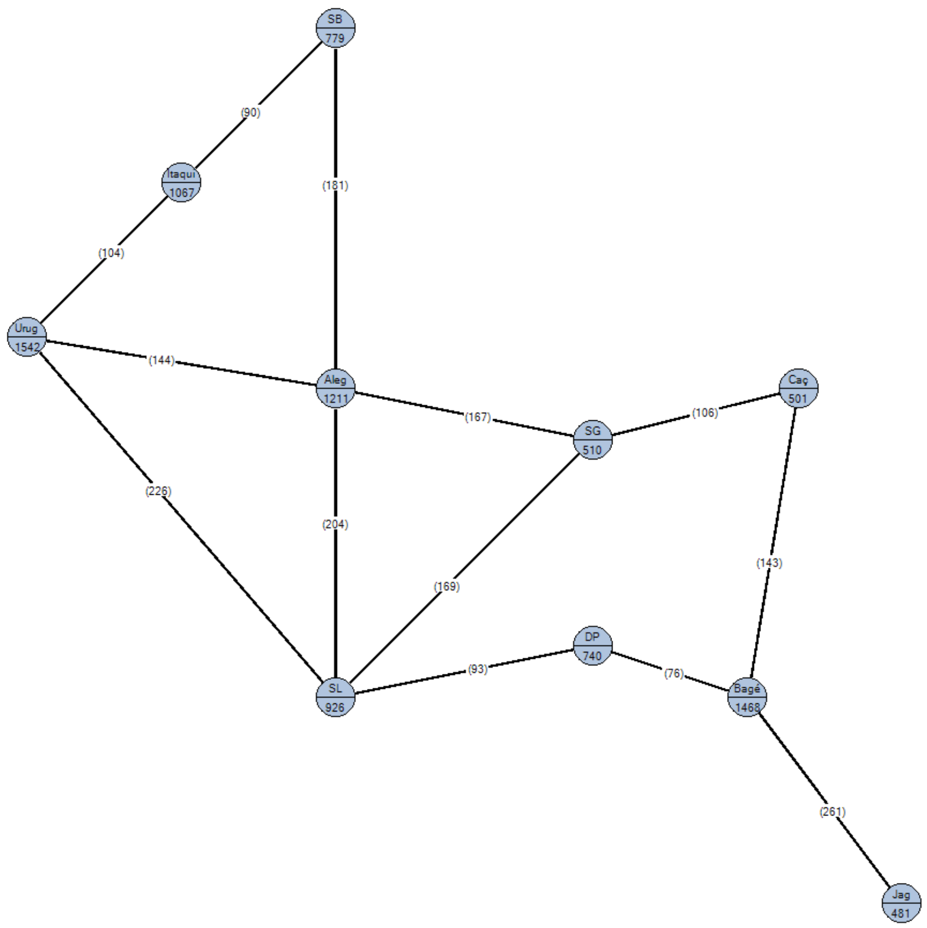
Fonte: PDI 2014/2018 - e-SIC

Figura 24 - Campus Bagé

Fonte: Google Maps

Desta forma, transferindo todos os dados acima para um grafo, priorizando as devidas localizações das cidades como vértices e as distâncias entre elas como arestas, através do Software Grafos v.1.3.5 podemos observar o grafo abaixo na Figura 25.

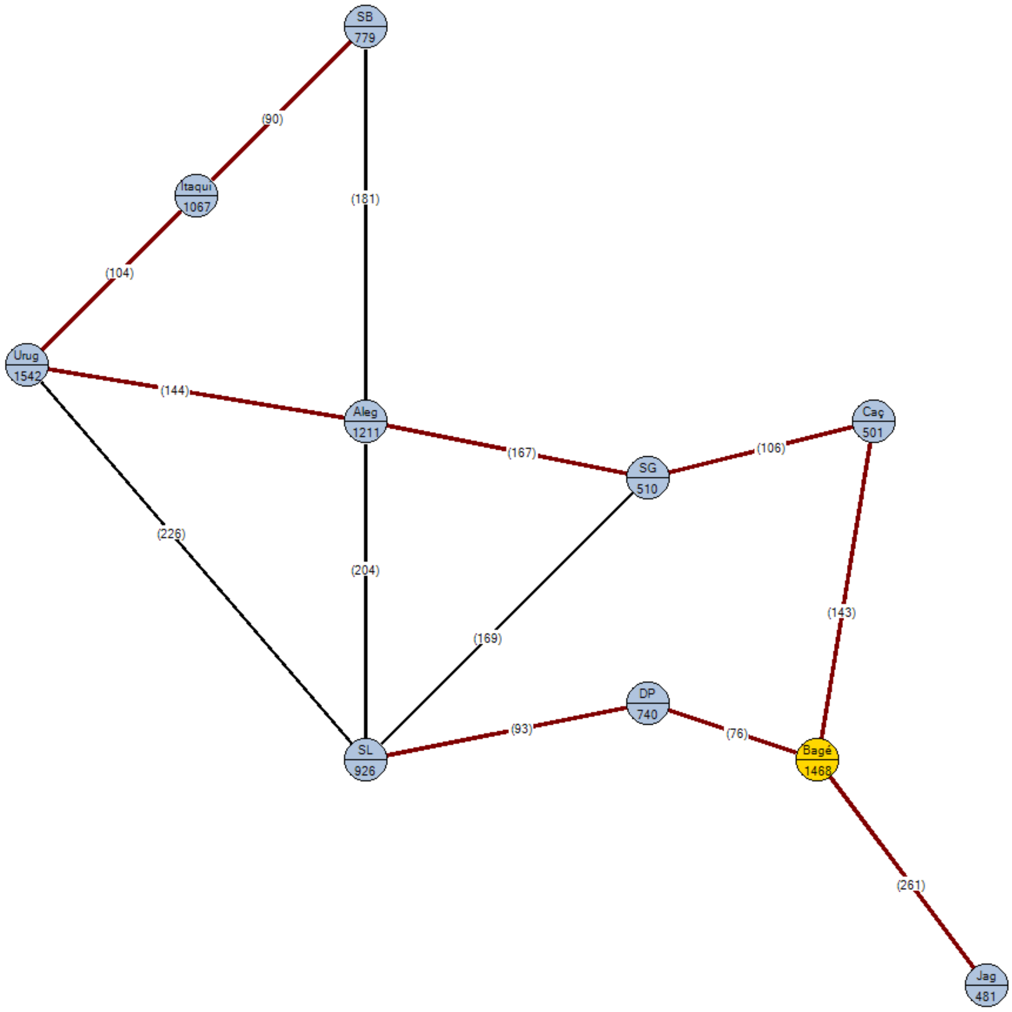
Figura 25 - Grafos com todos os dados.



Fonte: Autora (2019) - Software Grafos v.1.3.5

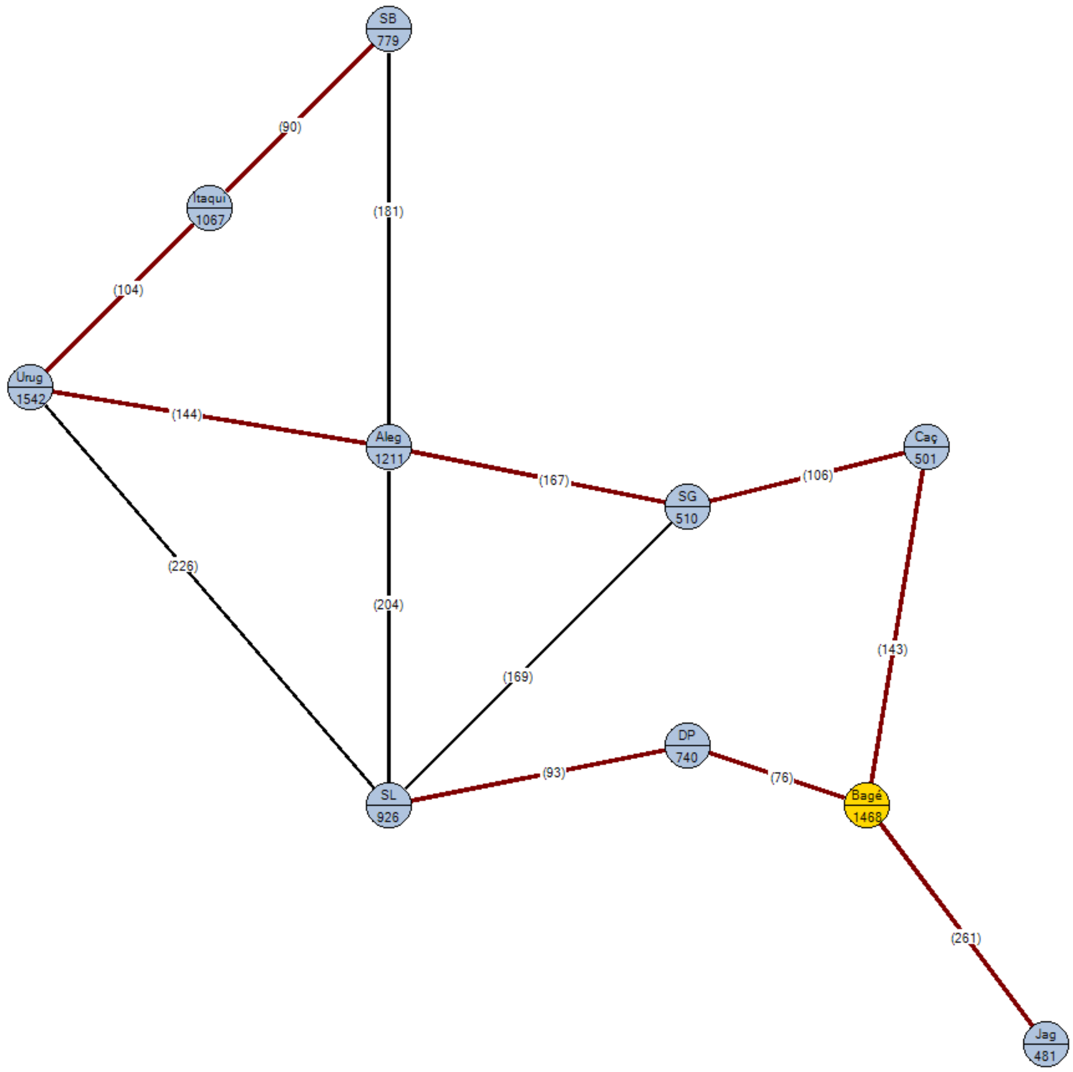
Pode-se lembrar que foi recorrido as estradas em boas condições para o trajeto dos transportes e desconsiderados os caminhos com estradas de chão. Através deste, pode-se realizar a aplicação dos Algoritmos de Prim e Kruskal, no qual podemos observar nas Figura 26 e 27, respectivamente para obter a árvore geradora mínima.

Figura 26 - Grafo gerado por Algoritmo de Prim



Fonte: Autora (2019) - Software Grafos v.1.3.5

Figura 27 - Grafo gerado com o Algoritmo de Kruskal



Fonte: Autora (2019) - Software Grafos v.1.3.5

Logo, com a aplicação dos dois algoritmos, consegue-se observar através da árvore gerada mínima com tracejado na cor bordô, que o total de quilômetros foi de $1184\ km$. Desta forma os caminhos seriam: São Borja → Itaqui → Uruguaiana → Alegrete → São Gabriel → Caçapava do Sul → Bagé; Santana do Livramento → Dom Pedrito → Bagé; Jaguarão → Bagé. Assim como visto na seção 5, está à disposição dos alunos os micro-ônibus nas cidades de Dom Pedrito, Itaqui, Jaguarão, São Borja, São Gabriel, Santana do Livramento, Uruguaiana e a cidade de Caçapava do Sul e Bagé possuem ônibus.

6.2 Campus Uruguaiana

O Campus Uruguaiana da Universidade Federal do Pampa é localizado na BR 472 – km 585, composta pelo total de 1542 alunos, oito cursos de graduação, quatro especializações, cinco mestrados e quatro doutorados. Por conta de sua estrutura e número de alunos foi um dos escolhidos para a escolha de ser possível sede do evento SIEPE, na qual pode-se observar na Tabela 13 e Figura 28.

Tabela 13 - Campus Uruguaiana

INFRAESTRUTURA (m^2)	NÚMERO DE ALUNOS	TOTAL DE ALUNOS A PARTIR DA 7º EDIÇÃO
$11.378{,}05$	1542	1642

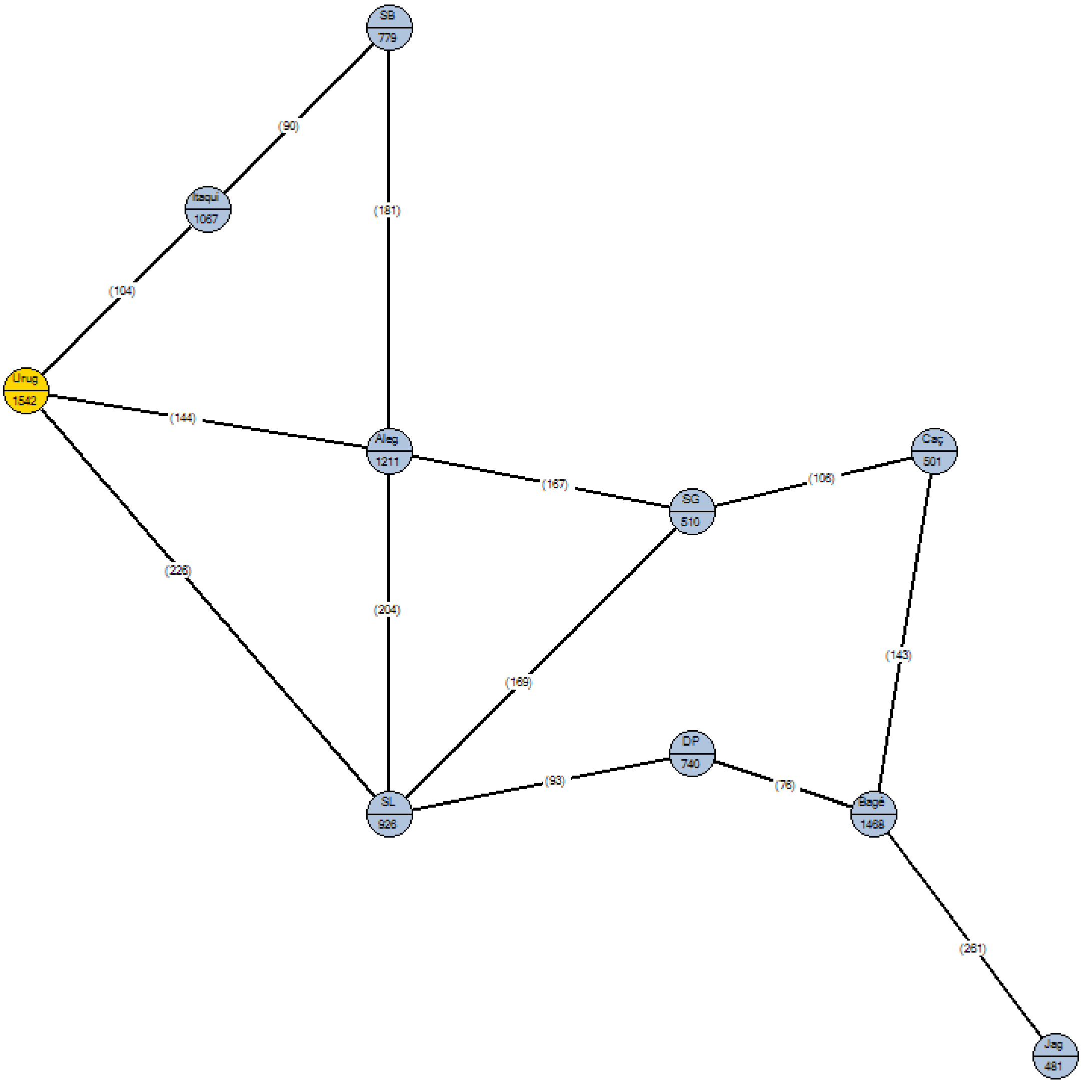
Fonte: PDI 2014/2018 - e-SIC

Figura 28 - Campus Uruguaiana

Fonte: Google Maps

Desta forma, transferindo todos os dados acima para um grafo no Software Grafos v.1.3.5 podemos observar o grafo abaixo na Figura 29.

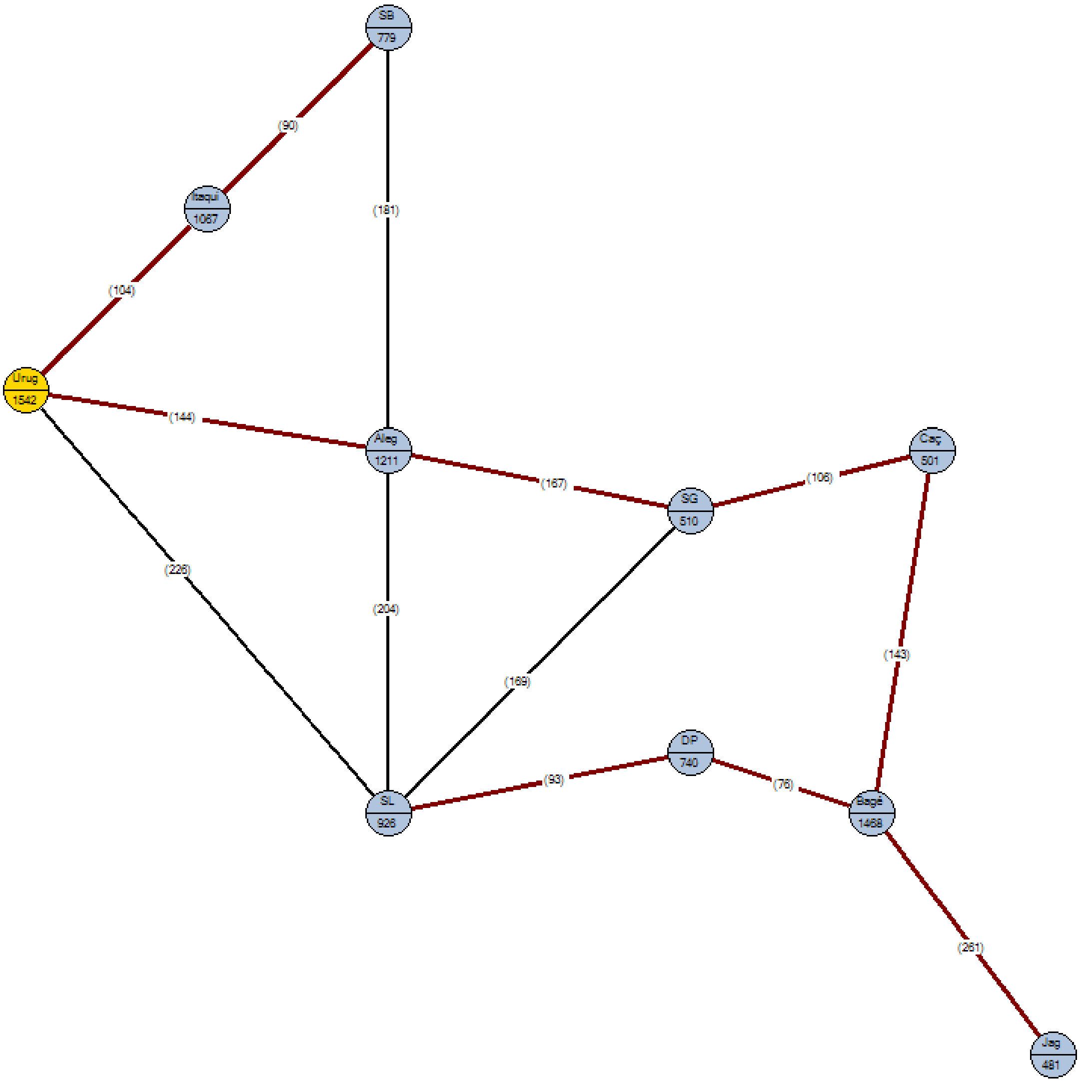
Figura 29 - Grafo com todos os dados



Fonte: Autora (2019) - Software Grafos v.1.3.5

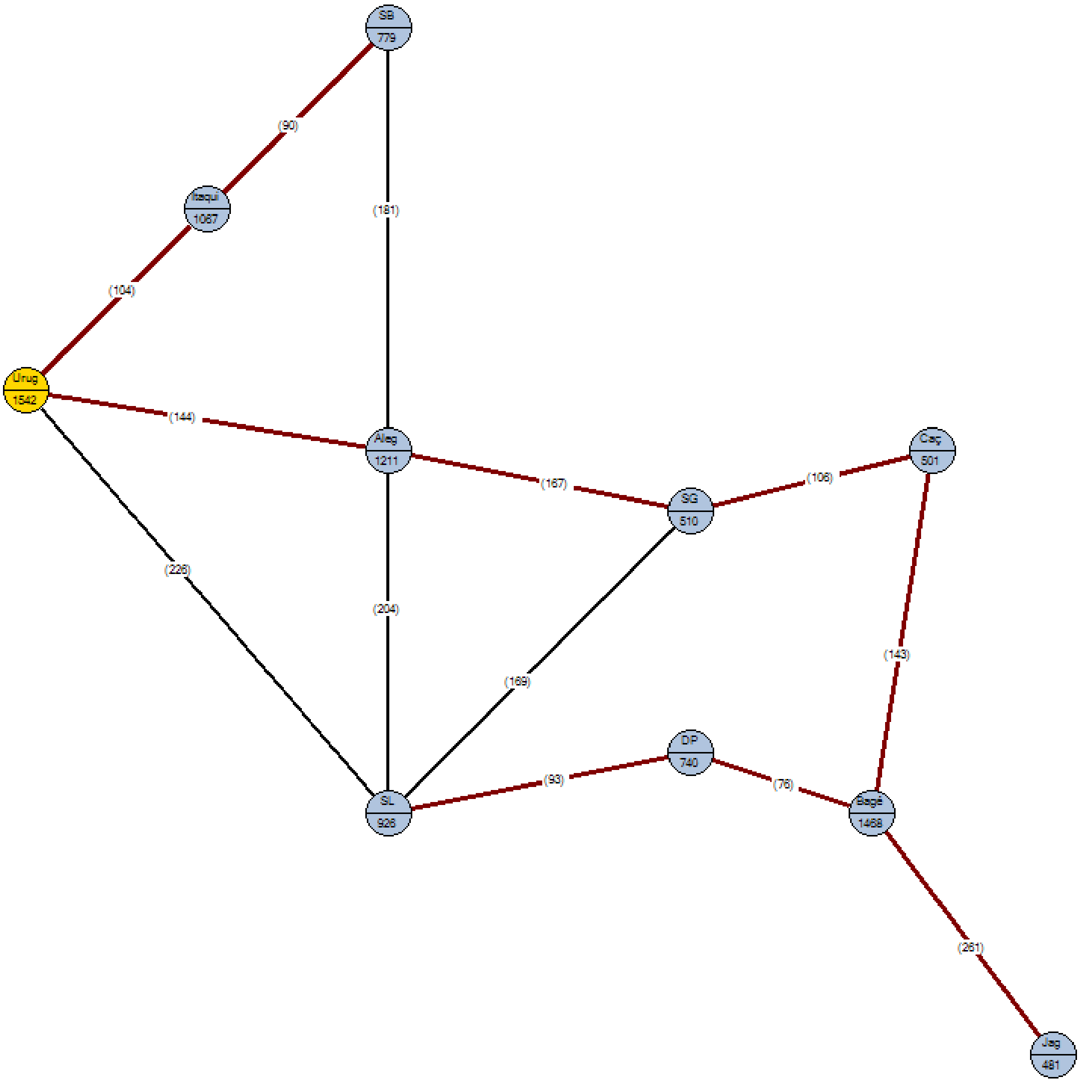
Este grafo também foi gerado através dos dados das pesquisas, em que foi recorrido as estradas em boas condições para o trajeto dos transportes. Através deste grafo gerado na Figura 27, pode-se realizar a aplicação dos Algoritmos de Prim e Kruskal, no qual podemos observar a árvore gerada mínima com tracejado na cor bordô em que mostra o menor caminho nas Figuras 30 e 31, respectivamente.

Figura 30 - Grafo gerado pelo Algoritmo de Prim



Fonte: Autora (2019) - Software Grafos v.1.3.5

Figura 31 - Grafo gerado pelo Algoritmo de Kruskal



Fonte: Autora (2019) - Software Grafos v.1.3.5

Assim com a aplicação dos dois Algoritmos, pode-se observar que o total de quilômetros também foi de $1184\,km$ resultando no mesmo caminho, definindo os caminhos como sendo: Jaguarão → Bagé; Santana do Livramento → Dom Pedrito → Bagé; Bagé → Caçapava do Sul → São Gabriel → Alegrete → Uruguaiana; São Borja → Itaqui → Uruguaiana.

É válido lembrar como citado nas seções 3.2.1 e 3.2.2 que a diferença dos dois algoritmos é que o Algoritmo de Prim encontra um subgrafo do grafo original no qual a soma total das arestas é minimizada e todos os vértices estão interligados e o Algoritmo de Kruskal encontra um subconjunto das arestas que forma uma árvore que inclui todos os vértices, onde o peso total é dado pela soma dos pesos das arestas da árvore.

Como informado na seção 5, o valor quantitativo de transportes que a UNIPAMPA possui não está disponível para acesso, mas deixa-se claro que os transportes passam pelos possíveis caminhos citados anteriormente recolhendo os alunos, desta maneira quando o ônibus já estiver com capacidade máxima, o próximo campus cede seu transporte para o restante do caminho, assim sucessivamente até a sede fixa do evento.

7. CONSIDERAÇÕES FINAIS

O presente trabalho teve como propósito encontrar o campus UNIPAMPA para o evento SIEPE que ao sediar obtivesse um menor custo logístico para a instituição, deste modo ocorreu a revisão bibliográfica da Teoria dos Grafos e o Problema de Fluxo de Redes, assim como a construção do modelo e o processo de busca de dados que, por fim pode-se realizar as construções dos grafos e aplicações dos algoritmos.

Priorizando encontrar o menor deslocamento acumulado entre todos os campi UNIPAMPA e enfatizado a infraestrutura dos campi, o número de alunos de cada campus, o número de participantes no evento, a informação em relação a frota de transportes da Universidade, todos os campi foram analisados e desta maneira os Campi Uruguaiana e Bagé se destacaram pela maior infraestrutura física e por possuírem maior quantidade de alunos e inscritos no evento.

Com isso, pode-se observar que através das pesquisas e dados encontrados no processo de busca, foi possível construir os grafos necessários para a aplicação dos algoritmos e definir o campus que pode sediar o evento gerando um custo operacional menor.

Os Algoritmos da Árvore Geradora Mínima determinaram o menor caminho entre todos os campi, ou seja, conecta todos os nós da rede, usando o menor comprimento total e os campi escolhidos por contar com uma melhor infraestrutura, bem como o maior número de alunos.

Desta forma pode-se concluir que tanto o Campus Uruguaiana quanto o Campus Bagé podem sediar o evento, por prevalecerem em relação a sua infraestrutura, no qual comportaria bem o evento SIEPE e por serem dois dos campi com maior número de alunos matriculados e inscritos ao evento, que neste caso não precisaria deslocar seus alunos, diminuindo o número de transportes.

Para este trabalho não foram levados em consideração alguns fatores que poderiam ser atribuídos na escolha do campus para o evento, assim para trabalhos futuros considera-se importante acrescentar a infraestrutura tecnológica, a infraestrutura das cidades de cada campus UNIPAMPA, assim como também seria importante a criação de um algoritmo específico para este problema.

REFERÊNCIAS BIBLIOGRÁFICAS

BUENO, J. **Algoritmos e Estrutura de Dados II – Introdução a Grafos**.[S.l] 2012.

CAIXEIRO. Disponível em: <http://www.mat.ufrgs.br/~portosil/caixeiro.html>. Acesso em 09 mai. 2009.

FEOFILOFF, P. **FLUXO EM REDES.** Instituto de Matemática e Estatística da Universidade de São Paulo. 2018.

GODOY, A. **Pesquisa Qualitativa – Tipos Fundamentais. Revista de Administração de Empresas.** São Paulo, v.35, n.3, p. 20-29.

GOODRICH, M. TAMASSIA, R. **Estrutura de dados e algoritmos em Java.** 4. ed. Porto Alegre: Bookman, 2007.

GOOGLE MAPS. Disponível em: < https://www.google.com.br/maps>. Acesso em 9 de nov. de 2019.

GÜNTHER, H. **Pesquisa Qualitativa Versus Pesquisa Quantitativa: Esta é a Questão?** Psicologia: Teoria e Pesquisa, 2006. Vol. 22 n. 2, p. 201-210.

GUTTOSKI, P. **Otimização de Consultas no PostgreSQL utilizando o Algoritmo de Kruskal.** Programa de Pós-Graduação em Informática Universidade Federal do Paraná. Curitiba. 2006.

J. F. Porto da Silveira. **O problema do caixeiro viajante.** 2009. Disponível em: <http://www.mat.ufrgs.br/~portosil/caixeiro.html>. Acesso em 19 de out. de 2019.

MÉNDEZ, Y. GUARDIA, L. **Problema de caminho mais curto – Algoritmo de Dijkstra.** Universidade Federal Fluminense. Rio de Janeiro, 2008. Disponível em: <https://www.marinha.mil.br/spolm/sites/www.marinha.mil.br.spolm/files/006_1.pdf> Acesso em: 17 de out. de 2019.

MOTA, B. **História da Matemática: Teoria dos Grafos.** Doutoramento em Ensino e Divulgação das Ciências. Faculdade de Ciências, Universidade do Porto. Portugal. [ca. 2013]

PEREIRA, G. CAMARA, M. **ALGUMAS APLICAÇÕES DA TEORIA DOS GRAFOS.** Universidade Federal de Uberlândia. FAMAT em Revista, 2008.

PINA, J. **Algoritmos para Grafos**. IME – USP. 2011. Disponível em: <https://www.ime.usp.br/~pf/algoritmos_para_grafos/aulas/mst.html>. Acesso em: 30 de out. de 2019.

PINTO, J. **Tópicos de Matemática Discreta**. Universidade de Aveiro. 1999.

QUITZAU, J. **Árvores Espalhadas Mínimas. Complexidade de Algoritmos I.** Disponível em: <http://www.ic.unicamp.br/~meidanis/courses/mo417/2003s1/aulas/2003-05-16.html>. Acesso em 19 de out. de 2019.

RBHM. **EULER, UM MATEMÁTICO MULTIFACETADO.** Revista Brasileira de História da Matemática - Vol. 9 abr./2009 – set./2009.

RIZZI, R. VICENTE, A. **Uma aplicação de grafos a um problema agrícola, envolvendo distribuição de água e transportes.** Engenharia na agricultura, Viçosa/MG, Vol.19 n.3. Maio/Junho 2011. Disponível em: <https://periodicos.ufv.br/ojs/index.php/reveng/article/view/217>. Acesso em: 19 de out. de 2019.

SCHEEREN, V.; FONTOURA, T. P.; VAZ, F. A.; PEREIRA, E. D. . **PROGRAMAÇÃO DE HORÁRIOS DE EXAMES DE CURSOS UNIVERSITÁRIOS ATRAVÉS DA COLORAÇÃO DE GRAFOS.** Revista da Jornada da Pós-Graduação e Pesquisa Congrega, v. Ãºnico, p. 367-382, 2017.

SELONG, L; KRIPKA, R. **Otimização de roteiros: Estudo de caso de uma distribuidora de ferro de Passo Fundo/RS para a região.** Instituto de Ciências Exatas e Geociências, Universidade de Passo Fundo. Revista Ciatec – UPF. Vol. 1, p.p.14-31, 2009. Disponível em: <http://seer.upf.br/index.php/ciatec/article/view/613/412>. Acesso em: 07 de set. de 2019.

SILVA, J.GOMES, E.FRIGIERI, E. **Controle da Topologia de Redes de Sensores sem Fio para Economia de Energia Baseado no Algoritmo de Prim.** Revista de Tecnológia da Informação e Comunicação, Vol. 6, NO. 1, 2016.

SOFTWARE GRAFOS V.1.3.5. Disponivel em: <http://arodrigu.webs.upv.es/grafos/doku.php>. Acesso em: 31 de out. de 2019.

SOUZA, R. **Algoritmos para o problema da árvore geradora mínima probabilística**. Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais. Belo Horizonte. 2010

SOUZA, Rafael. SILVA, Alexsandro. SILVA, Wilamis. SANTOS Rennê. NETO, Francisco. **Uma abordagem paralela do Algoritmo de Floyd para solução do problema do caminho mínimo.** Programa de Pós-Graduação em Ciências da Computação – UERN/UFERSA. Porto de Galinhas/PE, 2015. Disponivel em: <http://www.din.uem.br/sbpo/sbpo2015/pdf/142081.pdf>. Acesso em: 17 de out. de 2019.

STEINER, Maria. COSTA, Clarice. COSTA, Deise. ANDRETTA. Élsio. ZAMBENEDETTI, Vóldi. **Técnicas da pesquisa operacional aplicadas à logística de atendimento aos usuários de uma rede de distribuição de energia elétrica.** Programa de Pós-Graduação em Métodos Numéricos em Engenharia. Curitiba, Paraná. 2006.

TAHA, H. **PESQUISA OPERACIONAL: Uma visão geral.** 8 ed. São Paulo: Pearson Prentice Hall, 2008.

UNIVERSIDADE FEDERAL DO PAMPA. Disponível em: <https://unipampa.edu.br/portal/>. Acesso em: 31 de out. de 2019.