

1.03.04 - Ciência da Computação / Sistemas de Computação.

## GERAÇÃO DE QUESTÕES DE LÓGICA BASEADA NA TÉCNICA DE GERAÇÃO POR ESBOÇO

Filipe da Silva Oliveira<sup>1,2\*</sup>, Elthon Allex da Silva Oliveira<sup>3</sup>

1. Estudante de Ciência da Computação na Universidade Federal de Alagoas (UFAL) - *Campus Arapiraca*
2. Bolsista PIBIC/CNPq-Af do Edital Propep/UFAL 2020-2021
3. Professor da UFAL *Campus Arapiraca* - Ciência da Computação/Orientador

### Resumo

A elaboração de problemas com características específicas da disciplina de Lógica é tida como uma tarefa tediosa, principalmente quando há a necessidade por problemas únicos para cada estudante. Neste trabalho é apresentada uma abordagem para geração automática de fórmulas e argumentos da Lógica Proposicional. Para possibilitar a geração automática mantendo características comuns, foi feito uso da técnica conhecida como Geração por Esboço, advinda da área de Síntese de Programas. Como prova de conceitos, foi desenvolvido um sintetizador capaz de gerar as fórmulas e os argumentos a partir de informações chave fornecidas pelo usuário. A abordagem utilizada gerou excelentes resultados na síntese de fórmulas. Entretanto, apesar de alcançada, a geração de argumentos ainda carece de melhorias.

**Autorização legal:** Não se aplica.

**Palavras-chave:** Síntese de Fórmulas; Síntese de Argumentos; Lógica Proposicional.

**Apoio financeiro:** Programa de Bolsa de Iniciação Científica - PIBIC/UFAL.

**Trabalho selecionado para a JNIC:** Pró-reitoria de Pesquisa e Pós-Graduação - Propep/UFAL.

### Introdução

Síntese de Programas é a tarefa de encontrar automaticamente um programa numa determinada linguagem de programação que satisfaça a intenção do usuário descrita na forma de alguma especificação. Este problema tem sido considerado o santo graal da Ciência da Computação. Contudo, atualmente dois problemas se destacam nesta área: a ambiguidade na descrição da intenção do usuário, e um espaço de pesquisa de programas quase sempre enorme. Apesar destes desafios inerentes ao problema, pesquisadores no campo de Síntese de Programas têm desenvolvido muitas técnicas diferentes que permitem sua aplicação em diversos domínios de aplicação da vida real.

Dentre os domínios de aplicação, destacam-se engenharia de software, descoberta biológica, programação de usuário final e educação assistida por computador. Este projeto de pesquisa foca neste último, mais especificamente, na síntese de elementos da Lógica Proposicional. Esses elementos podem ser usados como problemas na educação formal de alunos em disciplinas de Lógica. Sabe-se que gerar novos problemas que tenham características específicas de solução (como um determinado nível de dificuldade ou que envolva o uso de um determinado conjunto de conceitos) é uma tarefa tediosa para o professor. Mais claramente, foi utilizado no projeto a abordagem de Geração de Esboço para conseguir sintetizar os elementos da Lógica Proposicional. A ideia central deste projeto foi investigar a área de Síntese de Programas e aplicá-la. Para isso, foi abordado e resolvido um problema real: geração de problemas para disciplinas de Lógica. Este cenário foi o meio para a investigação e o entendimento dos problemas envolvidos, e soluções existentes, na área de pesquisa em questão.

Como objetivo principal deste trabalho teve-se a síntese de fórmulas e argumentos da Lógica Proposicional, utilizando Síntese de Programas. Como objetivos específicos: (i) entender em detalhes o potencial de aplicação do campo de Síntese de Programas; (ii) entender em detalhes os problemas envolvidos no campo de Síntese de Programas; (iii) entender em detalhes os problemas na abordagem de Programação por Exemplos (PBE, em inglês) e as técnicas usadas no tratamento destes problemas; (iv) por último, implementar o sintetizador de fórmulas e argumentos da Lógica Proposicional.

### Metodologia

A metodologia adotada consistiu na execução das etapas descritas a seguir:

**Síntese de Programas.** Nesta etapa, foi realizado um estudo de maneira horizontal da área de Síntese de Programas usando como base o trabalho publicado em [1]. Foi tomado conhecimento sobre todos os subcampos existentes identificando informações, tais como: áreas de aplicação, técnicas usadas, problemas e limitações. Foi construído mapas conceituais explicitando as relações entre subcampos, abordagens, técnicas e problemas dentro do campo de Síntese de Programas.

**Levantamento bibliográfico.** Nesta etapa teve-se o objetivo de conhecer trabalhos que façam uso dos conceitos de Síntese de Programas, dando mais atenção aos trabalhos publicados nos últimos cinco anos. Além disso, foi feito um levantamento bibliográfico sobre geração de problemas/questões, objetivando conhecer quais técnicas têm sido aplicadas e identificar possíveis lacunas neste tipo de problema. Trabalhos foram encontrados no decorrer das demais etapas. Assim, o levantamento bibliográfico foi realizado ao longo da execução deste projeto.

**Programação por Exemplos.** Nesta etapa tomou-se conhecimento de maneira mais profunda os problemas existentes nesta abordagem. Foi estudado as técnicas utilizadas no tratamento destes problemas. Ao final desta etapa, foi realizada uma análise comparativa das técnicas utilizadas em PBE. Aqui, identificou-se também que tal técnica não era a ideal atualmente para ser aplicada na síntese dos elementos da Lógica Proposicional. Então, foi decidido utilizar a técnica descrita no próximo item.

**Geração de Esboço.** Como dito no item anterior, foi tido por utilizar esta técnica para sintetizar os elementos propostos nesta pesquisa. Então, foi realizado um estudo vertical desta técnica, observando vantagens e desvantagens, assim como também as restrições de utilização. A partir disso, foi feita uma adaptação na técnica para que pudesse ser utilizada na síntese dos elementos.

**Revisão dos conceitos de Lógica.** Considerando o objetivo do sintetizador a ser implementado, nesta etapa, foi revisado os conceitos de Lógica envolvidos: fórmula bem formada, tabela verdade, formas normais conjuntiva e disjuntiva, tableau semântico e dedução natural.

**Estudo aprofundado das técnicas adequadas ao problema envolvido.** Nesta etapa, de acordo com o que foi descoberto na etapa Programação por Exemplos, Geração de esboço e Revisão dos conceitos de Lógica foi eleito as técnicas, os algoritmos e as estruturas mais adequados à síntese proposta neste trabalho.

**Desenvolvimento do sintetizador.** O desenvolvimento consistiu de três sub etapas: (i) desenho do sintetizador - definição da arquitetura da ferramenta; (ii) implementação; (iii) verificação e validação - realização de testes e as saídas produzidas inicialmente validadas junto ao coordenador deste projeto, professor da disciplina de Lógica Aplicada à Computação.

## Resultados e Discussão

### Resultados:

**Estudo sobre Síntese de Programas.** Foi utilizado como base o trabalho em [1] para realização do estudo. Além disso, o estudo foi realizado com outros trabalhos científicos, para que fosse possível tomar conhecimento também sobre as ferramentas utilizadas pelos pesquisadores e desenvolvedores na área, tais como: arcabouço PROSE, ROSETTE e SKETCH.

**Estudo sobre Programação por Exemplos.** O estudo aconteceu tomando como base os trabalhos [1], [2], [3] e [4].

**Estudo sobre Geração de Esboço.** Devido a não adequação da técnica de PBE com o problema da pesquisa, foi realizado um estudo para utilização da abordagem de Geração por Esboço, tendo como base os trabalhos [5] e [6].

**Levantamento bibliográfico.** O levantamento bibliográfico aconteceu durante todo o período do projeto, tomando-se conhecimento dos principais trabalhos, ferramentas e abordagens de Síntese de Programas.

**Revisão dos conceitos de lógica.** O estudo consistiu na leitura dos conceitos e na resolução de exercícios. Os conceitos revisados foram: fórmula bem formada (fbf), tabela verdade, regras de inferência, argumentos válidos e inválidos, fórmula normal conjuntiva e disjuntiva (FNC e FND), tableau semântico e dedução natural.

**Estudo aprofundado para escolha das técnicas a serem utilizadas.** Nesta etapa, aconteceu um estudo das principais técnicas, escolha e adequação para o problema tratado neste trabalho.

**Desenvolvimento do sintetizador.** O desenvolvimento do sintetizador de elementos da Lógica Proposicional aconteceu conforme os passos citados na seção de metodologia. A ferramenta foi desenvolvida na linguagem de Programação Python, e nessa utilizou-se os conceitos de Geração por Esboço e uma versão do oráculo (adaptação), técnicas essas apresentadas em [5] e [6].

**Produção científica.** Neste ciclo do projeto nós escrevemos um artigo apresentando cerca de 50% dos resultados da pesquisa. Este artigo foi enviado para o Workshop de Pesquisa - PESQBASE da Escola Regional de Computação Bahia-Alagoas-Sergipe. O artigo apresenta em seus resultados a geração das fórmulas da

Lógica Proposicional. O trabalho foi aprovado para apresentação e publicação nos anais do evento. Além disso, já foi iniciado a escrita de um outro artigo apresentando o restante dos resultados. Ou seja, a geração automática dos argumentos da Lógica Proposicional, que depende da geração das fórmulas.

As Figuras 1, 2 e 3, mostram exemplos de elementos gerados automaticamente com o sintetizador. Nas imagens observa-se que a interface da ferramenta até o fim do ciclo tinha o formato de texto e executado no console do computador.

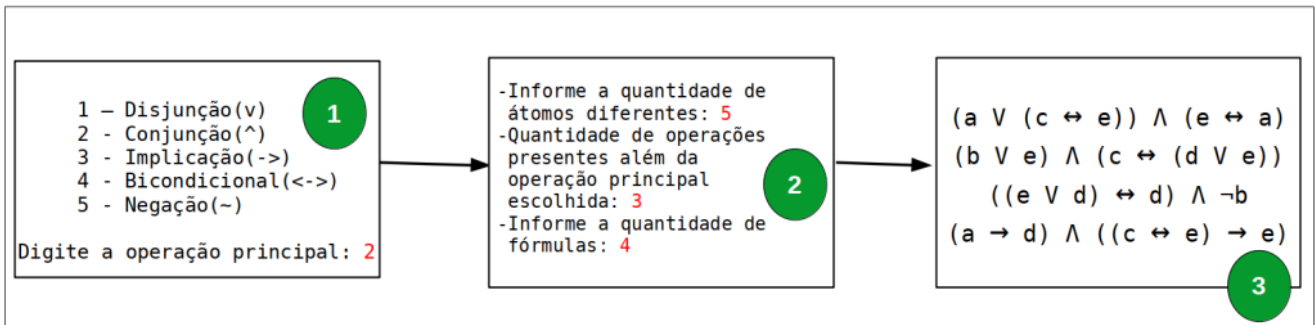


Figura 1: Geração de fórmulas onde a conjunção é o conectivo principal.

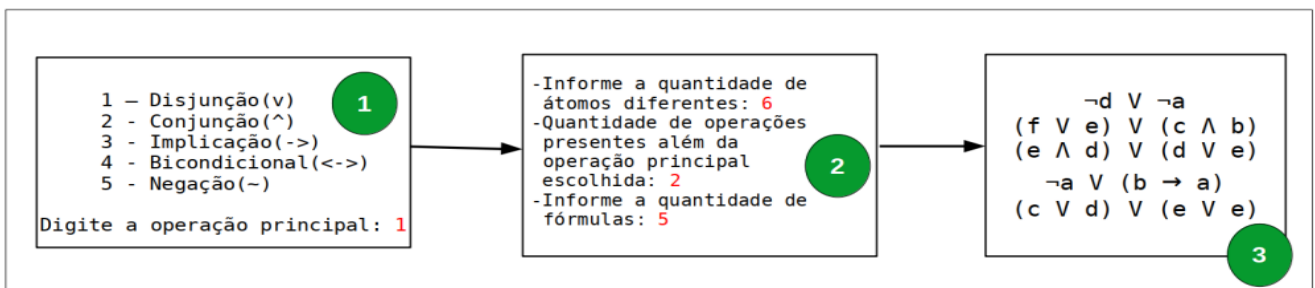


Figura 2: Geração de fórmulas onde a disjunção é o conectivo principal.

Nas Figuras 1 e 2 o comportamento da síntese é o mesmo. Em (1) o usuário escolhe o conectivo principal que as fórmulas irão conter. Em (2) escolhe a quantidade de átomos presente no processo de síntese, a quantidade de operações além da principal e a quantidade de fórmulas. Por fim, em (3) as fórmulas são apresentadas.

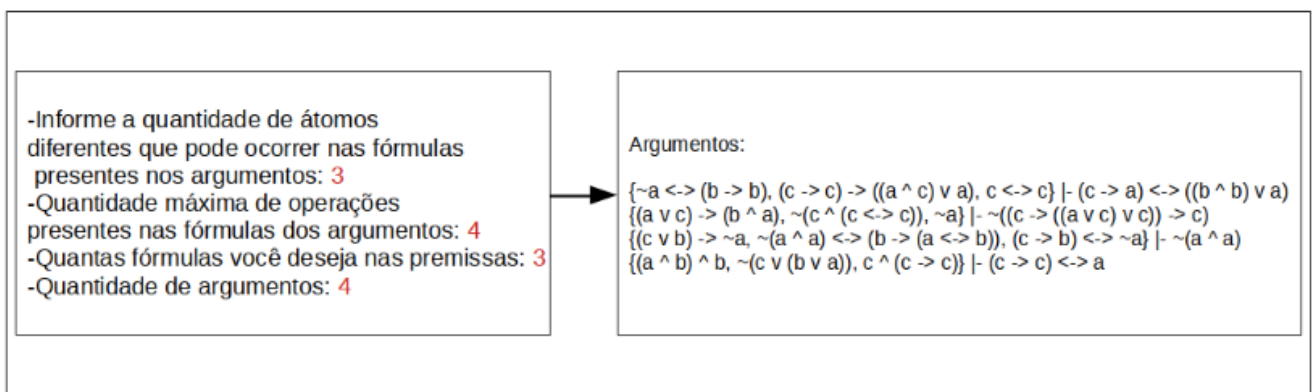


Figura 3: Síntese de argumentos

Na Figura 3 podemos observar a síntese de argumentos também parametrizável. Essa abordagem solicita quantidade de átomos presentes na síntese, quantidade máxima de operações, quantidade de premissas, e quantidade de argumentos e, os argumentos são gerados automaticamente. Essa parte da pesquisa até o momento da finalização do ciclo ainda não estava sendo realizada com eficiência, pois dado que os elementos são gerados de maneira aleatória, na maioria dos casos os argumentos não válidos

#### Discussão:

Como dito anteriormente, com o entendimento de algumas técnicas da abordagem de PBE, observou-se que tal segmento não seria o mais apropriado para o contexto atual da pesquisa. Com isso, decidiu-se então por utilizar a abordagem de Geração de esboço. Este processo de descoberta ocasionou um pequeno atraso na etapa do desenvolvimento do sintetizador. A abordagem de PBE é rica em conteúdos, o que possibilita a realização de um estudo considerável. Então, no próximo ciclo que já iniciou, a pesquisa está tendo foco também na abordagem de PBE.

Estudar Síntese de Programas desde o início do projeto foi um desafio. Uma área não tão nova, mas muito desconhecida. A Síntese de Programas aborda conceitos conhecidos por um estudante formado em ciência da computação, conceitos estes que muitas vezes passam despercebidos na sala de aula. O responsável pelo plano de trabalho muitas vezes não tinha conhecimento de tais conceitos, por ainda não ter estudado as disciplinas necessárias, então era preciso revisar e aprender esses conceitos.

A síntese de argumentos ainda apresenta alguns problemas. Devido à aleatoriedade no processo de síntese, os argumentos gerados são quase sempre logicamente inválidos. Ou seja, a conclusão não decorre do conjunto de premissas. Para contornar este problema, foi pensado em fazer uso de esquemas, baseados nas regras de inferência da dedução natural, no código fonte para garantir a validade por construção. Esta proposta de resolução está sendo desenvolvida e espera-se ter resultados promissores.

## Conclusões

O projeto chegou ao fim do ciclo tendo obtido o que fora proposto: geração de elementos da Lógica Proposicional por meio de técnicas de Síntese de Programas. Tal geração foi viabilizada pela implementação de um sintetizador desenvolvido na linguagem de programação Python. Como próximo passo no desenvolvimento desta pesquisa, além de aprimorar os resultados atualmente alcançados, é pretendida a construção de uma interface gráfica para o sintetizador, provavelmente na forma de um aplicativo móvel.

A pesquisa atende uma necessidade real, que é a geração automática de questões de Lógica Proposicional. Porém, sua importância também está no fato de que esta necessidade está sendo atendida através de uma abordagem nova no campo científico, que é a Síntese de Programas. Em um futuro próximo a Síntese de Programas estará resolvendo problemas que antes não se imaginava uma resolução. Para isso, é preciso iniciar a síntese de tarefas relativamente simples para que no futuro possamos solucionar problemas mais complexos, que é pretensão da equipe responsável pelo desenvolvimento desta pesquisa.

Durante o desenvolvimento desta pesquisa, foi produzida uma quantidade considerável de documentação sobre a área. Pode-se citar mapas conceituais, propostas de arquiteturas para o sintetizador, assim como um dicionário contendo conceitos essenciais para o entendimento da abordagem, resumos de artigos, livros e alguns testes de técnicas de síntese.

Para o responsável que trabalhou no plano de trabalho atual, a pesquisa obteve diversos resultados positivos. De um modo geral, conseguiu-se concluir a proposta geral da pesquisa, que resume-se em estudar de forma simplificada a área de Síntese de Programas e desenvolver o sintetizador supracitado. A pesquisa contribuiu diretamente à formação acadêmica do estudante, levando-o a aprender conceitos, técnicas e abordagens de estudo dos mais diversos tipos. Somado a isso, é importante ressaltar também que a estrutura estável do ambiente de trabalho junto à relação com o orientador, contribuíram fortemente para o aprendizado dos conteúdos, assim como também o protagonismo num novo ciclo deste projeto e evolução dos resultados do ciclo atual.

## Referências bibliográficas

- [1]. GULWANI, S., POLOZOV, O., & SINGH, R. (2017). Program synthesis. *Foundations and Trends® in Programming Languages*, v. 4, n. 1-2, p. 1-119, 2017.
- [2]. GULWANI, Sumit. *Programming by Examples (and its Applications in Data Wrangling)*. Verification and Synthesis of Correct and Secure Systems, 2016.
- [3]. GULWANI, Sumit. Programming by examples: Applications, algorithms, and ambiguity resolution. In: *International Joint Conference on Automated Reasoning*. Springer, Cham, 2016. p. 9-14.
- [4]. GULWANI, Sumit. Automating string processing in spreadsheets using input-output examples. *ACM Sigplan Notices*, v. 46, n. 1, p. 317-330, 2011.
- [5]. SOLAR-LEZAMA, Armando. The sketching approach to program synthesis. In: *Asian Symposium on Programming Languages and Systems*. Springer, Berlin, Heidelberg, 2009. p. 4-13.
- [6]. SOLAR-LEZAMA, Armando; BODIK, Rastislav. *Program synthesis by sketching*. University of California, Berkeley, 2008.