

Linguagem Visual para Resolução de Problemas Fundamentada no Pensamento Computacional: uma proposta*

Adriana Bordini¹, Simone André da Costa Cavalheiro¹,
Christiano Ávila¹, Luciana Foss¹

¹Programa de Pós-Graduação em Computação – Universidade Federal de Pelotas (UFPel)
96.010-610 – Pelotas – RS – Brasil

{adriana.bordini, simone.costa, lfoss}@inf.ufpel.edu.br

Abstract. *The goal of this proposal is to provide a methodology based on Computer techniques for the solution of problems of any area of knowledge that can be applied from Basic Education. We present the core of a visual language that allows the specification of the input/output information of a problem, as well as its solution through the concepts of abstraction and decomposition.*

Resumo. *O objetivo desta proposta é fornecer uma metodologia fundamentada em técnicas da Computação para a solução de problemas de qualquer área do conhecimento, que possa ser aplicada a partir do Ensino Básico. Apresenta-se o núcleo de uma linguagem visual que permite a especificação das informações de entrada/saída de um problema, bem como a sua solução por meio dos conceitos de abstração e decomposição.*

1. Introdução

As tecnologias inseridas em nossa vida cotidiana e em qualquer área do conhecimento mudam as formas de pensar e de resolver problemas. Consequentemente essas mudanças também alteram as habilidades necessárias para a educação do século XXI. Além do conhecimento de conteúdos a serem inseridos de forma interdisciplinar, tais como: matemática, geografia, idiomas, consciência global e cultural, alfabetização financeira, alfabetização ambiental, entre outros; estudos concordam em algumas áreas críticas a serem desenvolvidas, tais como: criatividade e imaginação, pensamento crítico, resolução de problemas, comunicação oral e escrita, colaboração e trabalho em equipe, alfabetização tecnológica, iniciativa, responsabilidade, entre outros; as quais são consideradas essenciais para que os alunos fiquem preparados para o futuro [P21.org 2018, Experience 2018].

Diversos autores concordam [Tabesh 2017, Mohaghegh and McCauley 2016, Yadav et al. 2016] que muitas das habilidades acima citadas podem ser desenvolvidas por meio do Pensamento Computacional (PC). Jeannette Wing [Wing 2006] descreveu o PC como um processo de resolução de problemas, fundamentado na Ciência da Computação (CC), capaz de promover as competências de abstração, decomposição, automação, análise, entre outras, e que podem ser aplicadas em qualquer área do conhecimento [Wing 2006]. Em particular, a *Computer Science Teachers Association* (CSTA) e a *International Society for Technology in Education* (ISTE) junto com colaboradores propuseram a definição operacional do PC, delineando seu escopo e dimensões [CSTA 2017].

*Projeto realizado com o apoio da PREC e PRPPG / UFPel.

Na Base Nacional Comum Curricular (BNCC) [Brasil 2018] muitos destes conhecimentos e habilidades são listados e considerados essenciais para o estudante brasileiro, para que possa se desenvolver como pessoa, se preparar para o exercício da cidadania e se qualificar para o trabalho. Na última versão da BNCC, dez competências são listadas, incluindo, resolução de problemas e comunicação.

Diversos trabalhos propuseram a introdução de conceitos da Computação na educação, bem como o desenvolvimento de habilidades do PC em seus diferentes níveis. Pesquisas já relatam as diferentes estratégias adotadas para este fim na Educação Básica [Marques et al. 2017, Bordini et al. 2016a, Martins et al. 2016, Mota et al. 2014], bem como descrevem os desdobramentos do PC [Bordini et al. 2016b, Oliveira et al. 2016, Andrade et al. 2013] no Brasil e no exterior [Bordini et al. 2017].

Diante dos recursos tecnológicos e a influência de diferentes estímulos multimídias, percebe-se que as habilidades destacadas como importantes para a educação do século XXI precisam ser trabalhadas de forma diferente do ensino tradicional. Dado que mudando o contexto de aprendizagem, deve-se mudar a forma de ensinar, é necessário que novas metodologias de ensino/aprendizagem sejam propostas.

Este trabalho propõe uma linguagem visual para resolução de problemas fundamentado no PC, com o objetivo de guiar estudantes a construírem as suas soluções. Optou-se por uma linguagem visual, por esta ser uma representação mais próxima do dia a dia dos estudantes. Segundo [Martins et al. 2005], as linguagens visuais “são mais facilmente lembradas do que suas correspondentes representações verbais”.

O texto está organizado como segue. A Seção 2 apresenta conceitos introdutórios do PC. Na Seção 3 descreve-se o núcleo da linguagem visual proposta. Na Seção 4 é delineada a metodologia de uso da linguagem e na Seção 5 mostram-se os resultados de um estudo de caso. Por fim, na Seção 6, são apresentadas as considerações finais.

2. Pensamento Computacional

O PC foi definido como um processo de resolução de problemas, que inclui (mas não está limitado) as seguintes características/habilidades [CSTA 2017]: formulação de problemas de uma forma que seja possível usar um computador e outras ferramentas para ajudar a resolvê-los; organização lógica e análise de dados; representação de dados através de abstrações, como modelos e simulações; automatizações de soluções através do pensamento algorítmico (uma série de passos ordenados); identificação, análise e implementação de soluções possíveis com o objetivo de alcançar a combinação mais eficiente e eficaz das medidas e recursos; generalização e transferência desse processo de resolução de problemas para uma grande variedade de problemas.

As dimensões essenciais do PC, incluem as seguintes disposições [CSTA 2017]: confiança em lidar com a complexidade; persistência em trabalhar com problemas difíceis; tolerância para a ambiguidade; capacidade de lidar com os problemas em aberto; capacidade de comunicar e trabalhar com outros para atingir um objetivo comum.

Além disso, foram estabelecidos 9 conceitos (estrutura e vocabulário), que estão relacionados com os conceitos da Ciência da Computação e são considerados importantes de serem trabalhados para o desenvolvimento do PC [CSTA 2017]: coleta de dados - o processo de reunir informação apropriadas; análise de dados - dar sentido a dados, encon-

trar padrões e tirar conclusões; representação de dados - descrever e organizar dados em gráficos, mapas, palavras ou imagens apropriadas; decomposição do problema - quebrar as tarefas em partes menores gerenciáveis; abstração - reduzir a complexidade para definir a ideia principal; algoritmos e procedimentos - resolver um problema por meio de uma série de passos ordenados; automação - usar máquinas para fazerem tarefas repetitivas ou tediosas; simulação - executar experimentos usando modelos; paralelização - organizar recursos para, simultaneamente, realizar tarefas para alcançar um objetivo comum.

Muitas destas habilidades destacadas pela [CSTA 2017] também são consideradas importantes para o século XXI. E o desafio é trabalhar essas habilidades de forma atraente para o aluno. Observando isso, pensou-se em propor uma metodologia para resolução de problemas por meio de uma linguagem visual, ferramenta que é utilizada em todos os contextos que cercam o aluno fora da sala de aula.

A linguagem visual proposta fundamenta-se nos conceitos de abstração e decomposição. Abstração em [Wing 2008], “é o processo de decidir quais detalhes precisamos destacar e que detalhes podemos ignorar - subjaz pensamento computacional”. Por exemplo, tendo um problema para solucionar, divide-se, foca-se em alguns pontos que são considerados mais importantes, isolando as etapas que levam a sua solução em partes menores, de forma a simplificá-lo [Papert 1994]. “Abstrações são as ferramentas ‘mentais’ da computação” [Wing 2008], que ajudam na resolução de problemas. A abstração da CC é mais rica e complexa que a da matemática e das ciências físicas. Pois as abstrações da CC “... são extremamente gerais, porque elas são simbólicas, onde abstrações numéricas são apenas um caso especial” [Wing 2008]. Na computação, o processo de abstração é feito em camadas, num trabalho “... simultâneo com pelo menos duas camadas de abstração: a camada de interesse e a camada inferior; ou a camada de interesse e a camada superior” [Wing 2008]. A abstração em camadas ajuda a decompor o problema em pequenas partes e trabalhar com estas partes menores de forma independente do todo, o que possibilita a re-utilização da solução para outros processos.

A decomposição, para *Computing At School* (CAS) [CAS 2015], “é uma maneira de pensar sobre artefatos em termos de suas partes. As partes podem então ser compreendidas, resolvidas, desenvolvidas e avaliadas separadamente. Isso torna os problemas complexos mais fáceis de resolver, entender melhor novas situações e mais fácil de projetar sistemas maiores”. Assim na decomposição se quebra as tarefas em partes menores para melhor se gerenciar o todo. Por exemplo, fazer um bolo com cobertura pode-se dividir e decompor em atividades separadas, como: fazer a massa; fazer a cobertura; montar o bolo; onde cada uma dessas atividades, também podem ser divididas em um subconjunto de etapas. “Através da decomposição da tarefa original, cada parte pode ser desenvolvida e integrada posteriormente no processo” [CAS 2015].

3. Linguagem Visual para o desenvolvimento do PC

Assim como as habilidades e competências necessárias para a resolução de problemas vêm se modificando ao longo do tempo, as linguagens utilizadas para a especificação das soluções também. Desde a pré-história até os dias atuais a linguagem é utilizada para a comunicação. E aos poucos os meios de comunicação, tais como: jornal, rádio, televisão, telefonia fixa e móvel tiveram uma influência universal sobre o caráter das línguas [Crystal 2005]. Sendo a Internet o último desses meios, o seu impacto na língua

tem sido o mais revolucionário de todos [Crystal 2005].

A Internet antes continha na sua maioria hipertextos e algumas imagens. Hoje ela possui muitas linguagens, com multimídias, trazendo textos, imagens, sons, vídeos, *emojis* e animações. Tornando-se um “...veículo eletrônico, global e interativo...” [Crystal 2005] (p. 80), trazendo consequências para a linguagem, tanto a encontrada na rede, como a falada entre as pessoas na sociedade. A Internet explora linguagens visuais como forma de atrair e alcançar um maior número de pessoas. Linguagens estas que estão presentes no dia a dia dos estudantes, por meio da TV, dos vídeos, das animações, dos jogos, dos aplicativos, dos ícones e dos dispositivos móveis. A linguagem que cerca os estudantes fora da escola, nas mídias e redes sociais, é uma linguagem mais visual. Uma linguagem mais fácil de se compreender, de identificar os componentes e as relações entre estes. Mas quando o aluno chega na escola tudo é diferente do mundo que o cerca lá fora, sendo menos atrativo e voltado à educação do século passado, uma educação textual e oral, uma educação desconectada da sua realidade [Demo 2012].

E como fica a aprendizagem neste contexto de imagens, de multimídias, onde os alunos recebem estímulos visuais e sonoros? Devemos ter o mesmo sistema de ensino de anos atrás? Como justificativa para a realização deste trabalho, destaca-se o potencial didático da imagem como facilitadora no processo de ensino-aprendizagem, pois “... são mais facilmente lembradas do que suas correspondentes representações verbais” [Martins et al. 2005].

Na computação existem várias linguagens com diversos níveis de abstração para solucionar problemas tais como: na engenharia de software, os diagramas de fluxo de dados (DFDs) e os diagramas da Linguagem de Modelagem Unificada (UML - *Unified Modeling Language*); em banco de dados, os diagramas de entidade-relacionamento; em algoritmos, os diagramas de blocos e os fluxogramas; em sistemas distribuídos, as redes de Petri; na teoria da computação, os autômatos e as gramáticas, entre outros. As linguagens de especificação permitem descrever um sistema em alto nível de abstração, e através de um processo de refinamento, com a adição de detalhes de implementação, pode-se chegar a um algoritmo descrito em uma linguagem de programação. Existem muitos tipos de linguagens de programação, variando de textuais a visuais. Quanto mais visual é a linguagem maior é o nível de abstração, facilitando a construção das soluções, em geral, em um ambiente mais lúdico e atraente. No entanto, todas estas abordagens focam na área da Computação.

Assim, o objetivo deste trabalho é propor uma linguagem visual de alto nível de abstração, fundamentada nas técnicas da computação para resolver problemas, e que possa ser aplicada por um não especialista em qualquer área do conhecimento.

3.1. Sintaxe

O núcleo da linguagem proposta para a resolução de problemas fundamenta-se nos conceitos de decomposição e abstração. Estes são os fundamentos geralmente utilizados para especificar a solução de problemas computacionais, e que podem ser (re)aplicados em qualquer área de conhecimento. Propõe-se a descrição das soluções em etapas. A especificação de uma solução parte de um nível alto de abstração (onde detalhes não são considerados) e vai sendo incrementalmente decomposta e detalhada até que o dispositivo, máquina ou pessoa que irá executá-la seja capaz de interpretá-la. Primeiro, identificam-

se quais são as informações conhecidas e necessárias à resolução do problema e qual o resultado esperado. Nesta etapa, o conjunto de ações ou sequência de passos que levam à solução do problema são desconsiderados, apenas identifica-se a ação principal que levará à solução do problema. A partir disso, a ação principal vai sendo decomposta em um conjunto de ações que, combinadas, levam a solução do problema. Essas ações, por sua vez, devem ser especificadas e as relações entre elas estabelecidas.

Uma solução na linguagem visual proposta é especificada por um conjunto de componentes representados por retângulos. Em particular, uma solução é dada por um componente principal e um conjunto de componentes decompostos. O componente principal é um componente elementar que descreve a ação que soluciona o problema como um todo. Um componente elementar, ilustrado na Figura 1, descreve uma ação a ser executada sem detalhá-la, apenas especificando as informações de entrada necessárias para sua execução, bem como os resultados obtidos. Cada informação de entrada (setas rotuladas por $\mathbf{in}_i$) é associada a uma porta de entrada do componente (quadrados pretos na borda superior do componente), assim como cada resultado (setas rotuladas por $\mathbf{out}_i$) é associado a uma porta de saída (quadrados pretos na borda inferior do componente).

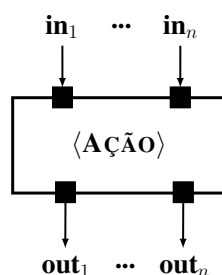


Figura 1. Componente Elementar

Um componente decomposto, Figura 2(a), apresenta um conjunto de subcomponentes que descrevem o seu comportamento com mais detalhes. Cada subcomponente é definido por um componente elementar e as conexões (dependências) entre eles são descritas por setas que conectam suas portas. Cada porta pode ter uma conexão que chega e uma ou mais (quando a informação deve ser distribuída para mais de um componente) que saem. A Figura 2(b) ilustra os três tipos possíveis de conexões: entre uma porta de entrada do componente e uma porta de entrada de um subcomponente (seta azul); entre uma porta de saída de um subcomponente e uma porta de saída do componente (seta laranja); e entre uma porta de saída de um subcomponente e uma porta de entrada de outro subcomponente (seta verde). Os dois primeiros tipos definem como as entradas e saídas do componente decomposto estão relacionadas com os subcomponentes e o último tipo define as dependências entre os subcomponentes.

Além disso, as informações de entrada ou saída do componente decomposto também podem ser mais detalhadas, conforme as Figuras 3(a) e 3(b), respectivamente. O detalhamento de uma entrada $\mathbf{in}$ é definido por um “*split*” dando origem a diversas entradas ($\mathbf{in}_1 \dots \mathbf{in}_k$), as quais especificam a entrada $\mathbf{in}$. Já o detalhamento de uma saída $\mathbf{out}$ é definido por um “*join*” juntando várias saídas ($\mathbf{out}_1 \dots \mathbf{out}_k$), as quais especificam a saída $\mathbf{out}$. As operações de “*split*” e “*join*” são representadas por barras horizontais que interceptam setas de entrada e saída, respectivamente.

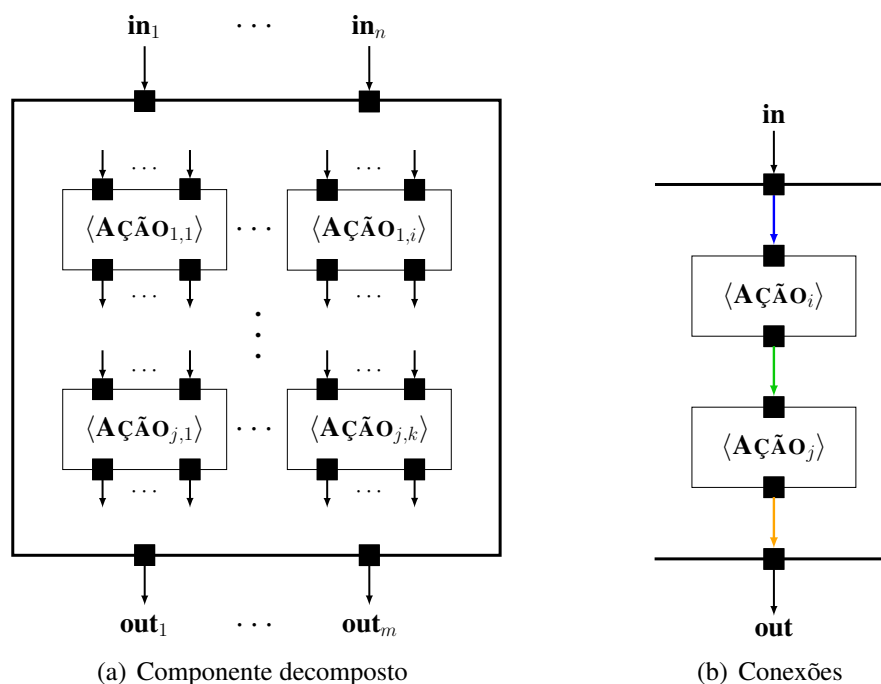


Figura 2. Componente decomposto e os diferentes tipos de conexão.



Figura 3. Detalhamento das entradas e das saídas.

4. Metodologia de solução de problemas utilizando a linguagem visual proposta

Nesta Seção, define-se a metodologia de uso da linguagem. Partindo de um problema, a descrição da solução deve seguir as seguintes etapas:

- E1 Especifica-se a solução do problema por um componente elementar, denominado componente principal. Nesta etapa são identificadas todas informações de entrada (mesmo em um nível mais alto de abstração) assim como os resultados a serem alcançados.
- E2 Refina-se o componente principal por meio de um componente decomposto. Nesta etapa são identificados subcomponentes que detalham a ação do componente principal e suas conexões.
- E3 Refina-se cada subcomponente identificado na etapa anterior. Esta etapa é executada sucessivas vezes até que o nível de detalhamento desejado seja alcançado (podendo não ser executada caso o nível de detalhamento seja alcançado na E2).

Caso se identifique a falta de alguma informação em algum nível de refinamento, deve-se voltar na especificação mais abstrata para adicioná-la. Neste caso, todos os componentes de níveis anteriores devem ser revistos.

Para exemplificar o uso da linguagem, especifica-se a solução do problema de confeccionar um bolo com cobertura. Na primeira etapa (E1), no nível mais alto de abstração, descreve-se o componente principal, conforme Figura 4, identificando a ação a ser executada (FAZER BOLO), as informações de entrada (os **Ingredientes** e os **Utensílios**) e os resultados/saídas (o **Bolo com Cobertura** e os **Utensílios**).

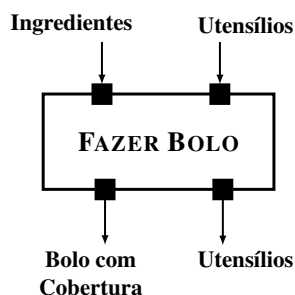


Figura 4. Componente elementar - Fazer Bolo

Na segunda etapa (E2), em um nível mais baixo de abstração, detalha-se a ação FAZER BOLO em um conjunto de ações que levam ao resultado esperado. Neste nível identificam-se quais ações menores compõem a ação principal. Em particular, a ação FAZER BOLO foi decomposta em três ações: FAZER MASSA, FAZER COBERTURA e MONTAR BOLO. As relações (dependências) entre elas estão especificadas pelas conexões entre os subcomponentes. Neste caso, a ação MONTAR BOLO depende dos resultados das ações FAZER MASSA e FAZER COBERTURA. Na descrição da solução estas dependências podem ser observadas pelas conexões representadas na Figura 5. Também é possível observar que as ações FAZER MASSA e FAZER COBERTURA são independentes entre si, já que as entradas de ambas não são resultados de outros subcomponentes. Além disso, as entradas do componente decomposto foram detalhadas. O *split* da entrada **Ingredientes** a detalha diferenciando os ingredientes da massa dos ingredientes da cobertura. Já o *split* da entrada **Utensílios** a detalha diferenciando os utensílios usados para fazer a massa, daqueles para fazer a cobertura e dos usados para montar o bolo. A saída **Utensílios** também foi detalhada, onde o *join* reúne todos os tipos de utensílios utilizados para compor a saída do componente decomposto. Caso a descrição já esteja detalhada o suficiente para alguém fazer o bolo, a especificação seria encerrada na etapa E2, caso contrário, passa-se para a etapa E3 onde as ações FAZER MASSA, FAZER COBERTURA e MONTAR BOLO também devem ser especificadas por componentes decompostos.

5. Um estudo de caso: metodologia de aplicação e resultados

Um estudo de caso com duração de 90 minutos foi realizado com 24 alunos do primeiro semestre de um curso superior de CC. A turma foi dividida em 3 grupos de 8 alunos, cada um organizado em 4 duplas. Foi apresentado um problema (descrito no quadro a seguir) para que os alunos o solucionassem: um grupo (4 duplas) usando linguagem natural e um grupo (4 duplas) usando a linguagem visual e sua metodologia de aplicação.

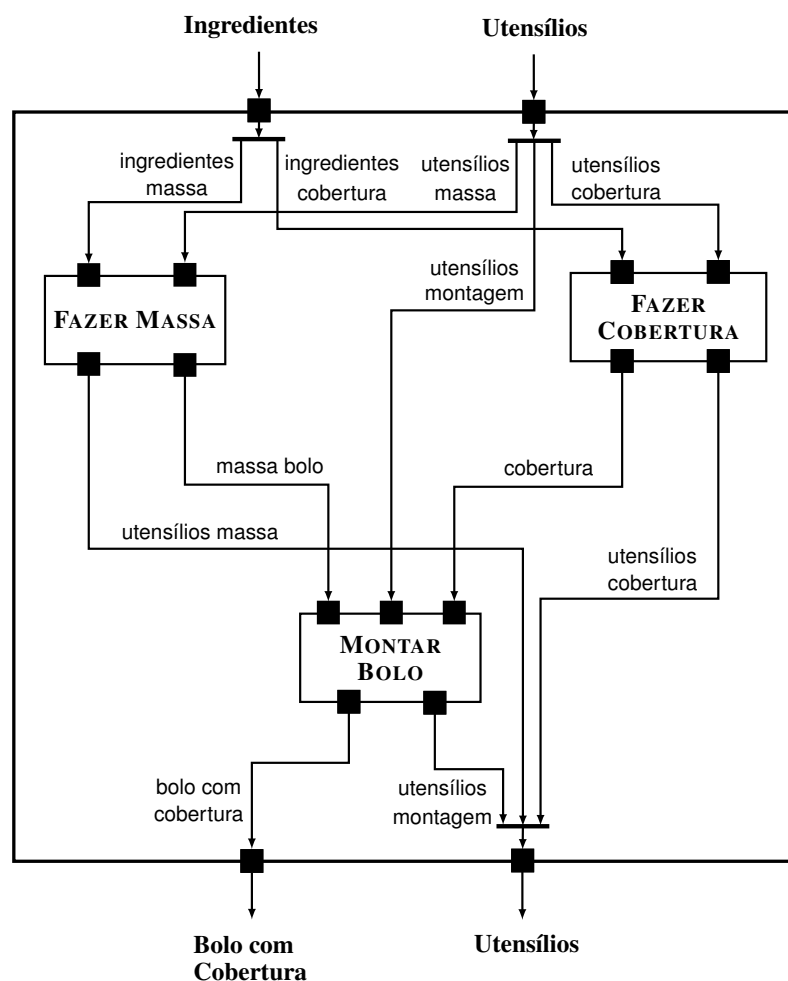


Figura 5. Componente decomposto - Fazer Bolo

PROBLEMA:

Uma empresa deseja calcular mensalmente o valor que deve ser destinado ao pagamento de Pro Labore de seus sócios e o valor que deve ser investido na própria empresa. O valor destinado ao Pro Labore deve ser de 40% do lucro e o restante deve ser destinado a investimentos na empresa. Os valores recebidos podem ser obtidos através da listagem dos produtos vendidos durante o mês. Já os valores gastos podem ser obtidos através das notas fiscais dos insumos comprados e da listagem dos salários pagos aos funcionários no mês. Alguns custos associados devem ser considerados: ICMS é o imposto sobre circulação de mercadorias que deve ser pago sobre os produtos vendidos; e o FGTS é um valor recolhido pela empresa sobre o salário dos funcionários. O ICMS corresponde a 18% do valor bruto de vendas e o valor do FGTS é de 8% do valor dos salários pagos aos funcionários.

A apresentação da linguagem visual e de sua metodologia de aplicação foi feita em aproximadamente 25 minutos somente para o grupo que trabalhou nesta linguagem. Para o grupo que utilizou linguagem natural para a descrição da solução do problema, houve tempo para realizar simulações das soluções propostas. Estas simulações foram realizadas pelos colegas do terceiro grupo, sendo que estes não participaram de atividades de resolução do problema em nenhuma das linguagens. Já para o grupo que trabalhou

com a linguagem visual não foi possível realizar simulações devido à necessidade de tempo para apresentação da linguagem. Para este grupo, as simulações foram realizadas posteriormente pelos próprios autores.

Diversos pontos positivos foram observados nas soluções propostas na linguagem visual. Cabe destacar: todos os grupos usaram corretamente a linguagem e chegaram em uma solução correta para o problema; apenas um dos grupos não utilizou diferentes níveis de abstração, já identificando todos os subcomponentes na etapa E2; todos os grupos identificaram corretamente as entradas e saídas, sendo que dois deles também utilizaram abstração de dados; embora os grupos tenham dado soluções diferentes para o problema, todos aplicaram a decomposição de forma adequada.

Por sua vez, para as soluções descritas usando a linguagem natural, observou-se que: nenhum grupo forneceu uma solução totalmente correta, isto é, seguindo literalmente as soluções descritas não foi possível chegar ao resultado esperado, porém dois dos grupos descreveram uma sequência de passos que leva a uma solução parcialmente correta; algumas simulações apresentaram erros, devido à ambiguidade e detalhamento insuficiente na descrição das soluções (por exemplo, operações ficaram subentendidas) ou uso de terminologias equivocadas (por exemplo, “calcular o lucro bruto da empresa listando todos os produtos vendidos”); os grupos tiveram dificuldade em estabelecer as relações entre os resultados de um passo e as entradas de outros, sendo que dois deles não conseguiram estabelecer relação alguma.

Comparando as soluções nas duas linguagens, tem-se algumas observações: os alunos que usaram a linguagem visual, por terem decomposto o problema, acabaram chegando a um nível maior de detalhamento e exatidão; e o uso da linguagem visual, embora ainda não definida formalmente, exigiu que todas as conexões (dependências) entre as ações fossem especificadas, o que não ocorreu com as descrições em linguagem natural.

6. Considerações Finais

O Pensamento Computacional abrange um conjunto de habilidades que são importantes para a resolução de problemas no século XXI. Neste momento em que os estudantes estão expostos a muitos estímulos visuais, propõe-se trabalhar o PC em sala de aula de uma forma que seja mais próxima e atrativa, por meio de uma linguagem visual. Propõe-se aproximar o contexto do aluno de fora da sala de aula para dentro desta, com a ideia de motivar e facilitar o ensino e a aprendizagem.

O núcleo da linguagem visual sugerida neste trabalho fundamenta-se nos conceitos de abstração e decomposição. Acredita-se que estes dois conceitos são geralmente utilizados para a resolução de problemas em qualquer área do conhecimento. Também apresentou-se uma metodologia de uso da linguagem bem como um estudo de caso.

Alguns alunos comentaram que gostaram da metodologia de solução através da linguagem visual, mas acharam trabalhoso desenhar os diagramas. Assim, como trabalho futuro pretende-se, além de especificar formalmente a linguagem, desenvolver uma ferramenta de edição e tradução. Ademais, pretende-se estender a mesma com outros conceitos da computação, tais como, generalização, recursão, dentre outros.

Referências

Andrade, D. et al. (2013). Proposta de atividades para o desenvolvimento do pensamento

- computacional no ensino fundamental. In *Anais do WIE 2013*, pages 169–178.
- Bordini, A. et al. (2016a). Computação na educação básica no Brasil: o estado da arte. *Revista de Informática Teórica e Aplicada*, 23(2):210–238.
- Bordini, A. et al. (2016b). Desdobramentos do pensamento computacional no Brasil. In *Anais do SBIE 2016*, pages 200–209.
- Bordini, A. et al. (2017). Pensamento computacional nos ensinos fundamental e médio: uma revisão sistemática. In *Anais do SBIE 2017*, pages 123–132.
- Brasil (2018). Base nacional comum curricular. <http://basenacionalcomum.mec.gov.br/>.
- CAS, C. A. S. (2015). Computational thinking, a guide for teachers. <http://community.computingatschool.org.uk/resources/2324/single>.
- Crystal, D. (2005). *A revolução da linguagem*. Zahar.
- CSTA (2017). Csta k-12 computer science standards. <https://www.csteachers.org/page/standards>.
- Demo, P. (2012). *O porvir: desafio das linguagens do século XXI*. InterSaberes.
- Experience, E. (2018). 13 essential 21st century skills for todays students. <https://goo.gl/xnSrXE>.
- Marques, M. et al. (2017). Uma proposta para o desenvolvimento do pensamento computacional integrado ao ensino de matemática. In *Anais do SBIE 2017*, pages 314–323.
- Martins, I. et al. (2005). Aprendendo com imagens. *Ciência e Cultura*, 57(4):38–40.
- Martins, L. et al. (2016). Ensinando lógica de programação aplicada a robótica para alunos do ensino fundamental. In *Anais do SBIE 2016*, pages 31–41.
- Mohaghegh, D. M. and McCauley, M. (2016). Computational thinking: the skill set of the 21st century. pages 1524–1530.
- Mota, F. et al. (2014). Desenvolvendo o raciocínio lógico no ensino médio: uma proposta utilizando a ferramenta scratch. In *Anais do SBIE 2014*, pages 377–381.
- Oliveira, E. et al. (2016). Pensamento computacional e robótica: Um estudo sobre habilidades desenvolvidas em oficinas de robótica educacional. In *Anais do SBIE 2016*, pages 530–539.
- P21.org (2018). Framework for 21st century learning. <http://goo.gl/VS7HGD>.
- Papert, S. (1994). *A máquina das crianças: repensando a escola na era da informática*. Porto Alegre: Artes Médicas.
- Tabesh, Y. (2017). Computational thinking: A 21st century skill. 11:65–70.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3):33–35.
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical transactions of the royal society of London A: mathematical, physical and engineering sciences*, 366(1881):3717–3725.
- Yadav, A. et al. (2016). Computational thinking for all: pedagogical approaches to embedding 21st century problem solving in k-12 classrooms. *TechTrends*, 60(6):565–568.