

## Um Instrumento para Diagnóstico do Pensamento Computacional

André Luís Alice Raabe<sup>1,2,3</sup>, André Luiz. Maciel Santana<sup>1</sup>, Filipe Augusto Gonçalves<sup>3</sup>, Natália Ellery Ribeiro Couto<sup>3</sup>

<sup>1</sup>Laboratório de Inovação Tecnológica na Educação - CTTMar - UNIVALI

<sup>2</sup>Programa de Pós-Graduação em Educação

<sup>3</sup>Mestrado em Computação Aplicada

{raabe, andrelms}@univali.br, {nataliaellery, flpgst}@gmail.com

*Abstract. Authors of the scientific community have been discussing the possibility of the computational thinking's development improve aspects of the individual's cognitive ability. This research's goal is to develop an instrument to diagnose computational thinking and identify relationships between the individual's formation and computational thinking's features. I also seeks to evaluate the instrument and validate the computational thinking diagnosis of an individual. It also seeks to validate how the results obtained can reflect if the computational thinking can be diagnosed in an individual. Due to the statistical analyzes carried out were found clues where students who had more contact with computation's elements had better performance diagnosed by the instrument..*

*Resumo. Autores da comunidade científica têm discutido amplamente sobre a possibilidade do desenvolvimento do pensamento computacional melhorar aspectos da capacidade cognitiva do indivíduo. Esta pesquisa tem como objetivo desenvolver um instrumento de diagnóstico do pensamento computacional e aplicar para identificar relações entre a formação dos indivíduos e o pensamento computacional. Busca também validar como os resultados obtidos podem refletir se o pensamento computacional pode ser diagnosticado em um indivíduo. Busca também avaliar o instrumento através dos resultados obtidos em relação a validade do diagnóstico gerado sobre pensamento computacional de um indivíduo. Através da aplicação foram encontrados indícios onde indivíduos que tiveram contato com elementos da computação tiveram melhor desempenho diagnosticado pelo instrumento desenvolvido.*

### 1. Introdução

O Pensamento Computacional é um termo que sintetiza o conjunto de habilidades cognitivas que os profissionais da área de Computação geralmente desenvolvem em seu processo de formação. Wing apresentou este termo através da publicação de um artigo em 2006, defendendo que o Pensamento Computacional deve ser introduzido como uma habilidade na formação da criança. Após esta declaração de Wing, estão surgindo diversas iniciativas promovendo a inserção do Pensamento Computacional como componente do currículo na educação escolar (Wing, 2006).

A CSTA define o pensamento computacional como um processo de resolução de problemas que inclui, não exclusivamente, as seguintes características: (i) Formulação

de problemas de forma que computadores e outras ferramentas possam ajudar a resolvê-los; (ii) Organização lógica e análise de dados; (iii) Representação de dados através de abstrações como modelos e simulações; (iv) Automatização de soluções através do pensamento algorítmico; (v) Identificação, análise e implementação de soluções visando a combinação mais eficiente e eficaz de etapas e recursos (vi) Generalização e transferência de soluções para uma ampla gama de problemas (CSTA, 2011).

Um fator motivador para a exploração do pensamento computacional no âmbito da educação básica no Brasil é o baixo desempenho publicado pelo Programme for International Student Assessment (PISA) o qual avalia a performance escolar em matemática, ciências, letramento, resolução de problemas e cognição de alunos de 15 anos de idade. No último relatório no ano de 2014 o Brasil consta na trigésima oitava posição dentre os quarenta e quatro países analisados no teste de resolução de problemas. “Ao longo da última década, o PISA, tornou-se critério mundial para avaliar a qualidade, equidade e eficiência dos sistemas escolares” (OECD, 2014, p. 03).

Existem algumas abordagens que buscam se aproximar de uma mensuração do Pensamento Computacional. A equipe de Brennan e Resnick utilizou o Scratch como ferramenta para estudantes desenvolverem projetos e avaliaram os resultados através de contagem de comandos de programação, entrevistas e atividades. A conclusão mostra que as avaliações consomem bastante tempo e são realizadas de forma bastante qualitativas (RESNICK, BRENNAN, 2012). Já a SRI criou um framework que descreve quais habilidades devem ser avaliadas a fim de medir o PC com base no currículo da Exploring Science Computer e do Advanced Placement Science Computer Principles (SRI, 2015). O framework apresenta padrões de estratégias que visam avaliar o PC nas práticas: analisar os efeitos do desenvolvimento na computação; projetar e implementar soluções e artefatos criativos; projetar e aplicar abstrações e modelos; analisar o próprio trabalho computacional e o dos outros; comunicação através de processos e resultados; colaborar com outros em atividades computacionais.

É possível verificar que as abordagens encontradas na literatura são qualitativas, por este motivo não é apresentado um meio de quantificar as habilidades desenvolvidas pelo pensamento computacional nos estudantes, o qual seria de grande ajuda na avaliação do conteúdo para a implantação na formação básica dos indivíduos.

Tendo em conta as abordagens já existentes, este trabalho busca contribuir com o desenvolvimento de um método para calcular uma pontuação a fim de quantificar o desenvolvimento da habilidade no indivíduo, além de identificar quais conceitos utilizados em atividades podem ser utilizados para mensurar esta pontuação. Para isso foi desenvolvido um instrumento com diversas atividades em conformidade com definições do pensamento computacional segundo a CSTA. Estas atividades foram aplicadas em amostras considerando diversas faixas etárias para ampliar a validade de constructo e a fidedignidade das mensurações conforme ferramental metodológico tradicional na área de psicometria.

## **2. O instrumento**

O instrumento aborda atividades de resolução de problemas, planejadas para que não seja necessário conhecimento prévio de conceitos computacionais, focando nos elementos constituintes do pensamento computacional, os quais são mensurados através

dos resultados e desempenho na resolução das atividades. As atividades foram baseadas em testes de QI e Questões do Programa de Enriquecimento Instrumental de Feuerstein (2006), e adaptadas para o formato de jogo puzzle. Este formato foi escolhido pois foi identificado que as iniciativas que abordam atividades gameficadas envolvendo conceitos do pensamento computacional são em sua maioria neste formato, como são os casos verificados em: A hora do código, LightBot, The Foos, Blockly Games e Tynker. Hong et al. (2012) define que jogos de gênero puzzle incentivam os jogadores a terem atenção imediata nos seus erros, facilitando a autoavaliação mais rapidamente do que em outras abordagens.

Cada uma das questões contidas no instrumento é apoiada em uma ou mais expressões do vocabulário definido em CSTA (2011), o qual sintetiza o pensamento computacional em termos. Foram elaborados nove tipos de questões, onde cada qual corresponde a uma fase do jogo que é composta por mais de um nível. Para melhor entendimento, a partir deste momento cada questão do instrumento será abordada como uma fase que compõe o jogo. Na tabela 1 é apresentada a relação entre as questões e os termos da CSTA.

**Tabela 1. Elementos do pensamento computacional presentes em cada fase**

Termos da CSTA	F a s e 1	F a s e 2	F a s e 3	F a s e 4	F a s e 5	F a s e 6	F a s e 7	F a s e 8	F a s e 9
Formulação de problemas de forma que computadores e outras ferramentas possam ajudar a resolvê-los								X	
Organização lógica e análise de dados						X	X	X	
Representação de dados através de abstrações como modelos e simulações	X	X	X	X					X
Automatização de soluções através do pensamento algorítmico	X								X
Identificação, análise e implementação de soluções visando a combinação mais eficiente e eficaz de etapas e recursos	X	X		X			X	X	X
Generalização e transferência de soluções para uma ampla gama de problemas	X	X	X	X	X		X	X	X
Confiança em lidar com complexidade									X
Persistência em trabalhar com problemas difíceis	X	X	X	X	X	X	X	X	X
Tolerância para ambiguidades	X					X	X		X
A habilidade de lidar com problemas abertos	X				X		X	X	X
A habilidade de comunicar e trabalhar com outros para atingir um objetivo comum ou uma solução									
Coleta de dados						X			
Análise de dados		X	X	X	X	X		X	
Representação de dados						X		X	
Decomposição	X					X		X	X
Abstração	X							X	X
Algoritmos e procedimentos	X								X
Automação									
Simulação	X								X
Paralelismo								X	
<b>Total</b>	<b>11</b>	<b>5</b>	<b>4</b>	<b>5</b>	<b>4</b>	<b>7</b>	<b>6</b>	<b>11</b>	<b>12</b>

Em cada uma das fases são capturadas as interações para posteriormente construir o diagnóstico do pensamento computacional do indivíduo exposto ao instrumento, então é gerado um escore com base na captura destes dados, o qual é utilizado para identificar o quanto está desenvolvida a habilidade relacionada ao elemento abordado.

Na primeira fase do jogo o indivíduo deve informar as direções para fazer com que o personagem se movimente até o ponto final. Cada passo do personagem é

representado por um comando que deve ser arrastado para o bloco PROGRAMA. Após posicionar os comandos desejados é necessário apertar o ícone play, então o personagem começa a se movimentar conforme os passos da execução do programa construído. Esta fase é composta por 7 níveis e a cada nível a posição do personagem, dos obstáculos e do ponto de chegada são alterados, tornando o desafio cada vez mais difícil. Os níveis são intercalados entre as demais fases para não se tornar repetitivo, já que cada nível pode levar mais tempo para ser resolvido do que os outros desafios.

A segunda fase, que é composta por oito níveis, é baseada na atividade de conectar os pontos para formar figuras. São apresentadas as figuras que deverão ser encontradas e em seguida os pontos dispersos na tela, o indivíduo deve conectá-los de forma a criar o formato da figura apresentada. A partir desta fase, todas as demais apresentam a possibilidade do indivíduo requisitar três dicas. Para este desafio são disponibilizadas dicas sinalizando uma conexão entre pontos que leva ao resultado. Também é apresentado o comando limpar que é utilizado para desfazer todas as conexões entre os pontos que não formaram a figura a ser encontrada.

A terceira fase é composta por duas figuras: a menor é uma peça que pode ser encontrada n vezes dentro da figura maior, esta fase é dividida em cinco níveis. Para resolver este desafio o indivíduo deve arrastar e rotacionar a peça fornecida para dentro da figura de modo que essa encaixe no espaço correspondente, o procedimento deve ser repetido até todas as peças se esgotarem.

A quarta fase é baseada no quebra-cabeça Tangram, o qual possui partes de um triângulo retângulo dispersas e precisam ser agrupadas para formar a figura. Esta fase é composta por dois níveis onde o mesmo triângulo é apresentado, porém as peças que o formam são fatiadas de formas diferentes.

Na elaboração da quinta fase foram utilizados os conceitos de análise de padrões, ela é composta por dois níveis. Para resolver este desafio é necessário identificar entre as opções fornecidas, qual completa a sequência de figuras que compõem este nível.

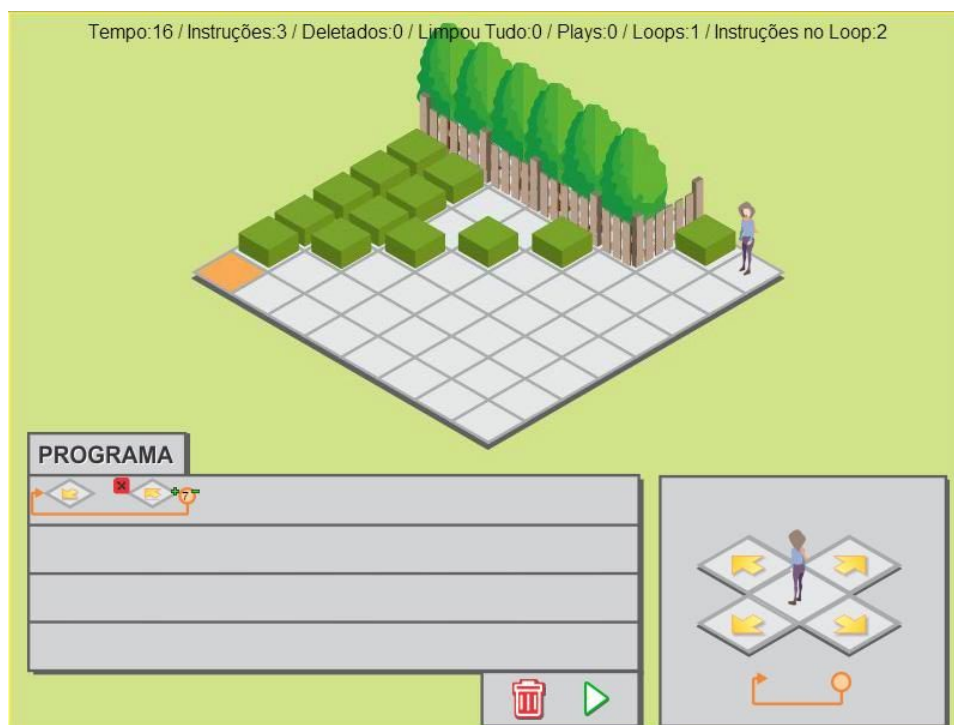
A sexta fase foi criada baseada no conceito de classificação e organização de dados, o indivíduo deve encontrar a relação entre os objetos espalhados pela tela e organizá-los em linhas e colunas de uma tabela considerando suas características referente a cor e o tipo do objeto, podendo ter diferentes respostas. Nesta fase é apresentado o botão limpar com a função de reiniciar a fase espalhando todos os itens para fora da tabela.

A sétima fase é apresentada como o desafio dos palitos e é composta por quatro níveis. Os palitos iniciam cada nível em um determinado formato, porém outro formato deve ser construído com um número limitado de movimentos. Os palitos podem ser movidos e rotacionados livremente, até a forma correta ser encontrada. Quando o indivíduo ultrapassa a quantidade máxima de movimentos o nível é reiniciado e todos os palitos retornam à posição inicial.

A oitava fase é composta por um nível que apresenta uma estrutura dividida por trinta e dois quadrados que devem ser pintados de forma a produzir duas figuras simétricas. O cursor do mouse é transformado em um pincel e uma paleta de cores é apresentada, a qual permite que a cor do pincel possa ser alterada. Quando todos os

quadrados são pintados e a forma correta não é encontrada, uma mensagem informando que a forma não está correta é apresentada.

A última fase do jogo é muito similar à primeira, porém é acrescido o comando loop. Esta fase é composta por dois níveis, apesar do comando loop estar disponível para utilização ambos os níveis podem ser solucionados sem a utilização deste comando porém o escore é influenciado pela não utilização do loop. Para fazer a utilização do comando loop, ele deve ser arrastado para dentro do bloco PROGRAMA e as instruções que o irão compor devem ser arrastadas para dentro do loop, a quantidade de vezes que o loop vai ocorrer pode ser selecionada clicando nas opções + e – que aparecem ao lado da instrução. Na figura 1 pode ser visualizada a última fase do jogo.



**Figura 1. Fase 9**

### 3. Validação e resultados

Ao todo foram efetuadas 5 diferentes aplicações do instrumento, com o objetivo de avaliar a validade aparente, a qual avalia a proximidade entre a operacionalização e o conceito da pesquisa (Trochim, 2008); a validade de conteúdo, referindo-se a avaliação de proximidade entre as informações que a amostra foi exposta e as informações presentes no teste (Viana 1982); a validade concorrente, a qual avalia a correlação entre a operacionalização e outro critério relacionado; e a validade preditiva, que consiste em prever um comportamento futuro. Em todas as aplicações foi avaliada também a fidedignidade do instrumento. As aplicações envolveram especialistas e estudantes universitários e do ensino médio.

O foco do instrumento é compor um diagnóstico do pensamento computacional do indivíduo, então não é construído um escore final, a finalidade não é gerar uma nota de pensamento computacional, mas pontuar em quais elementos o indivíduo é mais ou

menos desenvolvido. Para atender este requisito, foi definida a cada questão uma equação matemática que resulta em um escore.

A análise dos escores coletados durante a resolução do instrumento possibilita criar diagnósticos individuais que relatam quais elementos do pensamento computacional estão mais fortemente presentes no indivíduo. A elaboração das equações teve como base as interações capturadas durante a resolução do instrumento, por isso foram definidas diferentes equações para cada fase, porém existem definições gerais aplicadas em todas as equações:

- O escore máximo das fases é 1 e o escore mínimo é 0;
- Se o indivíduo desistiu do nível utilizando a opção pular, o escore desse nível é 0;
- O escore final de cada fase é a somatória dos escores de seus níveis, dividido pela quantidade de níveis que formam a fase;
- O valor total da fase, nunca é menor que 0;
- O tempo máximo de resolução de cada nível é de 30 minutos, ultrapassado esse valor o nível recebe o escore 0;
- Cada parâmetro é multiplicado pelo peso que representa na fase, os quais foram ajustados por relevância através de testes com as amostras.

A Tabela 2 é apresentada como exemplo de escores calculados em cima de dados obtidos de um indivíduo ao qual o instrumento foi aplicado. Os escores são referentes à Fase 2, onde C é a quantidade de cliques utilizados, C<sub>min</sub> é a quantidade mínima de cliques necessários para completar a fase, D é a quantidade de dicas utilizadas, L a quantidade de vezes que utilizou o comando limpar e T o total do tempo utilizado.

**Tabela 2. Exemplo de escores coletados da Fase 2**

Fase 2	C	C <sub>min</sub>	D	L	T	Pulou	Escore
Nível 1	9	9	0	0	11	Não	0,99
Nível 2	9	9	0	0	7	Não	1,00
Nível 3	9	9	0	0	7	Não	1,00
Nível 4	9	9	0	0	6	Não	1,00
Nível 5	13	9	0	0	13	Não	0,91
Nível 6	32	9	0	20	86	Não	0,30
Nível 7	13	9	0	0	12	Não	0,91
Nível 8	11	9	0	0	7	Não	0,96

Para chegar no escore, é realizado um cálculo com os dados coletados, o jogador sempre irá iniciar com a pontuação 1, conforme o desempenho durante a resolução da atividade a pontuação é reduzida. Na fase 2, o resultado de (C – C<sub>min</sub>) é multiplicado por 0,02 por gerar menos impacto no resultado, já a quantidade de dicas utilizadas é o dado mais relevante neste exercício, por isso é multiplicado por 0,1, sendo o maior redutor de pontuação nesta equação, a qual é apresentada na equação 1.

$$F_2 = 1 - ((C - C_{min})0.02) - (D * 0.1) - (L * 0.01) - (T * 0.0005) \quad (1)$$

A relevância de cada dado utilizado nas equações foi ajustada conforme foram realizados testes nas amostras levando em consideração a formação dos indivíduos envolvidos nos testes. A tabela 3 demonstra os dados utilizados em cada fase para construir a equação que gera o escore.

**Tabela 3. Dados coletados em cada fase**

Dados Coletados	F a s e 1	F a s e 2	F a s e 3	F a s e 4	F a s e 5	F a s e 6	F a s e 7	F a s e 8	F a s e 9
Tempo utilizado para resolver o desafio.	X	X	X	X	X	X	X	X	X
Quantidade de instruções inseridas no bloco de instruções.	X								X
Quantidade de instruções deletadas.	X								X
Quantidade do uso de "Limpar tudo"	X	X				X	X	X	X
Quantidade de cliques na função "Play"	X								X
Quantidade de dicas utilizadas.		X	X	X	X	X	X	X	
Quantidade de giros.			X	X			X		
Quantidade de tentativas.					X	X	X		
Quantidade de cliques nos objetos das fases.		X	X	X			X	X	
Quantidade de loops utilizados									X
Quantidade de instruções dentro dos loops									X

Os dados coletados variam conforme os elementos existentes em cada fase, todas elas possuem a contagem do tempo mas existem dados muito específicos que aparecem apenas em uma ou duas fases, como por exemplo os elementos de programação ou a possibilidade de girar peças.

Os escores foram convertidos para uma escala de habilidades composta por três itens: habilidade amplamente desenvolvida (HAD), habilidade moderadamente desenvolvida (HMD) e habilidade pouco desenvolvida (HPD). A conversão dos escores foi efetuada verificando o intervalo entre o máximo valor das fases que é 1 e mínimo que é 0, assim os itens que compõem a escala estão divididos pelo intervalo de 0,33.

### 3.1. Validade aparente e de conteúdo

As 2 primeiras aplicações do instrumento foram utilizadas para a avaliação da validade aparente e de conteúdo, envolvendo especialistas do Grupo de Informática na Educação (GIE) - UNIVALI e duas turmas do projeto Lite is Cool do Laboratório de Inovação Tecnológica na Educação (LITE) – UNIVALI compreendidas por 11 estudantes do 1º e 2º ano do ensino médio.

Estas aplicações resultaram na formatação do instrumento como um jogo de computador, mascarando o fato de que os indivíduos estão sendo testados. Foi possível também: excluir três questões que apresentaram difícil interpretação da resolução, reformular as questões de programação, implementar um assistente que explica como funciona cada interação nas fases, encontrar problemas e possíveis soluções não mapeadas.

### 3.2. Validade concorrente

Outras duas aplicações foram utilizadas para a validação concorrente, com estudantes dos cursos de graduação da UNIVALI, sendo 6 de Ciência da Computação, 11 de Engenharia da Computação e 8 de Psicologia. Foi verificada a correlação entre o escore em cada elemento do pensamento computacional e os indivíduos entre as turmas.

Para encontrar o coeficiente de correlação entre as turmas foi efetuado a média aritmética. Entre as turmas de Ciência e Engenharia da Computação, foi de 0,8. Já as turmas de Engenharia da Computação e Psicologia geraram 0,9. A última análise, entre as turmas de Ciência da Computação e Psicologia geraram 0,7.

Apesar de todas as médias das correlações estarem no intervalo de valores que podem ser interpretados como alta correlação, ao verificar as correlações individualmente é possível notar que entre os estudantes dos cursos de Ciência da Computação e Psicologia, é possível perceber mais ocorrências de baixa ou moderada correlação do que nas demais comparações, tendo 6 ocorrências de baixa correlação e o mesmo número para correlação moderada.

Ao fazer a correlação entre os cursos de Engenharia da Computação e Psicologia todas as ocorrências são de alta correlação, enquanto na correlação entre Engenharia da Computação e Ciência da Computação são apontadas 11 ocorrências de baixa correlação. Com este resultado é possível concluir que aos estudantes do curso de Ciência da Computação que foram aplicados ao instrumento tiveram um melhor desempenho do que os demais, tornando admissível considerar que este instrumento tem a validade concorrente comprovada.

### **3.3. Validade preditiva**

A última aplicação foi utilizada para testar a validade preditiva, envolvendo estudantes da rede pública de ensino do município de Itajaí-SC, com 16 estudantes do 9º ano do ensino fundamental. Foi realizada a correlação entre os escores obtidos pelos alunos e suas médias finais da disciplina de matemática.

A escolha da utilização das médias da disciplina de matemática se deve ao fato do pensamento computacional ser fortemente conectado aos fundamentos da matemática, conforme Sussman (NRC, 2010) o pensamento computacional e a matemática, ambos são constituídos por uma estrutura linguística onde são necessárias descrições precisas sobre como fazer as coisas. Wing (NRC, 2010) relata que apesar do pensamento computacional estar restrito às limitações físicas do computador, é similar ao pensamento matemático em muitos aspectos.

Ao interpretar os resultados das correlações utilizando a definição de Cohen (1988) é possível identificar nos resultados obtidos das correlações efetuadas que para 6 elementos do pensamento computacional pode-se obter alta correlação tendo o coeficiente de correlação maior que 0,50, para 10 elementos o coeficiente de correlação se demonstrou moderado e para 2 elementos a correlação foi baixa.

### **3.4. Fidedignidade**

A avaliação de fidedignidade foi efetuada em todas as aplicações que obtiveram os coletados dados para avaliação de validade do instrumento, como forma de estimar a fidedignidade e consistência interna do instrumento, foi utilizado o coeficiente alpha ( $\alpha$ ).

Assim utilizando a fórmula do  $\alpha$ , com os resultados obtidos na aplicação do instrumento efetuada com a turma do curso de Ciência da Computação, foi obtido o valor da estimativa de fidedignidade de 0,739. Para este caso, pode se dizer que ela é razoável, baseado em George e Mallery (2003) que definem uma escala de interpretação



para o  $\alpha$  onde informam que resultados entre a 0,70 e 0,79 consideram-se como consistência aceitável.

No curso de Engenharia da Computação o valor foi de 0,58 considerado como consistência ruim, o curso de Psicologia obteve um valor negativo e os estudantes do 9º ano do ensino fundamental obtiveram 0,75, considerado como aceitável. Nos casos em que o coeficiente de alpha é muito baixo, para aprimorar o instrumento é possível analisar se o fator que está influenciando no  $\alpha$  são alguns alunos que estão tendo resultados muito diferentes durante a execução do teste, ou se são algumas questões que não estão recebendo a mesma resposta.

No caso da aplicação do instrumento com o curso de Psicologia, a fase 9 tem a maior variância no seu resultado, retirando esta questão da análise, o valor encontrado passa de -0,64 para 0,62, apesar de ainda ser considerado como consistência questionável é possível verificar que determinadas questões têm maior impacto na verificação da fidedignidade do instrumento. O mesmo procedimento foi efetuado com a aplicação sobre os estudantes de Ciência da Computação, retirando a fase 8 o coeficiente de alpha aumenta em 0,3 pontos. Porém para fins estatísticos, a aplicação com mais indivíduos se faz necessária para fazer uma nova estimativa de fidedignidade.

#### **4. Conclusão**

Apesar do pensamento computacional ainda não ter uma definição única e não estarem definidas quais capacidades cognitivas estão intrínsecas a ele, durante este trabalho foi possível verificar que é possível fazer avaliações quantitativas sobre o seu desenvolvimento através da exploração de questões que abordam determinados elementos do pensamento computacional.

O fato de explorar os métodos da psicometria na construção do instrumento e utilizar como embasamento a definição operacional do pensamento computacional divulgado em CSTA (2011) possibilitou categorizar as habilidades dos estudantes em habilidade amplamente desenvolvida, habilidade moderadamente desenvolvida e habilidade pouco desenvolvida para cada um dos elementos do pensamento computacional presentes no instrumento.

As correlações mostram diferenças nos resultados, conforme esperado, os estudantes de Ciência da Computação tiveram melhor desempenho e a correlação obtida entre os estudantes de ensino fundamental e suas médias na disciplina de matemática mostrou que seis elementos do pensamento computacional possuem alta correlação. Mas devido ao pequeno número de participantes da pesquisa o resultado não é estatisticamente significativo.

O principal experimento a ser realizado no futuro com base neste trabalho é a aplicação do instrumento em uma quantidade maior de indivíduos em que os testes estatísticos possam garantir um resultado mais significativo. A elaboração de novas questões pode ainda criar a possibilidade de utilizar este instrumento como pré e pós-testes para avaliação de experimentos que exploram o desenvolvimento pensamento computacional e a resolução de problemas.

## Agradecimentos

Agradecimento especial a CAPES por fornecer bolsas aos pesquisadores proponentes por meio do programa PROSUP e ao Colégio de Aplicação da UNIVALI.

## References

- Cohen, Jacob (1988), Statistical power analysis for the behavioral sciences. 2.ed.
- Computer Science Teachers Association – CSTA (2011) “Computational thinking: teacher resources.” 2. ed.
- Feuerstein, R. (2006) “The Feuerstein Instrumental Enrichment Program” In: ICELP Publications, Jerusalem.
- George, D. e Mallery, P. (2003), SPSS for Windows step by step: A simple guide and reference, 4 ed.
- Hong, J.C. et al. (2012), “Effects of cognitive style on digital jigsaw puzzle performance: A GridWare analysis” In: Computers in Human Behavior, v. 28, n. 3, p. 920–928.
- National Research Council - NRC. (2010) “Report of a Workshop on the Scope and Nature of Computational Thinking”
- OECD. PISA (2012) “Results :Creative Problem Solving: Students’ Skills In Tackling Real-Life Problems”, <http://www.oecd.org/pisa/keyfindings/pisa-2012-results-volume-v.htm>, Dezembro
- Papert, Seymour. (1985) “LOGO: Computadores e educação Tradutor: José Armando Valente; Beatriz Bitelman; Afira Vianna Ripper. 3. ed. São Paulo: Editora Brasiliense.
- Resnick, Mitchel; Brennan, Karen. (2012) “New frameworks for studying and assessing the development of computational thinking” In: Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada.
- Sociedade da Informação no Brasil. (2000)“Livro verde” Tadao Takahashi (Org). Brasília: Ministério da Ciência e Tecnologia.
- SRI International (2015) “Assessment Design Patterns for Computational Thinking Practices in Secondary Computer Science: A First Look” <http://pact.sri.com/downloads/Assessment-Design-Patterns-for-Computational%20Thinking-Practices-Secondary-Computer-Science.pdf> , Dezembro
- Trochim, W. e Donnelly, J. (2008) “The Research Methods Knowledge Base” 3. ed. Cengage Learning: Atomic Dog.
- Vianna, Heraldo Marelim (1982) “Testes em educação” 4. ed. São Paulo: Ibrasa, 1982.
- Wing, Jeannette M. (2006) “Computational Thinking” Communications of the ACM, v.49, n.3, p.33-35, mar. 2006.