

Metodologia Didático Simbólica como Alternativa para o Ensino de Programação de Computadores a Alunos Surdos

Gil Eduardo de Andrade¹, Diego Jonathan Hoss¹,
Ana Paula Marques Barbosa², Lana Mara Gomes³

¹Docente – Eixo de Informação e Comunicação
Instituto Federal do Paraná (IFPR) – Paranaguá, PR – Brazil

²Discente – Técnico em Informática
Instituto Federal do Paraná (IFPR) – Paranaguá, PR – Brazil

³Tradutora e Intérprete de Libras / Mestranda – PPG em Ciência Tecnologia e Sociedade
Instituto Federal do Paraná (IFPR) – Paranaguá, PR – Brazil

{gil.andrade, diego.hoss, lana.gomes}@ifpr.edu.br, anapaullamb@gmail.com

Abstract. *Computers programming has been adopted as tool to conceive scientifically and technologically literate people. However, professors face challenges during the teaching process, especially when students have special needs. Recent studies show that visual tools, such as teaching software, facilitate the understanding of the contents, even if tool is not linked with the teaching material. Thus, we propose the creation of a programming teaching methodology for deaf students, called Symbolic Teaching Methodology. The Methodology integrates the Symbolic Didactic Material and Educational Software, both created for this purpose.*

Resumo. *O ensino de programação tem sido adotado como forma de contribuição para a formação de sujeitos científica e tecnologicamente alfabetizados. Contudo, seu processo de aprendizado exige o enfrentamento de desafios por parte dos docentes, principalmente quando os estudantes apresentam algum tipo de necessidade especial. Estudos recentes mostram que o uso de ferramentas visuais, como softwares de ensino, facilita a compreensão dos conteúdos, ainda que utilizadas sem vínculo direto com o material didático. Neste contexto, este trabalho propõe a criação de uma metodologia de ensino de programação voltada aos alunos surdos, chamada Metodologia Didático Simbólica (MDS). A metodologia integra o material didático simbólico e um software educacional.*

1. Introdução

Atualmente, o ensino da lógica de programação tem sido adotado como ferramenta para formação de sujeitos capazes de compreender e utilizar conceitos científicos básicos. Essa abordagem é possível porque a programação de computadores estimula o desenvolvimento do raciocínio lógico e a capacidade de abstração, além de estimular o desenvolvimento de habilidades como resolução de problemas e noções de causa e efeito [GOMES 2015].

Entretanto, efetivar a aprendizagem de programação exige, por parte dos docentes, constante reflexão e enfrentamento de desafios, que tornam-se ainda mais complexos

quando os alunos em questão apresentam algum tipo de necessidade especial. Esse cenário mostra-se crescente nos dias atuais, visto que alunos surdos têm ingressado em instituições de ensino públicas com maior frequência a cada ano, tendo como suporte a Lei de Diretrizes e Bases da Educação [BRASIL 1997].

Sendo assim, é notória a preocupação em torno da inclusão do aluno surdo no sistema regular de ensino. Porém, apenas a contratação de tradutores e intérpretes de libras não é capaz de tornar pleno o seu processo de inclusão, sendo necessária, além da quebra da barreira de comunicação, a quebra das barreiras atitudinais através da sensibilização da comunidade acadêmica [CAVALCANTI and MENEZES 2016].

O trabalho proposto apresenta uma nova metodologia de ensino, denominada Metodologia Didático Simbólica ou MDS. A MDS visa desenvolver e integrar elementos visuais que possibilitem o aperfeiçoamento do processo de ensino e aprendizagem de programação por parte dos alunos surdos.

2. Fundamentação Teórica

2.1. O aluno surdo e o processo de aprendizado

Nas últimas décadas, a concepção vigotskiana de desenvolvimento cognitivo têm ganho destaque no contexto da educação do aluno surdo, visto que aponta para a importância da relação pensamento-linguagem no desenvolvimento humano. Contudo, apesar da preocupação com as questões educacionais, pouco foi aplicado dos resultados teóricos obtidos dentro do âmbito escolar. Ao abordar a relação pensamento-linguagem, Vygotski destaca a importância da linguagem para a comunicação entre as pessoas e, principalmente, para o desenvolvimento do indivíduo, como ser pensante [VYGOTSKI 1993].

Neste contexto, entende-se, dentro da metodologia proposta, que para o processo pleno de ensino da programação não bastam apenas a língua de sinais (intérprete) e o material didático tradicional, num processo de conversão fiel do conteúdo lecionado para língua de sinais. Em outras palavras, o processo de ensino da programação exige um método alternativo que permita estabelecer uma comunicação eficiente entre o professor e o aluno. Esta comunicação pode ocorrer, por exemplo, por meio da criação de um glossário simbólico, que estaria presente tanto no material didático utilizado quanto no(s) software(s) de apoio ao ensino.

2.2. O aluno surdo no IFPR–Paranaguá

O IFPR–Paranaguá tem recebido alunos surdos desde 2016, tantos nos cursos médio integrado quanto nos cursos superiores da área de informática. No que tange o ensino de programação, todos apresentaram, além dos desafios inerentes ao processo de aprendizado desta disciplina, grande dificuldade de compreensão sobre a escrita do código-fonte. Adicionalmente a isso, os alunos surdos demonstram pouca capacidade de interpretação de texto. O material de aula tradicional, utilizado com sucesso até então dentre os alunos ouvintes, não é capaz de transmitir o conteúdo como o esperado. Em outras palavras, apenas a atuação dos intérpretes, ainda que imprescindível para o processo de educação, não se mostra, por si só, suficiente para contornar esse cenário.

Todas essas adversidades impossibilitam que os alunos surdos compreendam a funcionalidade e aplicação dos comandos básicos da linguagem C. Ademais, as dificuldades mostram-se ainda maiores quando é exigido o desenvolvimento da lógica necessária

para criação dos algoritmos que resolvem os problemas propostos. Combinada a impossibilidade do ensino da programação através dos moldes tradicionais, os alunos surdos apresentam relevante dificuldade em compreender o que os exercícios disponibilizados propõe como problemática. Isto é, antes mesmo que possam aplicar os recursos da linguagem os alunos não possuem entendimento sobre o que de fato precisa ser codificado.

2.3. Softwares didáticos e metodologia proposta

Pesquisas recentes mostram que o uso de ferramentas visuais é uma metodologia capaz de atrair a atenção dos alunos, possibilitando que compreendam com maior facilidade o conteúdo abordado e consigam desenvolver mais rapidamente suas habilidades. Neste contexto, o uso de ferramentas visuais dentro do processo de aprendizado do aluno tem como resultados positivos: tornar as tarefas prazerosas e desafiadoras, abranger a dimensão simbólica e não limitar as possibilidades do aluno [MACEDO et al. 2003].

Ainda que o uso das ferramentas visuais (softwares didáticos), tais como SCRATCH [OLIVEIRA et al. 2014], VisuAlg [APOIO INFORMÁTICA 2019] e ProgLib [SANTOS et al. 2011], facilite o processo de aprendizado, estas ferramentas não possuem ligação direta com o material didático utilizado pelo professor, condição que pode ser um diferencial positivo para o processo de ensino de programação. Por isso, a metodologia descrita na seção 3 propõe a criação e integração entre um material didático adaptado (conteúdo e exercícios) e um software didático para ensino de programação, denominado Visual Programmer (ViP). O ViP é utilizado como parte da metodologia e não apenas como uma ferramenta de auxílio ao ensino.

Diante disto, os símbolos definidos para representação dos comandos da linguagem, dentro do material didático criado, são os mesmos usados para criação do Visual Programmer. Nessa mesma lógica, as atividades e exercícios propostos são trabalhados de modo integrado, e o processo de aprendizado é dividido em etapas, desde o reconhecimento inicial sobre a estrutura da linguagem, tornando-a amigável, até o desenvolvimento da lógica para construção dos códigos-fontes.

A partir dessa ideia são apresentados na seção 3 os modelos de materiais e exercícios (baseados em símbolos) aplicados de modo integrado ao ViP, juntamente com todos os componentes educacionais propostos pela MDS.

3. Metodologia Didático Simbólica – MDS

A concepção da metodologia proposta tem como objetivo tornar o processo de aprendizado de programação, por parte dos alunos surdos matriculados nos cursos de informática do IFPR–Paranaguá, mais intuitivo. Desta forma, o uso da abordagem didática tradicional é evitado, visto que o mesmo tem por base a adoção do texto puro em língua portuguesa. Essa escolha norteia-se no fato de que os alunos surdos possuem a LIBRAS como primeira língua, e não o português. Tal característica impossibilita a leitura e interpretação de textos complexos. No que diz respeito ao uso da simbologia, ela justifica-se pelo fato dos alunos surdos mostrarem maior capacidade de aprendizado quando a comunicação visual é a base do processo de ensino [LEBEDEFF 2010].

Além dos objetivos descritos, tem-se como meta principal tornar o aluno surdo capaz de criar suas aplicações através da utilização da linguagem de programação definida para disciplina ao qual está cursando. Em outras palavras, o material simbólico criado,

assim como o software projetado para programação simbólica, são ferramentas que possibilitam compreender a lógica por trás dos comandos e da sintaxe da linguagem. Neste contexto, deseja-se não apenas que aluno desenvolva o raciocínio lógico via utilização de ferramentas que permitem a programação visual, mas sim, que sejam capazes de quebrar as primeiras barreiras educacionais do processo de ensino. Espera-se, ainda, que eles sejam preparados para lidar com a complexidade das linguagens e estejam prontos para ingressar no mercado de trabalho.

Portanto, a Metodologia Didático Simbólica propõe a divisão do processo de aprendizado de programação em três etapas: (1) Compreensão do Conceito Estudado; (2) Aplicação do Conceito Estudado; e (3) Compreensão da Sintaxe da Linguagem. As etapas 2 e 3, mais complexas, são subdivididas em aplicação parcial e total do conceito, e compreensão parcial e total da sintaxe. Além das etapas, também foram definidos seis recursos didáticos que compõem cada uma delas: (a) Glossário de Símbolos; (b) Material Didático Simbólico; (c) Exercícios Simbólicos; (d) Exercícios Fonte; (e) Software Visual Programmer (ViP); e (f) Editor de Código / Terminal de Comandos – ECTC.

A figura 1 apresenta o fluxo de aprendizado para a MDS, levando em consideração suas etapas de ensino e os recursos didáticos utilizados em cada uma delas:

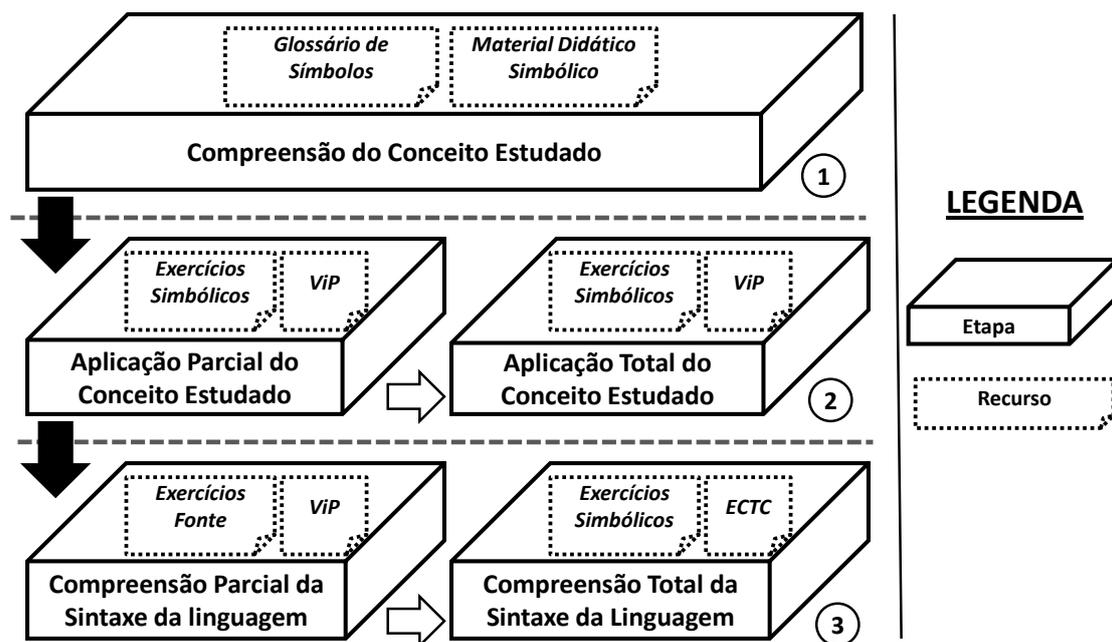


Figure 1. Esquema de Etapas e Recursos da Metodologia Didático Simbólica.

Analisando o esquema disposto é possível visualizar cada uma das três etapas do processo de ensino proposto pela Metodologia Didático Simbólica. O esquema descreve, ainda, quais recursos didáticos são utilizados dentro de cada uma das três etapas. A primeira etapa, *Compreensão do Conceito Estudado*, faz uso dos recursos didáticos “Glossário Simbólico” e “Material Didático Simbólico”, a segunda etapa, *Aplicação do Conceito Estudado*, faz uso dos recursos didáticos “Exercício Simbólico” e “Visual Programmer – ViP”, a terceira e última etapa, *Compreensão da Sintaxe da Linguagem*, faz uso dos recursos didáticos “Exercício Simbólico”, “Exercício Fonte”, “Visual Program-

mer – ViP” e “Editor de Código e Terminal de Comandos – ECTC”.

Na seção 3.1 são descritos, em detalhes, cada um dos seis recursos propostos pela MDS.

3.1. Recursos Didáticos

A MDS define como recursos didáticos: “*os instrumentos de ensino necessários e aplicados em cada uma de suas três etapas, que permitem atingir os seguintes objetivos: (1) transposição das já conhecidas barreiras que envolvem o ensino de programação; e (2) superação, por parte dos alunos surdos, das dificuldades apresentadas quando submetidos ao processo de ensino tradicional*”. Todos os seis recurso didáticos foram projetados tendo como base a comunicação visual, estabelecida através da criação de uma linguagem simbólica, especificada através de um glossário e aplicada dentro de todos os outros recurso concebidos: material didático, exercício e software educacional.

Glossário Simbólico e Material Didático Simbólico

O primeiro recurso criado para concepção da MDS foi o Glossário Simbólico, visto que ele permite estabelecer uma comunicação visual com o aluno surdo. O Glossário é elemento fundamental para introdução dos principais termos técnicos e comandos da linguagem, que por sua vez são essenciais durante o processo de ensino de programação. Essa estratégia é adotada porque os nomes dos comandos básicos da linguagem C não são auto explicativos, impossibilitando a compreensão por parte dos discentes. A partir do Glossário foi criado o Material Didático Simbólico, tendo como base a simbologia definida nesse primeiro recurso. O Material Didático Simbólico possibilita que o aluno surdo compreenda o conteúdo de programação a partir de uma abordagem visual, em contrapartida ao processo de ensino tradicional onde o material é baseado, fundamentalmente, em textos explicativos. A Figura 2 exemplifica, resumidamente, o formato dos recursos didáticos descritos.

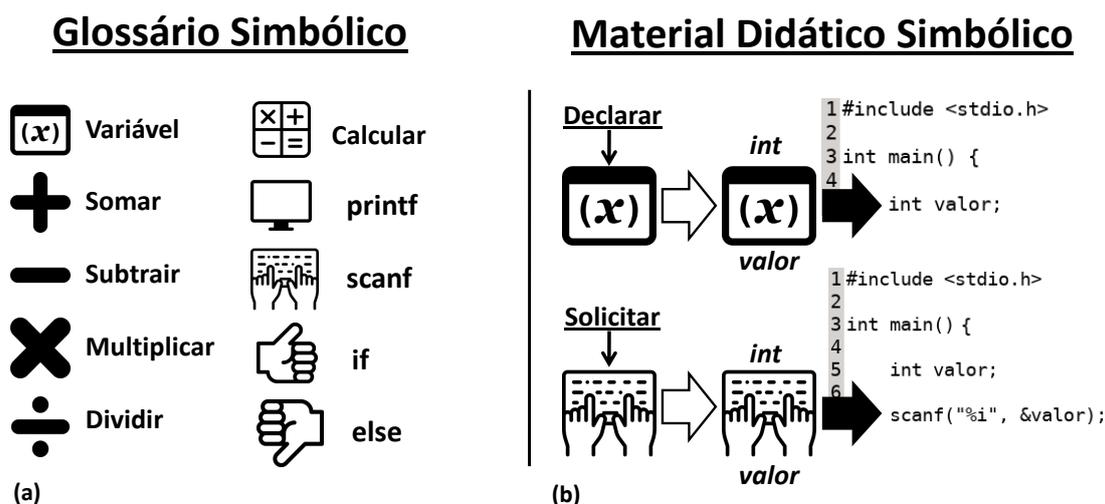


Figure 2. Recursos Didáticos: (a) Glossário Simbólico; (b) Material Didático Simbólico.

No glossário (Figura 2a) os conceitos de programação como variáveis e comandos da linguagem C (Ex.: “printf()” e “scanf()”) são vinculados à símbolos, possibilitando

que o aluno surdo compreenda sua aplicabilidade dentro do código-fonte. Por exemplo, as variáveis são representadas por uma caixa retangular porque este símbolo remete a capacidade de armazenar algo, exatamente a mesma função das variáveis em programação. Já os comandos “*printf()*” e “*scanf()*” são representados por uma tela de computador e um teclado porque possuem como funcionalidade, respectivamente, apresentar e receber informações via saída (monitor) e entrada (teclado) padrão do sistema de computação.

O material didático (Figura 2b) tem por base os símbolos definidos no Glossário, provendo comunicação visual e possibilitando a compreensão do conteúdo. Essa abordagem permite ao aluno surdo dar seus primeiros passos no universo da programação. A Figura 2b) mostra a declaração da variável “valor” com tipo “int” via linguagem simbólica e sua respectiva tradução para o código-fonte C. O mesmo procedimento é exibido para o comando “*scanf()*”. Em ambos os exemplos o nome da variável e seu tipo são especificados juntamente com o símbolo, permitindo a criação de uma codificação simbólica, projetada para ser usada em conjunto com outro recurso didático da MDS, o software Visual Programmer.

Exercícios Simbólicos e Visual Programmer

Os Exercícios Simbólicos permitem a criação de programas visuais baseados nos recursos didáticos anteriores. Os Exercícios Simbólicos possibilitam que o aluno surdo ingresse no processo de aprendizado de programação sem que possua conhecimento prévio sobre os detalhes da sintaxe da linguagem. Eles possibilitam, ainda, dividir a etapa de Aplicação dos Conceitos Estudados em dois momentos. No primeiro, as atividades práticas são passadas com a lógica do algoritmo já estruturada. No segundo, o aluno deve criar a lógica a partir do Exercício Simbólico proposto. Para que a MDS atinja os objetivos dessa etapa, um novo recurso didático, capaz de converter código simbólico em linguagem C, foi concebido – Visual Programmer (ViP). A Figura 3 exemplifica, resumidamente, o formato dos recursos didáticos descritos.

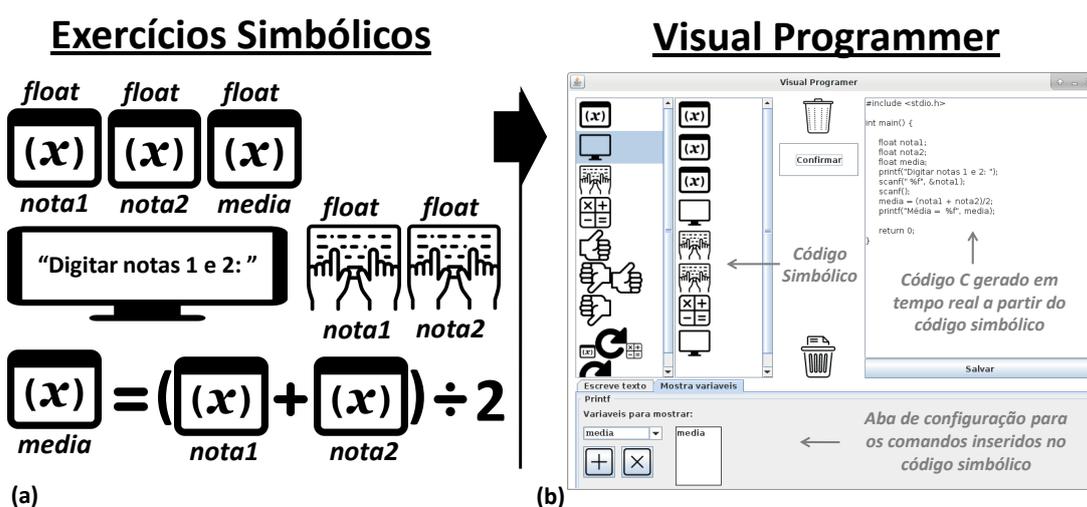


Figure 3. Recursos Didáticos: (a) Exercícios Simbólicos; (b) Visual Programmer.

Os Exercícios Simbólicos (Figura 3a) apresentam programas construídos com base nos símbolos definidos no Glossário. Através deles o aluno surdo deve utilizar o

Visual Programmer (Figura 3b) para criar o código simbólico correspondente. O ViP permite a criação de codificação visual, utilizando os mesmos símbolos do Glossário e do Material Didático Simbólico. A codificação visual é convertida pelo ViP em tempo real para linguagem C. É desta maneira que os recursos didáticos criados são unificados ao longo de todas as etapas de ensino propostas pela MDS. Importante destacar, que nessa fase do processo de ensino ainda não é exigido do aluno o raciocínio lógico necessário para construção dos algoritmos. Isso ocorre porque o objetivo da MDS, nesse momento, é tornar o aluno surdo capaz de construir a atividade proposta dentro do Visual Programmer (segunda coluna). Como o ViP converte em tempo real a linguagem simbólica em linguagem C, o aluno tem dentro do seu campo visual as duas linguagens dispostas lado a lado. Essa abordagem permite, ao aluno surdo, vincular os símbolos estudados (Material Didático) com os comandos da linguagem C. Além disso, o aluno surdo tem a possibilidade de ver o código simbólico construído em execução, compreendendo na prática a funcionalidade de cada comando/símbolo, visto que o ViP salva, compila, abre o terminal de comandos e executa o programa criado – isso ocorre quando o botão “Salvar” é pressionado. É nessa fase que o processo de compreensão sobre a sintaxe da linguagem tem seu início.

Exercícios Fonte e ECTC

Os Exercícios Fonte têm por objetivo dar continuidade ao processo de aprendizado da sintaxe da linguagem. Neles, o aluno surdo é estimulado a analisar um código-fonte C, procedimento que lhe torna apto a compreender a estrutura e os comandos da linguagem. A partir do código-fonte o aluno deve construir a codificação simbólica correspondente dentro do ViP, comparando o código C gerado pelo software com o apresentado no Exercício Fonte. Ao efetuar adequadamente esse procedimento o aluno ingressa na etapa final do processo de ensino proposto pela MDS. Nesta fase, ele utiliza os Exercícios Simbólicos sem estruturação lógica e os recursos didáticos Editor de Código e Terminal de Comandos (ECTC). A Figura 4 exemplifica, de modo resumido, o formato desses recursos.

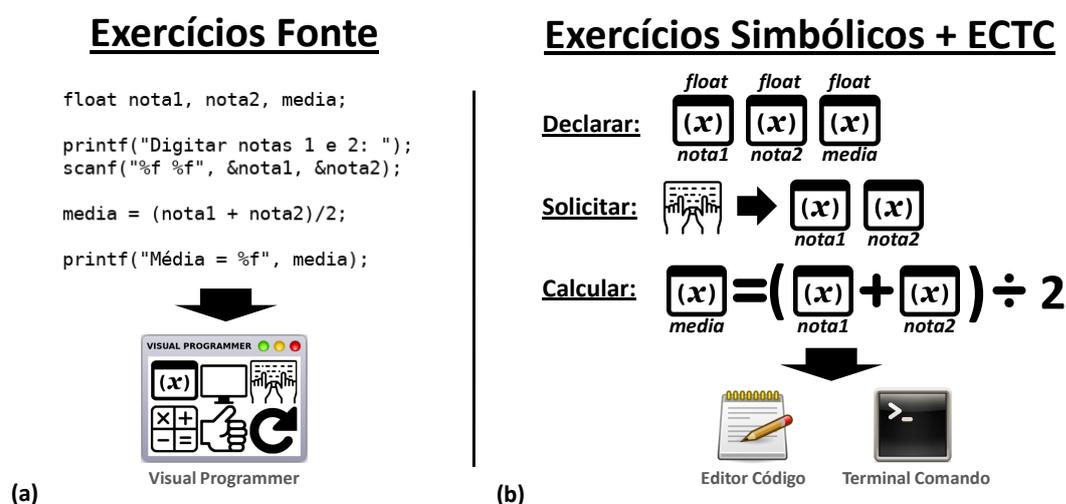


Figure 4. Recursos didáticos: (a) Exercícios Fonte; (b) ECTC.

Os Exercícios Fonte (Figura 4a) são codificados pelos alunos surdos via lin-

guagem simbólica disponibilizada no Visual Programmer. Os Exercícios Simbólicos (Figura 4b) substituem os enunciados tradicionais usados por docentes por um modelo que permite manter a comunicação visual com o surdo – abordagem utilizada pela MDS em todos níveis do processo de aprendizado. A implementação em linguagem C e execução do exercício devem ser feitas no Editor de Códigos e Terminal de Comandos. Nesse momento o aluno está por finalizar a etapa de Compreensão da Sintaxe da Linguagem.

Ao final do processo de ensino proposto pela MDS, o aluno surdo possui o conhecimento necessário sobre a sintaxe e os comandos da linguagem C estudados até o momento. Agora, ele já é capaz de interpretar e resolver, com maior nível de compreensão, problemas apresentados no formato textual, como no caso da Figura 5. Em outras palavras, ele está pronto para desenvolver algoritmos que são soluções para problemas reais, tendo como fonte especificações semelhantes as que encontrará quando for profissional da área de computação e estiver desempenhando suas atividades de programador.



Figure 5. Exercício Textual – Pós conclusão de todas as etapas da MDS.

Na figura 5 é possível observar que os símbolos definidos no Glossário, e utilizados pelos outros recursos didáticos, são omitidos nesse momento, dando espaço ao enunciado textual. Contudo, e tendo como objetivo manter um canal parcial de comunicação visual, é disponibilizado junto ao enunciado, um esquema figurativo que exemplifica o problema abordado. A remoção definitiva da comunicação visual deve ser adotada pelo docente no momento em o mesmo detectar que ela já não é mais necessária ao aluno surdo.

4. Aplicação da Metodologia Didático Simbólica

Atualmente, o IPFR–Paranaguá possui três alunos surdos matriculados no Ensino Médio Integrado de Informática, cada qual ingresso de um ano letivo diferente. Entretanto, devido as dificuldades de aprendizado de programação encontradas por todos eles, todos ainda encontram-se cursando a disciplina de Linguagem de Programação do primeiro ano. Diante dessa dificuldade, a MDS foi criada por professores, alunos do ensino médio e tradu-

tores e intérpretes de libras. A MDS vem sendo aplicada desde o início de fevereiro, exclusivamente para os três alunos surdos.

Os conteúdos abordados até o momento são: variável e tipo de variável; operadores matemáticos; comandos de entrada e saída; e comandos de condição. As três etapas de ensino e seus respectivos recursos didáticos são aplicados para cada conteúdo abordado pelo professor. Por exemplo, as três etapas de ensino da MDS foram aplicadas para o processo de aprendizado dos comandos de entrada e saída, após seu término e com os alunos compreendendo efetivamente tal conteúdo foi dado início ao estudo dos comandos de condição. Nesse momento, o processo de aprendizado volta a primeira etapa de ensino da MDS para o novo conteúdo abordado, retomando o ciclo.

4.1. Resultados Preliminares

Os resultados coletados inicialmente foram mensurados em três dimensões: (1) receptividade do aluno surdo para com a nova metodologia simbólica proposta; (2) otimização do processo de comunicação e transmissão do conteúdo; (3) melhoria do desempenho escolar do aluno. O primeiro item foi avaliado através da utilização dos materiais didáticos tradicionais em paralelo ao material didático simbólico. Esse procedimento foi efetuado em uma mesma aula, tanto para apresentação do conteúdo quanto para proposição de exercícios. Os alunos foram questionados sobre qual dos recursos fazia com que se sentissem mais confortáveis para aprender programação, a resposta foi unânime em afirmar que a linguagem simbólica aprimora substancialmente a forma de compreensão sobre o conteúdo lecionado.

O segundo item foi avaliado pelo próprio professor tendo como base a dificuldade encontrada pelos alunos para abstrair os conceitos de programação quando submetidos aos dois tipos materiais didáticos, o textual tradicional e simbólico adaptado. Essa análise foi feita via questionamentos colocados para os alunos surdos para as duas abordagens de ensino, tradicional e MDS. Constatou-se que a comunicação visual proposta pelo MDS permitiu uma melhora significativa sobre a compreensão e abstração de conceitos complexos que envolvem o processo de ensino de programação. O terceiro e último item foi avaliado via comparação do desempenho obtido pelos alunos surdos em anos anteriores, através da aplicação de atividades e avaliações dentro dos mesmos moldes (mesma complexidade). Mais uma vez os discentes apresentaram desempenhos melhores, evoluindo de conceitos que, em anos anteriores, eram D ou C para conceitos C, B e A.

5. Conclusão

O processo de aprendizado de programação tem se mostrado desafiador, tanto para docentes quanto discentes dos mais variados níveis de ensino. Além disso, contextos específicos, com novos fatores de dificuldade de compreensão, são encontrados em maior frequência nas instituições de ensino federal. Isso deve-se ao crescente ingresso de alunos que apresentam alguma necessidade especial. Esses fatores recorrem à necessidade de concepção de novas metodologias de ensino, que possam transpor as barreiras apresentadas, superando os desafios e possibilitando o aprendizado dos discentes.

A partir deste contexto, foi projetada e construída uma nova metodologia de ensino de programação voltada à alunos surdos, tendo como base a comunicação visual. Esta metodologia foi denominada Metodologia Didático Simbólica (MDS). A MDS está

sendo utilizada atualmente no IFPR–Paranaguá, curso Médio Integrado de Informática. Ela tem obtido resultados iniciais satisfatórios, já que os alunos tem apresentado maior poder de compreensão e abstração se comparado ao desempenho obtido em anos anteriores. Como trabalhos futuros tem-se a aplicação da MDS na construção do material didático dos conteúdos do segundo semestre da disciplina, juntamente com um portal que já disponibiliza o material adaptado criado até o momento e o software ViP. Ele pode ser acessado em <http://www.gileduardo.com.br/ifpr/lps>

References

- APOIO INFORMÁTICA, A. (2019). Visualalg. In *Disponível em: <http://www.apoioinformatica.inf.br/produtos/visualg>. Acesso em: 12 de março de 2019.*
- BRASIL (1997). Decreto - lei 2.208, de 17 de abril de 1997. regulamenta o § 2º do art. 36 e os arts. 39 a 42 da lei nº 9.394, de 20 de dezembro de 1996, que estabelece as diretrizes e bases da educação nacional. In *Disponível em: <http://www.planalto.gov.br/>.*
- CAVALCANTI, A. M. B. and MENEZES, R. D. (2016). A inclusão de alunos surdos no ifpe campus pesqueira: Um pequeno passo para a garantia do direito à educação. In *II CINTEDI: II Congresso Internacional de Educação Inclusiva (II Jornada Chilena Brasileira de Educação Inclusiva)*.
- GOMES, M. C. P. (2015). Os benefícios do ensino de linguagem de programação no currículo regular. In *Disponível em: <http://www.administradores.com.br/artigos/carreira/os-beneficios-do-ensino-de-linguagem-de-programacao-no-curriculo-regular/89064/>. Acessado em: 10 de Março de 2019.*
- LEBEDEFF, T. B. (2010). Aprendendo a ler “com outros olhos”: relatos de oficinas de letramento visual com professores surdos. In *Pelotas: FaE/PPGE/UFPeL*.
- MACEDO, L., PETTY, A. L. S., and PASSOS, N. C. (2003). Os jogos e o lúdico na aprendizagem escolar.
- OLIVEIRA, M. L. S. d., SOUZA, A. A. d., BARBOZA, A. F., and BARREIROS, E. F. S. (2014). Ensino de lógica de programação no ensino fundamental utilizando o scratch: Um relato de experiência. In *XXXIV Congresso da Sociedade Brasileira de Computação - CSBC*.
- SANTOS, R. E. S., Magalhães, C. V. C., Neto, J. S. C., and JÚNIOR, S. S. L. P. (2011). Proglib: Uma linguagem de programação baseada na escrita de libras. In *XVII Workshop de Informática na Escola - WIE*.
- VYGOTSKI, L. S. (1993). Pensamento y language: las raices genéticas del pensamiento y el language. In *Obras escogidas II. Madrid: Centro de Publicações del MEC y Visor Distribuciones*.