

## **Relato de Experiência sobre a Utilização da Computação Física no Ensino de Algoritmos e Lógica de Programação no IFPR**

**Olavo José Luiz Junior<sup>1</sup>**

<sup>1</sup>Instituto Federal de Educação, Ciência e Tecnologia do Paraná (IFPR)  
Av. Cívica, 475 – 85.935-000 – Assis Chateaubriand – PR – Brasil

olavo.junior@ifpr.edu.br

***Abstract.** This paper reports experience in Computer Maintenance & Support technical course in IFPR, campus Assis Chateaubriand. The aim of the experiment was to investigate the efficiency of the concept of Physical Computing to improve cognitive outcome of the student with respect to the knowledge and development of algorithms and programming logic. The preliminary results are promising.*

***Resumo.** Este artigo relata experiência no curso técnico em Manutenção e Suporte em Informática do IFPR, câmpus Assis Chateaubriand. O objetivo da experiência foi investigar a eficiência do conceito de Computação Física para melhorar o resultado cognitivo dos estudantes no que se refere ao conhecimento e desenvolvimento de algoritmos e lógica de programação. Os resultados parciais apresentados são promissores.*

### **1. Introdução**

A tradicional dificuldade de raciocínio lógico-matemático dos alunos de nossa região que ingressam em cursos de informática apresenta aos professores do IFPR continuamente um desafio: como melhorar a capacidade de abstração de alguns conceitos fundamentais da área de Algoritmos e Lógica de Programação? Em reunião de curso, os professores da área de informática discutiram essa questão, e o autor sugeriu o trabalho piloto em uma disciplina onde se pudesse aplicar práticas que utilizem o conceito de Computação Física para verificar sua eficiência no sentido de reduzir essas dificuldades e ampliar o índice de aprendizado.

A disciplina de algoritmos e lógica de programação, ou equivalente, tradicionalmente é conduzida através de exemplos hipotéticos de problemas a serem resolvidos pelo aluno. Em uma primeira fase o estudante raciocina e desenvolve sua solução através de técnicas como o português estruturado. Em uma segunda fase, o aluno deverá aplicar sua resolução em uma linguagem de programação escolhida pelo currículo do curso, dessa forma sendo iniciado tanto em lógica, como em programação. Ocorre que os exemplos muitas vezes são muito abstratos e dissociados da realidade, pois como os programas devem ser pequenos para se adequar à baixa complexidade esperada nesse nível de aprendizado, não se consegue criar situações com a concretude suficiente para facilitar o entendimento do aluno sobre todo o processo envolvido nesse desenvolvimento. Isso se reflete em baixa eficiência no aprendizado da disciplina, no alto índice de reprovações na mesma, no despreparo parcial dos que avançam, e na consequente demonização da disciplina de algoritmos.

Em todos os níveis de ensino da computação, os alunos já chegam com o estigma da dificuldade da disciplina de algoritmos, e relatam isso mesmo antes de tomar contato

com a mesma, como vê-se em estudos com em GIRAFFA e MORA (2013). Tem-se então que trabalhar em alternativas que desmistifiquem, facilitem, melhorem o resultado final, e, porque não, tornem agradável para todos o aprendizado de algoritmos e lógica de programação. Entendemos que o conceito e as práticas da Computação Física podem auxiliar muito nesse sentido, substituindo os problemas abstratos por problemas concretos, melhorando assim a execução da segunda fase descrita acima, ou seja, a da programação.

Segundo CAMARATA, *et al.* (2003), a Computação Física (*Physical Computing*) permite a interação entre o homem e a máquina através de um microcontrolador, que é um dispositivo eletrônico dotado de processamento próprio que, acoplado a sensores e atuadores, permite a interface com objetos do mundo real. Essa interface permite interação com o ambiente, capturando dados do mesmo, e tomando decisões através de programas desenvolvidos com uma linguagem de programação própria. As decisões dos programas podem agir no ambiente, comandando pequenos sistemas mecânicos e/ou eletrônicos.

## 2. Percurso Metodológico

No currículo deste curso, a disciplina de Programação Aplicada à Automação (PAA) se encontra no terceiro semestre e utiliza e retoma os conhecimentos adquiridos na disciplina de Lógica de Programação (LP), ministrada no primeiro. Assim, considera-se que os alunos possuam o pré-requisito para desenvolver pequenos programas. A constatação da ainda grande dificuldade em lógica de programação apresentada pela turma em PAA, originou a discussão entre os professores sobre as alternativas para melhorar o resultado da disciplina de LP em próximas turmas. Como o melhor parâmetro que os professores teriam para aferir o resultado de qualquer nova iniciativa seria mais fidedigno se aplicado em uma turma que vivenciasse as duas práticas pedagógicas, o autor propôs sugerir e utilizar uma nova prática na turma corrente da disciplina de PAA. O maior objetivo seria posteriormente aferir a percepção dos mesmos quanto ao resultado comparado das duas metodologias no seu aprendizado, embasando assim uma posterior discussão no colegiado de informática para uma possível mudança no currículo de LP, caso a prática se mostrasse eficiente. Assim, para a utilização dos conceitos de Computação Física, o professor de PAA apresentou na disciplina a identificação de componentes de circuitos eletrônicos básicos e a identificação das características técnicas destes componentes. Diferentes componentes eletrônicos como resistores, *leds*<sup>1</sup>, sensores de diversas finalidades, foram apresentados de maneira conceitual e prática aos alunos, através da utilização de *kits* da plataforma de prototipação Arduino<sup>2</sup>, a fim de mostrar os componentes disponíveis para as atividades práticas. Pela facilidade de uso do Arduino, e pelo nível das atividades a serem desenvolvidas, essa fase de apresentação teve a duração de apenas quatro horas/aula, não onerando o encadeamento geral da disciplina.

Após isso foram desenvolvidos pequenos trabalhos propostos em McROBERTS (2012), onde os alunos deveriam desenvolver na primeira fase um código em português estruturado, e na segunda fase a programação e compilação deste código na linguagem *Processing*<sup>3</sup> na IDE (*Integrated Development Environment*) do Arduino. Depois da programação, passou-se para a montagem física no microcontrolador Arduino, sempre em duplas ou trios, por conta da quantidade de *kits* disponíveis para uso. Com o

1 *Light-emitting Diode*, espécie de “lâmpada” digital (USBORNE, 1985).

2 Plataforma de prototipação em *hardware* livre e de baixo custo (BANZI, 2010).

3 Linguagem de programação oriunda das linguagens C e C++, utilizada na plataforma Arduino.

protótipo montado, a próxima fase é colocá-lo em funcionamento alimentando-o com energia, embarcar o código no microcontrolador, e verificar a eficiência do código embarcado através do funcionamento do protótipo. Se na prática algum resultado é diferente do esperado, os alunos passam a depurar o código até que o protótipo funcione de acordo com a composição original.

Inicia-se sempre com pequenos exemplos de manipulação de *leds*, onde se espera o acendimento e/ou apagamento dos mesmos. Esse exemplo clássico, torna-se o tradicional “*Hello World!*” na Computação Física. Após os trabalhos iniciais de reprodução de exemplos com pequenas alterações propostas pelo professor, passa-se para projetos mais elaborados, relacionados com situações do mundo real, e em que todas as fases sejam desenvolvidas pelos alunos. Para exemplificar, um dos projetos propostos e que desenvolve bem o uso dos conceitos de variáveis, tipos de dados, estruturas de repetição e estruturas de decisão, é o de um dispositivo que simula o funcionamento real de um sensor de estacionamento automotivo. Seguem abaixo as premissas do comportamento do protótipo, que fazem parte do enunciado do problema a ser resolvido com o projeto:

- Distâncias maiores que 15cm: nada a sinalizar;
- Distâncias entre 11 e 15cm: acender uma luz de alerta laranja no painel do carro;
- Distâncias entre 6 e 10cm: acender uma luz de alerta vermelha no painel do carro;
- Distâncias menores que 5cm: acionar um intermitente sinal sonoro de alerta.

O resultado esperado da montagem física para este exercício, com o protótipo já em funcionamento, é apresentado na Figura 1, criada a partir de foto da experiência feita em sala de aula. Para simular as luzes do painel foram utilizados *leds*, nem sempre laranjas e vermelhos, por não existirem em número suficiente. Para o sinal sonoro foi utilizado um *buzzer*, e para capturar a distância entre o carro (Arduino) e a parede (mão do aluno se movimentando), foi utilizado um sensor de presença.

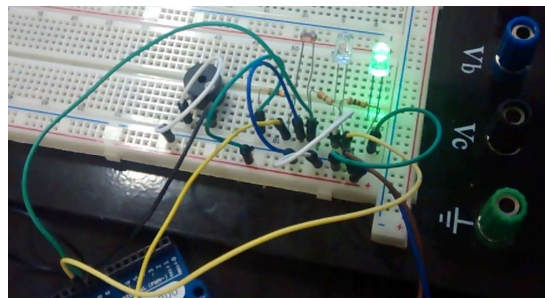


Figura 1: Protótipo de Sensor de Estacionamento (fonte: o autor)

### 3. Resultados e Discussão

Se a programação não estiver correta, um fato concreto não ocorre. A verificação se dá através de uma inspeção visual no funcionamento do protótipo. Se, por exemplo, um *led* não acende como era esperado, fica mais fácil a identificação entre a causa e o efeito de um erro na lógica de programação. Foi relativamente comum os alunos não perceberem que na transição entre as faixas de distâncias para sinalização tinham que desligar o alerta anterior. Assim, na execução do teste real do protótipo, na passagem de 11 para 10cm, os dois *leds* ficaram acesos, e o aluno percebeu imediatamente que o algoritmo deveria ter previsto o desligamento do *led* laranja antes de acender o *led* vermelho, e procedeu com facilidade a modificação do código no local correto para resolver essa inconformidade com as premissas propostas para o programa. Depois de encerradas todas as atividades, o autor aplicou um questionário para investigar a visão dos alunos

quanto às práticas vivenciadas. Foi composto por questões qualitativas e quantitativas. A questão inicial foi:

*Na sua opinião, se na disciplina de linguagem de programação fosse utilizado o kit arduino para aprender algoritmos e lógica de programação, seu aprendizado de lógica ia ser:*  
 *mesma coisa*       *mais fácil*       *mais difícil*

Nenhum aluno marcou a opção *mais difícil*. 22% dos alunos assinalaram *mesma coisa*, não entendendo assim, que houve maior facilidade para o aprendizado com a utilização do *kit*. Apesar do questionário ter garantido anonimato, um dos alunos que marcou essa opção procurou voluntariamente o professor para dizer que para ele não houve diferença porque a sua grande dificuldade está no pensamento lógico e na montagem das proposições lógicas ainda no fase do português estruturado. Que a partir de ter o português estruturado pronto, a montagem no *kit* é interessante e motivante, mas que ele não verifica maior facilidade de aprendizado por conta disso. A opção *mais fácil* foi assinalada por 78% dos alunos. O índice é considerável, porém respostas oriundas das questões qualitativas foram ainda mais interessantes. Algumas respostas apontaram uma maior facilidade em ver o resultado do algoritmo, e uma maneira mais divertida de aprender algoritmos. Além disso, alguns sugeriram em suas respostas que seria melhor desenvolver individualmente as atividades com a utilização de um *kit* Arduino por aluno, demonstrando um interesse em efetivo na técnica.

#### 4. Conclusão

A utilização dos conceitos da Computação Física na criação de algoritmos e pequenos programas proporcionou, para a maioria dos alunos, um entendimento que vai além de visões abstratas, e pode traduzir situações reais, o que facilita o desenvolvimento do raciocínio lógico fundamental para o domínio de algoritmos e lógica de programação. Essa prática permitiu aos alunos tratar os algoritmos e a lógica de programação como algo mais tangível e a visualizar a solução de diversos outros problemas da vida real, percebidos e propostos por eles mesmos na disciplina, como por exemplo, a programação de dispositivos para automação de aviários e para uma pequena estação meteorológica. A experiência desse trabalho apresentou grande potencial de sucesso para o aprendizado de nossos alunos. Trabalhamos agora para viabilizar a aquisição de *kits* para multiplicar seu uso em sala de aula e estudar melhor seus resultados.

#### Referências

- BANZI, Massimo. (2011). **Primeiros Passos com o Arduino**. São Paulo: Novatec.
- CAMARATA, Ken *et al.* (2003). *A Physical Computing Studio: Exploring Computational Artifacts and Environments*. International Journal of Architectural Computing, v. 1, n. 2, p. 169-190.
- GIRAFFA, Lucia Maria Martins; MORA, Michael da Costa (2013). **Evasão na Disciplina de Algoritmo e Programação: Um Estudo a Partir dos Fatores Intervenientes na Perspectiva do Aluno**. In: III CLABES. Madrid: EUIT de Telecomunicação, 2013. v. 1. p. 188-197.
- McROBERTS, Michael. (2011). **Arduino Básico**. São Paulo: Novatec.
- USBORNE. **Introdução à Eletrônica**. (1985). Rio de Janeiro: Lutécia.