

## Proposta de Atividades para o Desenvolvimento do Pensamento Computacional no Ensino Fundamental\*

Daiane Andrade, Tainã Carvalho, Jayne Silveira,  
Simone Cavalheiro, Luciana Foss, Ana Marilza Fleischmann,  
Marilton Aguiar, Renata Reiser

<sup>1</sup>Centro de Desenvolvimento Tecnológico – Universidade Federal de Pelotas (UFPel)  
Rua Gomes Carneiro, 1 – 96.010-610 – Pelotas – RS – Brasil

{dfandrade, trcarvalho, simone.costa}@inf.ufpel.edu.br

**Abstract.** *Computational thinking is the thought process involved in formulating problems and their solutions, based on the foundations of Computer Science. This paper proposes a set of activities for elementary students, which comprises through a ludic way nine concepts considered fundamental for the development of computational thinking in schools.*

**Resumo.** *Pensamento computacional é o processo de pensamento envolvido na formulação de problemas e das suas soluções, tendo como base os fundamentos da Ciência da Computação. Este artigo propõe um conjunto de atividades para alunos do ensino fundamental, as quais abordam, de forma lúdica, nove conceitos da Computação considerados fundamentais para o desenvolvimento do pensamento computacional nas escolas.*

### 1. Introdução

A combinação do pensamento crítico com os fundamentos da computação define uma metodologia para resolver problemas, denominada pensamento computacional (PC) [Wing 2006]. Esta visão defende que a cultura do computador ajuda a sociedade não somente a aprender, mas especialmente oferece uma nova maneira de aprender a aprender [Lu and Fletcher 2009, Papert 1980]. O PC é um método para solução de problemas que utiliza os fundamentos e técnicas da Ciência da Computação. De forma geral, habilidades comumente utilizadas na criação de programas computacionais para resolver problemas específicos são utilizadas como uma metodologia para resolver problemas nas mais diversas áreas [Bundy 2007, Denning 2009, Nunes 2011].

Empresas multinacionais apoiam a proposta do PC e se empenham na disseminação de sua metodologia. Em 2007, a Microsoft e a Universidade de Carnegie Mellon criaram o Centro de Pensamento Computacional [Carnegie Mellon 2013]. A Google, por sua vez, tem se empenhado em promover esta metodologia em todo o currículo do ensino primário e secundário nos Estados Unidos [Google 2013].

Com base no princípio de que todos os alunos devem demonstrar competência nas habilidades básicas do PC na conclusão do ensino médio, em 2010, as organizações CSTA (Computer Science Teachers Association), ISTE (International Society for Technology in Education) e a NSF (National Science Foundation) propuseram um conjunto de

---

\*Projeto com o apoio do PROEXT - MEC/SESu

ferramentas, denominadas *Computational Thinking toolkit* [CSTA et al. 2010], com o objetivo de desenvolver as habilidades do PC na educação primária e secundária dos EUA. O material proposto pelas organizações citadas conta com um quadro de progressão, no qual nove conceitos inerentes à computação são eleitos como fundamentais para o desenvolvimento do PC nas escolas: coleta, análise e representação de dados, decomposição de problemas, abstração, algoritmos, automação, simulação e paralelismo.

Tendo em vista este quadro de progressão e nos resultados obtidos em experiências prévias, entende-se como importante desenvolver uma metodologia adequada para a promoção do PC no âmbito do ensino fundamental e médio no Brasil. Um passo inicial consiste em trazer os fundamentos da Ciência da Computação para a educação, desde o nível básico. A principal contribuição deste artigo é a descrição de um conjunto de três atividades voltadas ao nível fundamental, sendo duas delas propostas como jogos educacionais nomeadas Cara a Cara e Caça ao Tesouro e a última uma atividade de integração denominada Organização de Festa. A metodologia empregada visa desenvolver habilidades específicas relacionadas ao PC. Elas trabalham de forma lúdica os nove conceitos propostos no *Computational Thinking toolkit*.

O artigo está organizado como segue. A seção 2 apresenta uma breve introdução ao pensamento computacional e a seção 3 descreve o *Computational Thinking toolkit*. Na seção 4 são propostas atividades que visam desenvolver habilidades específicas do pensamento computacional. Na seção 5 são apresentadas algumas considerações finais.

## 2. Pensamento Computacional

O pensamento computacional (do inglês, *computational thinking*) é um método para solução de problemas baseado nos fundamentos e técnicas da Ciência da Computação [Wing 2006]. Sua proposta é aplicar habilidades utilizadas para criação de programas computacionais como metodologia para resolução de problemas gerais.

Por exemplo, na computação, *hashing* constitui-se de um método intuitivo, que permite organizar grandes quantidades de dados e possui como ideia central a divisão de um universo de dados em subconjuntos mais gerenciáveis. No cotidiano, o processo de organizar porcas e parafusos em caixas e seções apropriadas também constitui uma forma de *hashing*. Conforme colocado pelo professor Danny Sleator em [Wing 2011], a função *hashing* utilizada pelas crianças para organizar seus blocos de lego influencia diretamente no tempo necessário para que elas encontrem a peça apropriada.

Conforme apresentado em [Wing 2006], o PC pode ser colocado como uma das habilidades intelectuais básicas de um ser humano, comparada a ler, escrever, falar e fazer operações aritméticas. Habilidades estas que servem para descrever e explicar situações complexas. Nesta linha de raciocínio, o PC é mais uma linguagem (como a linguagens escrita e a matemática) que pode ser utilizada para as pessoas falarem (descreverem) sobre o Universo e seus processos complexos.

O PC é considerado como um processo de solução de problemas que inclui (mas não se limita a) características tais como [ISTE 2013]: *i*) formular problemas de modo que seja possível usar um computador para ajudar a resolvê-los; *ii*) organizar os dados logicamente de modo que, futuramente, seja possível analisá-los; *iii*) representar dados através de abstrações, tais como modelos e simulações; *iv*) criar for-

mas de automatizar as soluções através do pensamento algorítmico; *v*) permitir identificar, analisar e implementar soluções possíveis, com o objetivo de conseguir a combinação mais eficiente e eficaz de etapas e recursos; e, *vi*) generalizar e transferir este processo de solução de problemas a uma variedade de outros problemas.

Existem três pilares básicos que fundamentam o PC: **abstração**, **automação** e **análise**. A **abstração** é a capacidade de extrair apenas as características importantes de um problema para chegar a sua solução, levando em consideração que as demais já foram solucionadas. Problemas com grau de complexidade grande tornam-se difíceis de serem resolvidos, caso não haja uma abstração adequada. A **automação** é a utilização de um meio eletrônico na substituição do trabalho manual. Um computador é um bom exemplo de meio eletrônico que pode substituir o trabalho de um ser humano. No entanto, ele não será útil se não forem dadas as instruções certas para a execução da tarefa desejada. E, por fim, a **análise** é o estudo dos resultados gerados pela automação. Se eles não saírem como o esperado, há a possibilidade de não ter sido escolhido o nível de abstração adequado ou até mesmo de não ter sido planejada a solução automatizada corretamente.

Diversos programas e projetos [Barr and Stephenson 2011, Bell et al. 2010, Carnegie Mellon 2013, França et al. 2012, Google 2013, Hambruch et al. 2009, Lee et al. 2011, Kafura and Tatar 2011, Repenning et al. 2010] têm se preocupado em propor atividades para o desenvolvimento do PC nos mais diversos níveis de ensino. No Brasil, podemos destacar dois projetos voltados à introdução do PC no âmbito do ensino básico: o *Game Design Brasil*, vinculado a Pontifícia Universidade Católica do Rio de Janeiro, que tem como objetivo o desenvolvimento de uma tecnologia brasileira para estimular o ensino-aprendizado de computação em escolas de ensino fundamental e médio [PUC-Rio 2013]; e o *Computação Desplugada*, que tem como meta propagar os fundamentos da ciência da computação sem o uso de computador, para alunos de escolas públicas [Scaico et al. 2012].

### 3. *Computational Thinking Toolkit*

Iniciado em 2010, o *Computational Thinking in K-12 Education Leadership Toolkit* é um conjunto de ferramentas desenvolvido por líderes de diferentes escolas, em conjunto com a *Computer Science Teachers Association* (CSTA), *International Society for Technology in Education* (ISTE) e *National Science Foundation* (NSF) [CSTA et al. 2010]. Está baseado no princípio de que todos os alunos devem demonstrar competências nas habilidades básicas do PC na conclusão do ensino médio.

O material contido no *toolkit* pode ser dividido em duas partes: *i*) uma fundamentação teórica para o processo de desenvolvimento do PC, discutindo o seu potencial e a sua importância em todos os níveis de ensino; *ii*) uma documentação dos recursos necessários para uma mudança sistemática, descrevendo uma definição operacional para o PC, criando um vocabulário e uma tabela de progressão para as habilidades envolvidas. Além disso, apresenta um modelo para uma mudança sistemática, propondo um guia com estratégias de implementação e levantando questões a serem discutidas pelas comunidades interessadas.

Dentre todo o material apresentado pelo *toolkit*, destaca-se o quadro de progressão, que mostra nove conceitos da área da computação entendidos como fundamentais de serem trabalhados para o desenvolvimento do PC. Cada um dos conceitos podem ser

definidos como segue. A **coleta de dados** é o processo de reunir dados de forma apropriada. A **análise de dados** é o passo que objetiva tornar os dados coletados coerentes, encontrando padrões e tirando conclusões a partir destes dados. A **representação de dados** é o processo de organizar apropriadamente as informações por meio de tabelas, gráficos, palavras, imagens ou qualquer outro recurso disponível. A **decomposição de problemas** é capacidade de divisão das tarefas em partes menores e manuseáveis. A **abstração** é a redução da complexidade de um problema para focar na questão principal. **Algoritmos e procedimentos** são definidos como uma possível série organizada de passos para resolver um problema ou atingir algum objetivo. A **automação** é a utilização de computadores ou máquinas para fazer tarefas repetitivas ou tediosas. A **simulação** é a representação ou a modelagem de um processo e a sua execução. E o **paralelismo** é a forma de organizar recursos para simultaneamente desenvolver tarefas que atinjam um objetivo em comum.

Partindo desses conceitos, são propostas atividades que envolvem cada um deles para as mais diversas faixas de ensino. Estas atividades são listadas mas não possuem uma descrição detalhada ou mesmo uma proposta metodológica. O objetivo no quadro é apenas de relacionar os conceitos com possibilidades de tarefas que os desenvolvam.

#### 4. Proposta de atividades para o ensino fundamental

Nesta seção são propostas três atividades que visam desenvolver habilidades específicas do PC. As atividades foram especificadas de forma a englobar todos os conceitos enfatizados no quadro de progressão apresentados no *toolkit*. Estas atividades não pretendem esgotar estes conceitos, mas sim introduzi-los para turmas de ensino fundamental que saibam ler, escrever e conheçam as operações aritméticas básicas.

##### 4.1. Atividade I: Cara a Cara

O jogo infantil Cara a Cara [Estrela 1986] foi escolhido como modelo de inspiração para a criação de uma atividade lúdica, voltada ao ensino de crianças do 4º ano do ensino fundamental. Os conceitos escolhidos para serem trabalhados nessa atividade foram **coleta**, **análise** e **representação de dados**, além do conceito de **abstração**. A **coleta de dados**, do quadro de progressão do *toolkit*, é abordada na criação de uma ficha padronizada de características dos alunos. Os alunos devem levantar características individuais presentes nos estudantes da turma, que possam ser representadas em uma ficha, modelando atributos tais como altura, comprimento de cabelo, etc. Em um segundo momento, cada aluno deve preencher a sua ficha. Então, em conjunto a turma deve especificar os valores válidos para os campos/atributos propostos. Não será permitida qualquer quantificação (ou qualificação) nos atributos da ficha, apenas uma quantidade arbitrária de valores (ou classes) para cada campo, fazendo com que os alunos utilizem da sua **abstração** para definir alternativas válidas que englobem todas as respostas levantadas. Após a parte lúdica do jogo é possível então trabalhar com o conceito de **análise de dados**, trabalhando com as informações resultantes da efetuação da atividade. Através de uma discussão sobre o que foi observado durante o jogo, deve-se tirar algumas conclusões sobre o que foi ou não efetivo para se obter respostas durante o jogo. Para trabalhar a **representação de dados**, a partir das conclusões tiradas, incita-se os alunos a pensarem em formas de mostrar visualmente as conclusões feitas. As formas de abordagem e de execução dessas atividades são explicitadas a seguir.

O jogo que originou esta atividade explora a coleta de dados de forma indireta. Cada um dos participantes possui o mesmo conjunto de personagens com aparências diferentes entre si e cada um recebe um cartão com um dos personagens. Cada jogador deve descobrir, através de perguntas, quem é o personagem do adversário. As perguntas servem para eliminar personagens que não estão no possível grupo de respostas, e a coleta de dados permite também trabalhar com outros conceitos para manipulação de informações, como será visto ao longo da descrição da proposta.

#### **4.1.1. Início da atividade**

**Discussão e conceitos necessários.** Para que a coleta de dados seja introduzida e haja a construção conjunta da atividade, é necessário começar uma discussão sobre características. Inicialmente, utiliza-se um objeto representativo, como uma mesa ou uma cadeira, e pergunta-se sobre o que é observável nesse objeto, sobre como é possível distingui-lo dos demais objetos existentes, descrevendo-o. Essa discussão deve levar ao conceito de característica, determinando quais são as características relevantes para se obter com sucesso a descrição de algo.

**Ficha de coleta.** Tendo em vista o conceito de característica, deve ser feita a criação de uma ficha comum para a autodescrição do aluno. Para isso, é iniciada uma nova discussão, a respeito das características possíveis de serem observadas nos colegas, como cor dos olhos, cor dos cabelos, textura dos cabelos, entre outras. Após um consenso sobre quais características são relevantes, parte-se para elaboração das fichas. É importante ressaltar que a turma deve acordar com o uso de simplificações, como por exemplo usar um conceito (baixo, médio e alto) ao invés de um valor numérico para altura ou agrupar diferentes tons da mesma cor de cabelo em apenas uma cor. Gerar essas simplificações trabalha o conceito de abstração durante o levantamento dos dados, desenvolvendo-o no jogo em si. A ficha pode conter dados sobre a personalidade, para tornar mais completa a descrição da pessoa. É importante que todos os alunos cheguem a um senso comum entre as características a serem utilizadas. Elas serão uma forma de descrever cada um individualmente para gerar uma linguagem em comum, por isso é importante que todos estejam em sintonia com as informações. Após a ficha ser gerada, ela deve ser preenchida individualmente por cada aluno.

#### **4.1.2. Execução do jogo**

Uma vez que todas as fichas estejam devidamente preenchidas, a turma deve ser dividida em grupos para que um grupo jogue contra o outro. Cada grupo deve debater entre seus membros qual pessoa da turma será a escolhida para ser o objetivo do grupo adversário. Essa decisão também pode ser aleatória, mediante a influência do instrutor que estiver aplicando a atividade. Após a escolha feita, é a hora de jogar. Cada grupo deve fazer uma única pergunta por rodada e através das mesmas começar a etapa de exclusão de possibilidades do conjunto de possíveis respostas até que a pessoa seja identificada corretamente. O jogo deve ser repetido algumas vezes para que as crianças possam reconhecer os padrões das características que facilitam a descoberta da resposta.

### 4.1.3. Análise dos resultados

Após o término do último jogo, deve-se iniciar uma discussão sobre as perguntas realizadas, questionando o critério que determinou a escolha das perguntas e qual conjunto de perguntas excluía mais rapidamente um grande grupo de possibilidades para que o aluno escolhido fosse achado. Não há recomendação da sequência de perguntas a seguir, permitindo que varie de acordo com a turma que está sendo aplicada a atividade.

A discussão deve englobar questões sobre *i*) a ordem de perguntas: qual era a melhor maneira de chegar na resposta ou qual era a ordem menos efetiva para chegar ao resultado positivo; ou *ii*) sobre as características da turma: qual era a mais comum ou aquela que facilitava na eliminação de respostas erradas. Após essa discussão, o passo seguinte é gerar uma tabela com as características mais e menos presentes entre os alunos.

Todas essas discussões geram o conceito de análise de dados. As crianças devem tirar conclusões sobre as informações que obtiveram no jogo e na coleta das características, seja para traçar padrões ou para ressaltar diferenças na turma.

Com a etapa de análise de dados pronta, deve-se propor que os alunos transformem os dados em informações visuais, como um painel com as características e uma quantidade de estrelas indicando o quão comum elas são, por exemplo, para a representação de dados ser posta em prática. Então, finaliza-se a atividade.

## 4.2. Atividade II: Caça ao Tesouro

O jogo de Caça ao Tesouro é uma atividade que gera movimentação e entretenimento. Jogos do tipo caça ao tesouro trabalham com mapas que contêm caminhos para se chegar até um tesouro escondido. Nestes caminhos, há uma série de passos a serem seguidos. Adaptando este princípio para o desenvolvimento do PC, o mapa seria constituído de um espaço amplo, potencialmente uma quadra poliesportiva, e o cenário poderia possuir obstáculos elaborados pelos próprios alunos, com objetivo de explorar a descrição algorítmica de passos a serem seguidos para se atingir um objetivo, no caso, encontrar o tesouro.

Inicialmente, deve-se dar um mapa amostral para a turma e, após a execução do mesmo, incitar os alunos a criarem seus próprios caminhos e obstáculos para chegar até o tesouro. Para que não haja um número excessivo de mapas na turma, é aconselhável dividir a turma em pequenas equipes.

Os conceitos referentes ao *toolkit* escolhidos para serem trabalhados são **algoritmos e procedimentos** e **simulação**. **Algoritmos e procedimentos** estão caracterizados na atividade pela criação de mapas pelos alunos, que deve ser inspirado no mapa amostral previamente discutido com a turma. A ideia é que os alunos criem seu próprios mapas com *i*) *estruturas condicionais*, onde diferentes respostas podem levar a diferentes caminhos; e, *ii*) *laços de repetição*, onde são determinados obstáculos a serem vencidos para prosseguimento do caminho ao tesouro. Já o conceito de **simulação** é abordado na execução do jogo tanto no mapa amostral quanto nos mapas propostos pela turma.

### 4.2.1. Início da atividade

**O mapa do jogo.** O mapa amostral contextualizado em uma quadra de esportes deve conter um ponto de partida onde o jogo deve iniciar, um ponto de chegada que representa

o tesouro escondido e um conjunto arbitrário de obstáculos entre os dois pontos. O ponto crucial nessa etapa são os obstáculos. Os obstáculos devem introduzir os conceitos de laços condicionais e sentenças condicionais para direcionar o caminho que deve ser seguido até o tesouro. Aqui, sugere-se a realização dos obstáculos utilizando um jogo de dados, com um número arbitrário de dados e de lados. Deve-se especificar uma condição a ser satisfeita, para que a equipe ultrapasse o obstáculo. Se a equipe conseguir satisfazer esta condição, recebe uma nova pista. Essa nova pista pode ser de dois tipos: *i*) continuar o caminho para o tesouro, indo para um novo obstáculo; ou, *ii*) um objetivo a ser alcançado para prosseguir no caminho. Por exemplo, se for utilizado um dado de vinte lados, pode-se estipular que o valor mínimo daquele obstáculo é 13. Logo, a equipe deve tirar um número igual ou superior a 13 para receber a pista. O número de tentativas possível à equipe fica a critério da especificação do obstáculo. Além disso, pode-se sugerir que seja pago algum tipo de preço para o obstáculo para adquirir a pista. Essa forma de obstáculo necessita de algum tipo de moeda *in-game* para poder ser implementada. É importante salientar que a utilização de dados torna desnecessário o uso de perguntas pré-prontas, onde uma equipe pode tomar proveito da resposta da outra, e trabalha com a sorte, onde uma equipe pode conseguir transpassar o obstáculo na primeira tentativa e outra pode não conseguir tão rápido. Cada uma dessas estruturas de obstáculos pode ser adaptada para o espaço real de aplicação.

**Definição das equipes e apresentação do mapa amostral.** Sugere-se que a turma seja dividida em um número de equipes igual ao número de obstáculos. Como exemplo, podemos considerar um mapa com 5 (cinco) obstáculos, apropriados para uma turma de 25 (vinte e cinco) alunos, considerando 5 membros por equipe. As equipes devem ultrapassar cada um dos obstáculos respeitando a sequência do mapa. Cada obstáculo leva para o próximo obstáculo se for acertada a condição de ultrapassagem. No final do percurso está o tesouro. A demarcação dos objetivos pode ser feita através de desenhos no chão ou como o professor preferir, de acordo com os recursos disponíveis. Essas equipes devem possuir um líder que pode ser escolhido através de sorteio ou por indicação dos próprios membros da equipe. A figura do líder é importante para determinar as escolhas do grupo e direcionar o trabalho posterior de criação do mapa. Para a implementação sugerida, será utilizado o sistema de moeda *in-game*, que servirá como forma de ultrapassagem dos obstáculos. Essa moeda poderá ser elaborada em cartolina com um valor simbólico e cada integrante da equipe deve receber seis unidades desta moeda. No início do jogo, além do mapa amostral, cada membro do grupo recebe suas moedas.

#### 4.2.2. Execução do jogo

Após a apresentação da atividade e a distribuição das equipes, com a definição dos respectivos líderes, deve-se decidir, utilizando os dados, a ordem de jogo das equipes para, assim, cada uma percorrer o caminho identificado no mapa. Neste exemplo, o líder deve manter o registro, para cada obstáculo, do valor resultante do jogo de dados de cada equipe, o qual permitirá ou não que ela ultrapasse o obstáculo. No caso da equipe passar mais de uma vez pelo mesmo obstáculo, somente o maior valor será registrado.

No primeiro obstáculo, a equipe encontrará cinco dados de seis lados. Cada membro da equipe deve jogar um dos dados. Se a soma dos valores resultar em um valor maior do que 14, é permitido à equipe ultrapassar o obstáculo, em caso contrário, a equipe deve

retornar ao início e aguardar, no final da fila, sua vez para nova tentativa. No segundo obstáculo, enquanto a equipe não obtiver exatamente o número 7 no dado de oito lados, não poderá prosseguir. No terceiro obstáculo, para prosseguir no caminho a equipe deverá obter em um dado de vinte lados, um valor maior ou igual a 12. A equipe pode tentar no máximo cinco vezes e deverá pagar uma moeda por tentativa. Se ao final da quinta tentativa a condição não for satisfeita, a equipe paga mais duas moedas e pode seguir no caminho. No quarto obstáculo, a equipe terá dois dados de doze lados. Se a soma dos dois dados for abaixo de 10, a equipe deve voltar para o segundo obstáculo; se obtiver entre 10 e 16 pode-se rolar os dados de novo sem custo algum e acima de 16 pode-se prosseguir para o último obstáculo. O último obstáculo do jogo envolve os valores registrados pelo líder. A equipe deve somar os valores registrados para cada obstáculo. Se o valor for abaixo de 60, eles devem dar metade das moedas que possuem; se o valor for entre 60 e 71, a equipe deve dar vinte e cinco por cento das moedas que restaram e se o valor for acima de 71 não há pagamento de moedas. Após o pagamento, a equipe chega no tesouro. Após todas as equipes chegarem ao final do percurso, é feita a contagem de suas moedas. A equipe que finalizar o jogo com a maior quantidade de moedas, ganha o tesouro.

#### 4.2.3. Análise dos resultados

**Elaboração de um novo mapa.** Após o término do jogo, deve-se manter os alunos em equipes e motivá-los a criar seus próprios mapas. O ponto de partida e o ponto de chegada de cada equipe devem ser estabelecidos pelo professor. As equipes devem receber auxílio do professor para elaborar os obstáculos. Os obstáculos podem ser baseados nos apresentados durante o jogo e o *layout* do caminho pode ser modificado.

**Simulação.** Com os mapas de cada equipe prontos, deve-se fazer a troca de mapas entre as equipes e estruturar a atividade de forma que enquanto uma ou mais equipes simularem um mapa, a equipe que o elaborou faça a fiscalização de forma a garantir que as instruções sejam executadas corretamente.

**Discussão.** Após todos os mapas serem simulados, deve-se levantar algumas questões com os alunos, referente aos obstáculos. Uma das questões importantes é discutir como foi a etapa de criação, como foram estruturadas as novas condições ou o rearranjo do mapa para tornar os obstáculos diferenciados. Em relação aos mapas propostos, deve-se analisar se algum possui um conjunto de obstáculos que sempre permita que a equipe encontre o tesouro, ou se algum deles poderá levar a equipe a ficar presa. Nessa etapa, deve-se também definir os conceitos de procedimentos e condições de repetição, e de qual forma elas estão associadas ao cotidiano.

#### 4.3. Atividade III: Organização de uma festa

A ideia desta atividade é trabalhar de maneira lúdica com os conceitos de **decomposição de problemas**, **paralelismo** e **automação**. A atividade consiste em organizar uma festa. A **decomposição de problemas** e o **paralelismo** são abordados na forma de dividir a organização da festa em pequenas tarefas que devem ser executadas em paralelo por diferentes grupos de alunos. Já a **automação** será trabalhada através do uso de meios eletrônicos para substituir trabalhos que seriam executados manualmente.



#### 4.3.1. Início da atividade

Inicialmente, é importante discutir com os alunos a metodologia de organização a ser adotada. Primeiramente, deve-se questionar como é possível organizar uma festa de forma eficiente. Essa discussão, moderada pelo professor, deve levar os alunos à conclusão que dividir essa atividade em pequenas tarefas será a melhor forma de organização. Outra questão a ser considerada são os tipos de tarefas a serem realizadas para organização da festa. O objetivo é que as tarefas fiquem bem especificadas para que não haja problemas de execução no decorrer do processo, ou sobreposição de responsabilidades.

#### 4.3.2. Execução da Atividade

Depois da divisão da organização da festa em tarefas, os alunos estarão aptos a executar a atividade. Um grupo de estudantes será associado a cada tarefa a ser executada. É importante que cada estudante faça parte de um (único) grupo, de forma que a execução das tarefas possa ser simultânea. Para que a organização da festa seja concluída, todas as tarefas devem ter finais sincronizados. Vale salientar da importância do uso de equipamento de dispositivo eletrônico para a aplicação da automação, considerando que esta tem como objetivo substituir o trabalho manual com uso de dispositivos. Por exemplo, pode-se utilizar um *notebook* para escolher a *playlist* das músicas que serão reproduzidas na festa.

#### 4.3.3. Análise dos Resultados

Após a etapa de execução, deve-se retomar as discussões em relação a qual melhor forma de organizar os alunos para cumprir as tarefas de modo eficiente, convergindo para a conclusão sobre a eficácia de dividir as tarefas em pequenos subgrupos.

### 5. Considerações finais

Neste artigo foram propostas três atividades que visam abordar de forma lúdica conceitos fundamentais da Ciência da Computação. Particularmente, as atividades visam introduzir os conceitos de coleta, análise e representação de dados, decomposição de problemas, abstração, automação, simulação, algoritmos e paralelismo, sem o uso de computadores.

Considera-se esta proposta mais um passo em direção a disseminação do PC no ensino fundamental, tornando-o acessível a comunidade escolar. Trabalhos futuros visam definir uma metodologia adequada para o desenvolvimento do PC nas diversas séries do ensino fundamental no Brasil.

### Referências

- Barr, V. and Stephenson, C. (2011). Bringing computational thinking to k-12: what is involved and what is the role of the computer science education community? *ACM Inroads*, 2(1):48–54.
- Bell, T., Witten, I. H., and Fellows, M. F. (2010). *Computer Science Unplugged*.
- Bundy, A. (2007). Computational thinking is pervasive. *Journal of Scientific and Practical Computing*, 1:67–69.

- Carnegie Mellon (2013). Center for Computational Thinking. <http://www.cs.cmu.edu/~CompThink/>. Acesso: maio/2013.
- CSTA, ISTE, and NSF (2010). Computational thinking leadership toolkit. <http://www.csta.acm.org/Curriculum/sub/CompThinking.html>. Acesso: agosto/2013.
- Denning, P. J. (2009). The profession of it: Beyond computational thinking. *Commun. ACM*, 52(6):28–30.
- Estrela (1986). Cara a cara. <http://www.estrela.com.br/cara-a-cara>. Acesso: agosto/2013.
- França, R. S., Silva, W. C., and Amaral, C. J. H. (2012). Ensino de ciência da computação na educação básica: Experiências, desafios e possibilidades. In *Anais do XXXII Congresso da Sociedade Brasileira de Computação*.
- Google (2013). Exploring computational thinking. <http://www.google.com/edu/computational-thinking/>. Acesso: maio/2013.
- Hambrusch, S., Hoffmann, C., Korb, J. T., Haugan, M., and Hosking, A. L. (2009). A multidisciplinary approach towards computational thinking for science majors. In *Proc. 40th Tech. Symp. on Comp. Sci. Education*, pages 183–187, New York, USA. ACM.
- ISTE (2013). Operational definition of computational thinking. <http://www.iste.org/learn/computational-thinking/ct-operational-definition>. Acesso: maio/2013.
- Kafura, D. and Tatar, D. (2011). Initial experience with a computational thinking course for computer science students. In *Proc. 42nd Technical Symposium on Computer Science Education*, pages 251–256, New York, USA. ACM.
- Lee, I. et al. (2011). Computational thinking for youth in practice. *ACM Inroads*, 2(1):32–37.
- Lu, J. J. and Fletcher, G. H. (2009). Thinking about computational thinking. In *Proc. 40th Technical Symp. on Comp. Sci. Education*, pages 260–264, New York, USA. ACM.
- Nunes, D. J. (2011). Ciência da computação na educação básica. <http://www.jornaldaciencia.org.br/Detalhe.jsp?id=79207>. Acesso: maio/2013.
- Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas*. Basic Books, Inc., New York, NY, USA.
- PUC-Rio (2013). Scalable Game Design Brasil. <http://www.serg.inf.puc-rio.br/wiki/index.php>. Acesso: agosto/2013.
- Repenning, A., Webb, D., and Ioannidou, A. (2010). Scalable game design and the development of a checklist for getting computational thinking into public schools. In *Proc. 41st Tech. Symp. on Comp. Sci. Education*, pages 265–269, New York, USA. ACM.
- Scaico, P. D. et al. (2012). Um relato de experiências de estagiários da licenciatura em computação com o ensino de computação para crianças. *RENOTE - Revista Novas Tecnologias na Educação*, 10(3).
- Wing, J. M. (2006). Computational thinking. *Commun. ACM*, 49(3):33–35.
- Wing, J. M. (2011). Research notebook: Computational thinking - What and Why? The Link. Spring, 2011.