

X REUNIÃO DA ABAVE

Avaliação Educacional no Brasil:
"O DESAFIO DA QUALIDADE"
28 a 30 de agosto de 2019 – São Paulo/SP



CodeMaster

Um Modelo de Avaliação do Pensamento

Computacional na Educação Básica através da Análise

de Código de Linguagem de Programação Visual

CodeMaster

Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual

Home > News & Events > Computer programming and coding in schools

Computer programming and coding in schools — an emerging trend

Inclusão da programação nos currículos escolares avança no exterior

Enquanto outros países já adotaram a área de conhecimento como base curricular, Brasil está na fase

Why Computer Science Education in Settings Is Becoming Increasingly Essential

By ACM, the Association for Computing Machinery

EXAME

Revista EXAME Reforma trabalhista Black Friday Ag

TECNOLOGIA

Escolas da Inglaterra ensinam alunos de 5 anos a programar

Reino Unido reformulou a maneira de ensinar computação às crianças do país adicionando aulas obrigatórias de programação

Por Sam Chambers
16 out 2014, 16h15

Ministério da

Educação

Buscar no portal

Contato Serviços do MEC Área de imprensa

EDUCAÇÃO BÁSICA

CNE debate incluir computação na Base Nacional Comum Curricular

Segunda-feira, 30 de julho de 2018, 18h56



CodeMaster

Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual

Ensino de computação envolve o ensino do **pensamento computacional**

BNCC Base Nacional Comum Curricular

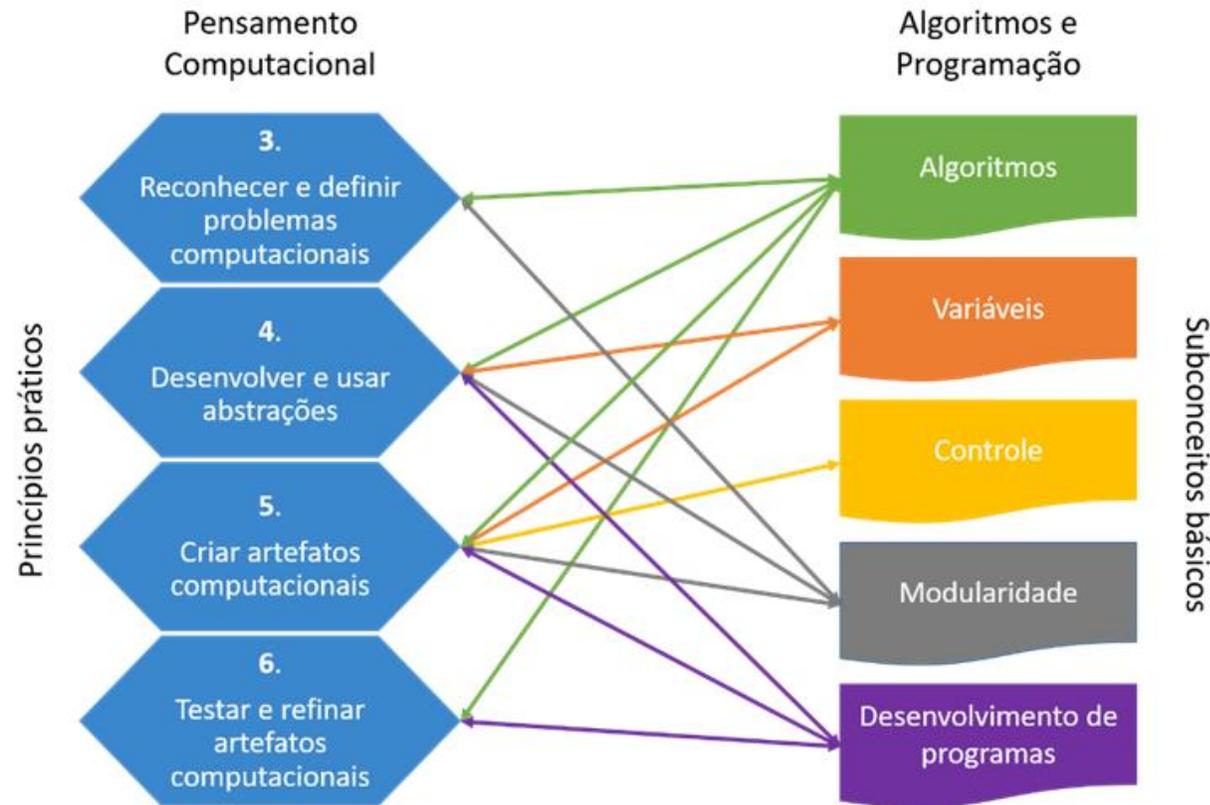
“A área de Matemática, no Ensino Fundamental, centra-se na compreensão de conceitos e procedimentos em seus diferentes campos e no **desenvolvimento do pensamento computacional**, visando à resolução e formulação de problemas em contextos diversos.”

(MEC; BNCC, 2018, p. 471).

CodeMaster

Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual

Foco do ensino do pensamento computacional em **algoritmos e programação**

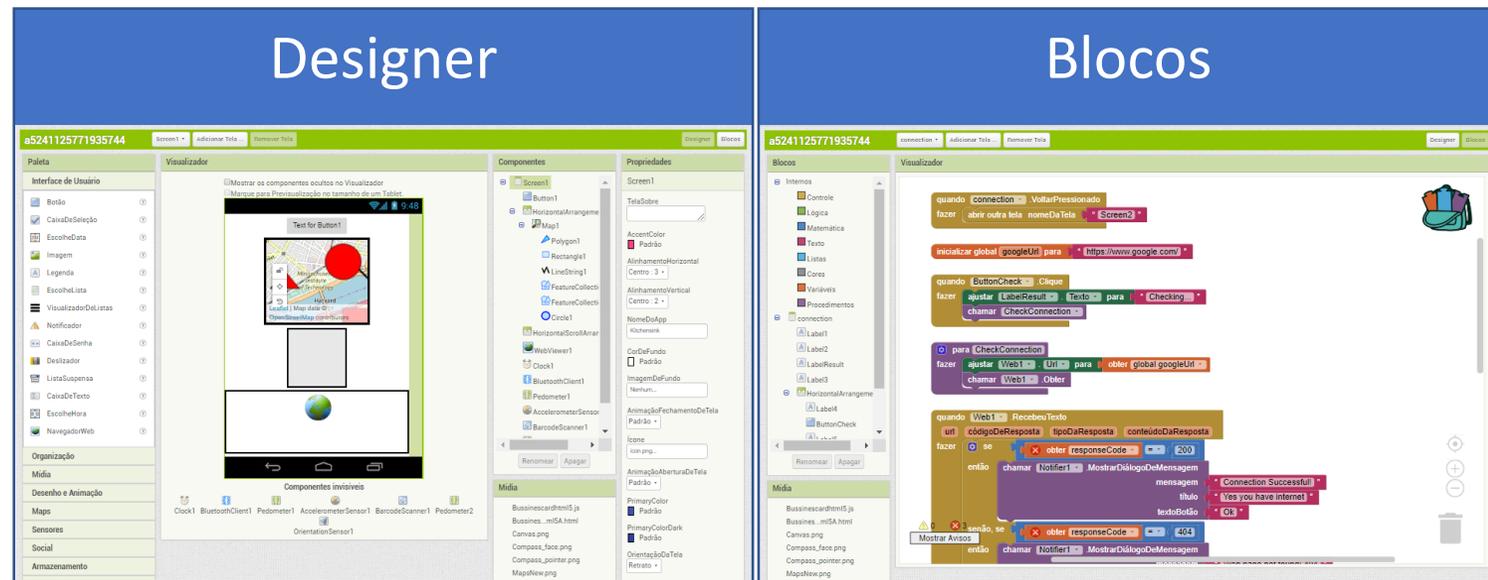


Ensino de algoritmos e programação pode ser por meio do desenvolvimento de aplicativos com App Inventor

App Inventor é uma linguagem de programação visual que permite:

- **Criar aplicativos** para Android
- **Programar** encaixando blocos visuais de comandos
- **Visualizar** em tempo real o aplicativo em qualquer *smartphone* Android

Interface do App Inventor



Avaliação do Pensamento Computacional

O professor define um **tema** em uma **atividade aberta**

O aluno **programa** um aplicativo sobre o tema

O **código-fonte** do aplicativo criado pelo aluno é **analisado**

O aplicativo do aluno é **avaliado** com base nos dados da análise

Exemplo:

Conscientização do combate à Dengue



```
quando clicar em [bandeira verde]
  mude pontos para 0
  sempre
    se tecla espaço pressionada? então
      adicione 10 a y
      mude x para número aleatório entre -180 e 180
      adicione a pontos 1
  se posição x < >150 então
    diga Fim de jogo
    pare todos
```

Critério	Pontuação
Variável	3
Laços	2
...	...
Nota final:	8

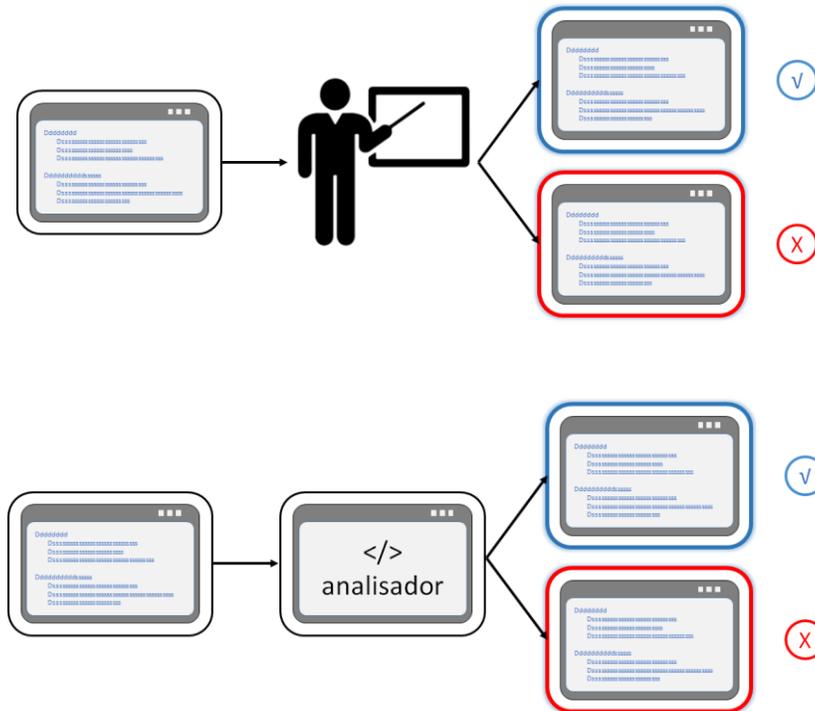
Exemplos de critérios de avaliação de conceitos de representação de dados

...				
P2. Qual o nível de desempenho em representação de dados com relação às práticas do pensamento computacional? (CSTA, 2016; 2017)				
Item	0 pontos	1 ponto	2 pontos	3 pontos
Variáveis: verificar se criou ou modificou valores de variáveis.	Sem uso de variáveis.	Modificação ou uso de variáveis predefinidas.	Criação e operação com variáveis	
Strings: verificar se criou ou modificou valores de strings.	Sem uso de strings.	Uso do comando de criação de string para alterar textos de elementos.	Criação e operação com strings.	
Nomeação: verificar se os nomes de variáveis são alterados do padrão.	Nenhum ou poucos nomes são alterado do padrão. (menos do que 10%)	De 10 a 25% dos nomes são alterados do padrão.	De 26 a 75% dos nomes são alterados do padrão.	Mais de 76% dos nomes são alterados do padrão.
Listas: verificar se são usadas listas.	Não usa listas.	Usa uma lista unidimensional.	Usa mais de uma lista unidimensional.	Usa uma lista de tuplas (map).
...				

CodeMaster

Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual

Por que automatizar?



Avaliação pelo professor

- Tarefa repetitiva
- Diferentes professores enfatizam características diferentes
- Consome esforço e tempo considerável
- Requer conhecimento substancial da computação

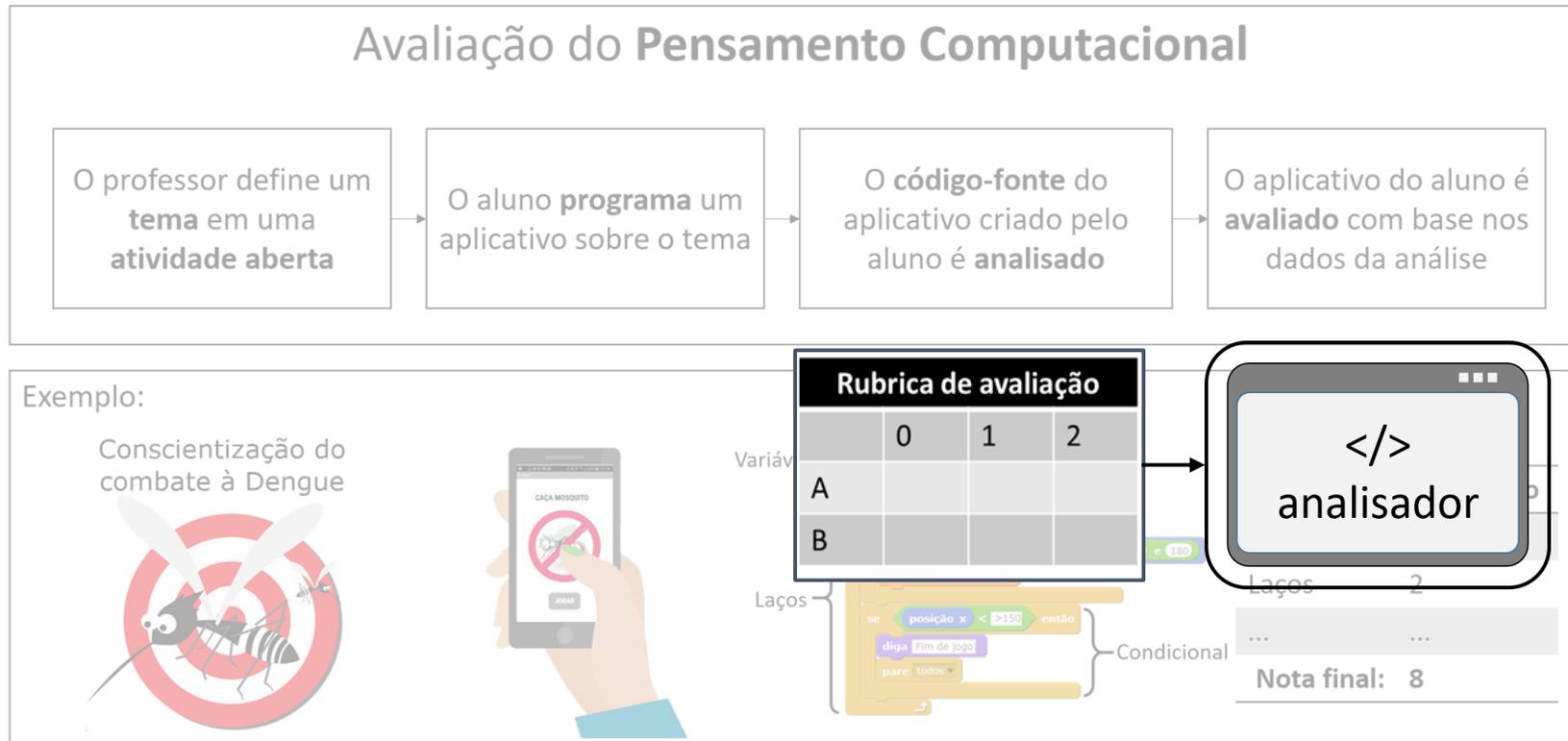
Avaliação por um analisador automatizado

- Consistência
- Rapidez
- Libera o professor para avaliar outros pontos (criatividade) e/ou ajudar o aluno.

CodeMaster

Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual

Como as métricas de avaliação são aplicadas?



X REUNIÃO DA ABAVE

CodeMaster

Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual

O avaliador automatizado CodeMaster

The screenshot shows the CodeMaster interface for a student. The page title is 'Avaliação de projeto App Inventor'. On the left, there is a profile card for a student named 'faixa turquesa' with a score of 6,2. The main area displays a table of evaluation criteria with progress bars. A link at the bottom says 'Clique aqui para descobrir como melhorar sua pontuação!'.

Conceito	Pontuação
Telas	3/3
Nomeação de Componentes	2/3
Eventos	3/3
Abstração de Procedimentos	0/3
Laços	2/3
Condicionais	2/3
Listas	2/3
Persistência de Dados	2/3
Sensores	1/3
Desenho e Animação	0/3
Operadores	3/3
Variáveis	2/2
Strings	2/2
Sincronização	1/1
Mapas	0/2
Extensões	0/1
Total	25/41

The screenshot shows the CodeMaster interface for a teacher. The page title is 'Avaliações de projetos App Inventor'. It features a table with columns for project name, various criteria (Telas, Nomeação de Componentes, etc.), total score, and grade level. A link at the bottom says 'Clique para visualizar a rubrica de avaliação'.

Projeto	Telas	Nomeação de Componentes	Eventos	Abstração de Procedimentos	Laços	Condicionais	Listas	Persistência de Dados	Sensores	Desenho e Animação	Operadores	Variáveis	Strings	Sincronização	Mapas	Extensões	Pontuação total	Nota	Nível
IHLogoQuiz.aia	3	1	3	0	0	0	0	0	0	0	0	2	1	0	0	0	10	2,44	faixa laranja
SchoolClassmateSchedule.aia	3	1	3	0	0	3	2	3	1	0	3	2	2	1	0	0	24	5,85	faixa azul
DolphPireMultifunctionalApplication.aia	3	1	3	3	0	3	2	3	1	0	3	2	2	1	0	0	27	6,59	faixa turquesa
EuroFrance2o16.aia	3	1	3	0	0	1	0	0	1	0	3	2	1	1	0	0	16	3,90	faixa vermelha
GPSFindthatthing.aia	2	2	3	0	0	1	0	2	1	0	3	2	2	0	0	0	18	4,39	faixa roxa
SpaceInvaders.aia	1	2	3	0	0	0	2	0	1	3	3	2	2	1	0	0	20	4,88	faixa roxa
AppInventorCHAT.aia	3	2	3	3	0	3	3	3	1	0	3	2	2	1	0	0	29	7,07	faixa verde
Math.aia	1	2	3	2	2	2	2	0	0	0	3	2	2	0	0	0	21	5,12	faixa azul
Pong.aia	1	2	3	3	0	2	0	0	0	3	3	2	2	0	0	0	21	5,12	faixa azul
FlappyBird.aia	1	3	3	3	3	3	2	2	1	3	3	2	2	1	0	0	32	7,80	faixa verde
Média	2,10	1,70	3,00	1,40	0,50	1,80	1,30	1,30	0,70	0,90	2,70	2,00	1,80	0,60	0,00	0,00	21,80	5,32	

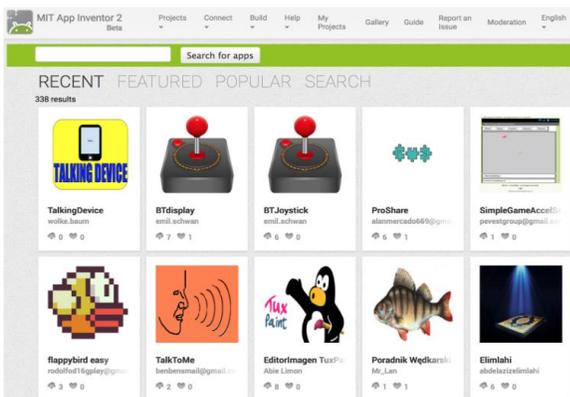
<https://www.youtube.com/watch?v=vCQwJtP2FFU>

CodeMaster 2.0 permite avaliar programas criados com App Inventor de forma **válida e confiável?**

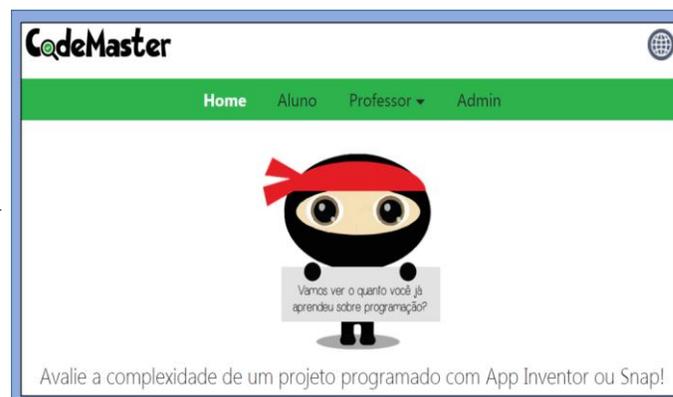
- **Dados:** 88.864 projetos de aplicativos reais da Galeria AppInventor
- **Itens:** avaliação de 16 critérios pelo CodeMaster v2.0
- **Pontuação:** escala ordinal 0 a 3, 0 a 2 e itens dicotômicos.

OBS: consideração de obrigatoriedade para todos os itens

Extração de programas



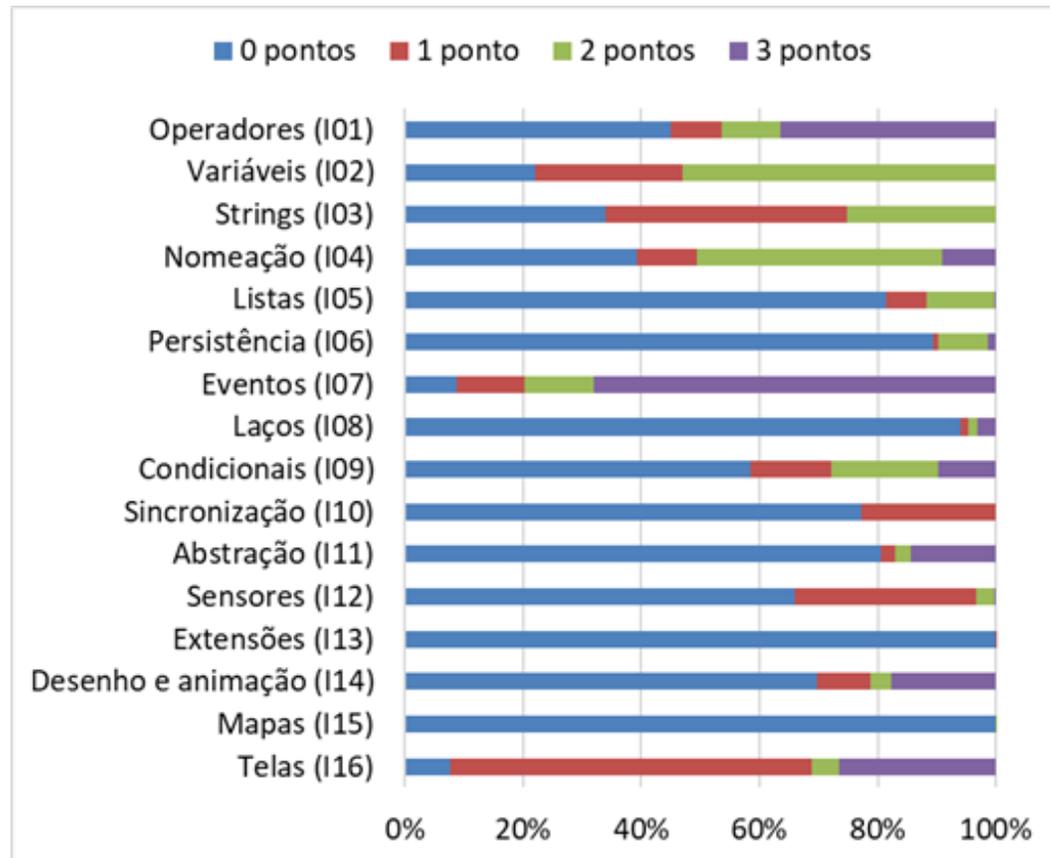
Análise automatizada



Conjunto de dados de projetos avaliados

Aplicativo	I01	I02	...	I16
1	3	2	...	0
2	1	0	...	2
...				
88.812	0	1	2	0

Conjunto de dados de projetos avaliados



- Dos 88.864 projetos, 88.812 foram analisados com sucesso
- Item 13 (Extensões) excluído da avaliação
- Item 15 (Mapas) possui apenas 72 programas com pontuação acima de 0
- Demais itens tem pontuação esperada

Resultados

Confiabilidade

- Análise da consistência interna: Alfa de Cronbach ($\alpha=0,84$)

Validade

- Apenas o item Mapas apresenta baixa correlação item-total

* Item 13 (Extensões) excluído da análise

Item	Correlação Item-total	Alfa de Cronbach se o item for removido
Operadores (I01)	0,694	0,82
Variáveis (I02)	0,686	0,82
Strings (I03)	0,583	0,83
Nomeação (I04)	0,585	0,82
Listas (I05)	0,364	0,84
Persistência (I06)	0,325	0,84
Eventos (I07)	0,596	0,82
Laços (I08)	0,286	0,84
Conditionais (I09)	0,618	0,82
Sincronização (I10)	0,562	0,83
Abstração (I11)	0,548	0,83
Sensores (I12)	0,448	0,84
Desenho e Animação (I14)	0,376	0,84
Mapas (I15)	0,015	0,85
Telas (I16)	0,324	0,84

Conclusões

Modelo CodeMaster

- Fornece suporte **automatizado** para avaliação de programas criados por aluno no contexto da Educação Básica como resultado de atividades abertas
- **Alinhado** ao guia de currículo CSTA e à BNCC
- Resultados da avaliação mostram evidências que o Modelo CodeMaster v2.0 é **válido e confiável**
- Pode-se criar uma escala com interpretação pedagógica para o contexto da Educação Básica usando o CodeMaster



CodeMaster

Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual

Obrigada!

Nathalia da Cruz Alves

nathalia.alves@posgrad.ufsc.br



www.computacaonaescola.ufsc.br

Referências

CSTA. K-12 Computer Science Framework. Technical Report. Computer Science Teachers Association ACM, New York, USA, 2016. Disponível em: <https://k12cs.org/wpcontent/uploads/2016/09/K%E2%80%9312-Computer-Science-Framework.pdf>. Acessado em: 03/07/2019.

MEC. Base Nacional Comum Curricular – BNCC. 2018. Disponível em: http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf. Acessado em: 03/07/2019.

Nathalia da Cruz Alves. CodeMaster: Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual. 2019. Dissertação (Programa de Pós-Graduação em Ciência da Computação (PPGCC)) – Universidade Federal de Santa Catarina.