

**XV SIMPÓSIO DOS PROGRAMAS DE MESTRADO PROFISSIONAL
UNIDADE DE PÓS-GRADUAÇÃO, EXTENSÃO E PESQUISA**



*Desafios de uma sociedade
digital nos Sistemas Produtivos
e na Educação*



**Ranqueamento de Linguagens Para Introdução de Programação
no Curso Técnico de Informática do IFFluminense**

Orpheu de Souza Ayres¹, Milton Erthal Júnior²

Resumo - Este artigo tem por objetivo o ranqueamento de linguagens de programação ou tecnologias de ensino de algoritmo para alunos iniciantes em programação. Este trabalho utilizou dados colhidos a partir de formulários online, utilizando-se a metodologia AHP FUZZY TOPSIS. O método de tomada de decisão multicritério AHP foi aplicado para obter os pesos sobre os critérios. Estes pesos foram utilizados no método FUZZY TOPSIS para a geração do *ranking*. O ranqueamento obtido sobre as linguagens de programação têm destaque para: portugol-webstudio.cubos.io, scratch.mit.edu e studio.code.org. Os resultados obtidos contribuíram para mudanças no planejamento curricular das turmas que iniciaram os estudos de programação no curso técnico de informática do IFFluminense.

Palavras-chave: Lógica Fuzzy, TOPSIS, AHP FUZZY TOPSIS, Auxílio Multicritério a Decisão, Aprendizagem na Educação sobre Algoritmos de Computadores.

Abstract - This article aims to rank programming languages or algorithm teaching technologies for students who are new to programming. From data collection through online forms, using the AHP FUZZY TOPSIS methodology. The AHP multicriteria decision-making method was applied to obtain the weights on the criteria. These weights were used in the FUZZY TOPSIS method to generate the ranking. The ranking obtained on the programming languages is highlighted by: portugol-webstudio.cubos.io, scratch.mit.edu and studio.code.org. The results obtained contributed to changes in the curricular planning of the classes that began programming studies in the IFFluminense IT technical course.

Keywords: Fuzzy Logic, TOPSIS, AHP FUZZY TOPSIS, Multicriteria Decision Support, Learning in Education on Computer Algorithms.

1 - Instituto Federal Fluminense, Campus Quissamã – orpheu.ayres@iff.edu.br
2 - Instituto Federal Fluminense, Campus Guarus - miltonerthal@hotmail.com

1. Introdução

Observado os índices de evasão e reprovação dos Institutos Federais entre 2014 e 2017, conforme dados do INEP (Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira), temos uma média de 20,83%, ou seja, mais de um quinto dos alunos evadem ou reprovam, sendo estes valores considerados uma média alta. A média do ensino médio no primeiro ano para outras instituições de ensino neste mesmo período foi de 18,35% (MEC/Inep/Deed, 2020).

Escolhido o IFFluminense de Quissamã para um estudo de caso, encontramos uma média ainda maior, com 62,91% neste mesmo período para o curso técnico de informática. Conforme mineração de dados sobre os dados do Sistema Acadêmico, a disciplina de algoritmo apareceu em primeiro lugar como a maior responsável pelas reprovações.

A busca de melhores resultados no ensino de programação de computadores constitui objeto de pesquisa de muitas instituições de ensino, algumas colocando o foco em fatores como: método, processo cognitivo, habilidades preexistentes (JENKINS, 2002).

Um curso introdutório de programação aprendido com sucesso constitui a base para o estudo eficaz dos cursos subsequentes e o domínio desta especialidade. Por outro lado, a excessiva complexidade de aprender um curso introdutório pode levar a uma queda na motivação e a baixos resultados de aprendizado em geral. A escolha de linguagem para este fim tem causado intensos debates nas últimas décadas (PROKOP et al, 2019).

Este artigo tem por objetivo utilizar um método de análise multicriterial capaz de eleger e ranquear linguagens de programação, sobre um conjunto de avaliações de docentes com possíveis características de grande dispersão das avaliações, a fim de, com este ranque sugerir linguagens de programação que possam facilitar o processo de ensino tanto para o discente quanto para o docente e conseqüentemente, reduzir a evasão e retenção no curso de formação de tecnólogo em informática ofertado pelo Instituto Federal Fluminense.

A hipótese é que a retenção e evasão pode ser reduzida com uma melhor escolha da linguagem de programação, a fim de reduzir o esforço de aprendizagem com algoritmos e dinamizar o conteúdo.

2 – Metodologia

2.1 – Desenvolvimento de Formulários

Formulários online foram desenvolvidos e aplicados utilizando lógica fuzzy, ou seja, avaliação qualitativa por expressões linguísticas, para avaliar as alternativas de acordo com os critérios.

O primeiro formulário foi aplicado somente para o núcleo de informática do IFFluminense de Quissamã, com contribuição de três docentes, a fim de levantar e definir as linguagens e critérios da pesquisa. Utilizou-se a metodologia AHP, com média aritmética das três avaliações, para dar consistência e gerar os pesos de cada critério a ser utilizado na pesquisa.

O segundo formulário para a avaliação de cada alternativa, de acordo com cada critério, foi disponibilizado aos docentes da área de informática do IFFluminense em seus diversos campi, tendo 21 respostas, submetidas à metodologia AHP FUZZY TOPSIS.

2.2 – Alternativas

As alternativas ficaram limitadas ao escopo de conhecimento e experiência dos docentes do IFFluminense e das linguagens de programação já utilizadas e não a demanda de mercado.

Com base no instrumento de coleta de dados, foram listadas as seguintes alternativas: Java, JavaScript, PHP, C, C#, C++, Python, Objective-C, Ruby, HTML com CSS, Typescript, VB.Net, R, Perl, Assembly, portugol-webstudio.cubos.io, studio.code.org, scratch.mit.edu em um total de dezoito linguagens/tecnologias de ensino de programação. Destas, foram consideradas oito: C; studio.code.org; Java; scratch.mit.edu; JavaScript; portugol-webstudio.cubos.io; Python e C#.

A escolha dessas alternativas levou em consideração a necessidade de promover o fundamento sobre algoritmos e programação estruturada, descartando a necessidade de considerar conceitos e conhecimentos necessários para os anos seguintes de formação, como: banco de dados, rede e programação orientada a objetos.

2.3 – Critérios

Essencialmente, o conhecimento sobre introdução à programação e algoritmos constituem a meta a ser alcançada no primeiro ano de programação do curso técnico de informática do IFFluminense, assim, os critérios utilizados procuraram reduzir redundâncias ou superposições dos critérios, bem como, facilitar a avaliação sobre os pontos mais importantes a serem julgados sobre as linguagens de programação.

Foram cinco os critérios selecionados: legibilidade; depuração de erro; curva de aprendizagem alta; ferramentas de escrita de código e aplicabilidade em projetos reais e baseados nos critérios de Parker e Chao (2006) e Prokop et al (2019).

A legibilidade diz respeito à facilidade de escrita e leitura do código, bem como, à de visualização e entendimento do algoritmo. A depuração de erro faz referência à facilidade sobre a percepção e depuração dos erros, fator determinante para um melhor desempenho, podendo inclusive ser utilizado propositalmente com fins didáticos. A curva de aprendizagem deve refletir a qualidade e velocidade de evolução dos alunos sobre o entendimento e capacidade de construção de algoritmos. As ferramentas de escrita de código, essenciais para a sua existência e devem considerar a disponibilidade, a usabilidade, a interface amigável e os recursos de auxílio na escrita, e, aplicabilidade em projetos reais, cujo critério, se encontra entre o desejável e o ideal, mas, não essencial, pois, a possibilidade de produzir aplicativos utilizáveis na prática, faz o aprendizado mais motivante e potencialmente mais significativo para a formação do programador.

2.4 – Método AHP para os Pesos dos Critérios

Um vetor de pesos feito utilizando a metodologia AHP foi gerado para cada um dos três decisores do núcleo de informática do IFFluminense de Quissamã. Os vetores foram reduzidos a um vetor pela média aritmética. Cada posição do vetor corresponde a um critério e serviu para a geração do peso na aplicação pelo método FUZZY TOPSIS. Definiu-se o valor *less* (abaixo) como o menor julgamento dentre os três e ao valor *up* (acima) o maior (Quadro 1). A combinação destes vetores com a metodologia FUZZY TOPSIS resulta na metodologia AHP FUZZY TOPSIS, mais adequada para tratar um grande volume de avaliações sobre as alternativas.

Quadro 1: Matriz de geração de pesos

	D1	D2	D3	Média	l	f	u
C1	r ₁₁	r ₁₂	r ₁₃	$(r_{11}+r_{12}+r_{13})/n$	Min(r ₁₁ ; r ₁₂ ; r ₁₃)	Média	Max(r ₁₁ ; r ₁₂ ; r ₁₃)
C2	r ₂₁	r ₂₂	r ₂₃	$(r_{21}+r_{22}+r_{23})/n$	Min(r ₂₁ ; r ₂₂ ; r ₂₃)	Média	Max(r ₂₁ ; r ₂₂ ; r ₂₃)
C3	r ₃₁	r ₃₂	r ₃₃	$(r_{31}+r_{32}+r_{33})/n$	Min(r ₃₁ ; r ₁₂ ; r ₁₃)	Média	Max(r ₃₁ ; r ₃₂ ; r ₃₃)

Fonte: próprio autor

l	=	menor valor avaliado	
f	=	média entre os valores correspondentes dos avaliadores	
u	=	maior valor avaliado	
i	=	linha/critério	
j	=	coluna/decisor	
n	=	número de decisores	
C _i	=	critério	
D _i	=	decisor	
r _{ij}	=	resultado do critério C _i do Decisor D _i	
M	=	$(r_{ij}+r_{ij+1}+r_{ij+2})/n$ (Média)	Eq. (1)

Para a geração dos valores de peso com fuzzy, optou-se pela média aritmética sobre as respostas dos decisores, visto que, esta teve maior abrangência que o desvio padrão, tendo, neste caso específico, maior relevância para a aplicação da metodologia AHP FUZZY TOPSIS.

2.5 Método AHP FUZZY TOPSIS com Múltiplos Decisores

Os resultados do formulário online, aplicados a partir do Google *Forms*, foram convertidos automaticamente para uma planilha. As respostas em formato de expressões linguísticas foram convertidas para números. Dois exemplos de como este procedimento foi realizado, podem ser observados no Quadro 2.

Quadro 2: Valores de conversão de literal para crisp

Critério: Legibilidade		Critério: Ferramentas e Recursos de Escrita de Código	
Valor literal	Escala	Valor literal	Escala
Péssima Legibilidade	1	Escrita muito complexa e/ ou muito difícil	1
Difícil Legibilidade	2	Escrita complexa e/ ou difícil	2
Média Legibilidade	3	Escrita Regular	3
Fácil Legibilidade	4	Escrita simples e/ ou fácil	4
Muito fácil Legibilidade	5	Escrita muito simples e/ ou muito fácil	5

Fonte: próprio autor

Cada termo linguístico foi substituído por um número difuso triangular, de acordo com a Tabela 1, em seguida fez-se classificação e agregação pela média aritmética ponderada dos termos; uma Matriz de decisão de valor normalizada foi gerada para obter valores dentro do intervalo [0, 1]; os pesos gerados pelo método AHP foram utilizados para a metodologia FUZZY TOPSIS com seus respectivos valores abaixo e acima; na etapa de soluções ideais baseou-se em Yuen (2014) com definição de um valor máximo na matriz de decisão (para positivo) e valor mínimo (para negativo) e não (PIS)= (1; 1; 1) e (NIS)= (0; 0; 0); a distância da solução ideal para cada alternativa foi calculada usando-se o método de vértice, com base na distância euclidiana. Duas matrizes foram geradas, uma com valores das distâncias em relação a solução ideal positiva e/ outra com em relação a solução ideal negativa; Para realizar a classificação das alternativas, considerando o maior valor. O coeficiente relativo de proximidade de cada alternativa foi calculado a partir das matrizes de distância das soluções positivas e negativas e por fim, fez-se a classificação das alternativas, onde, quanto maior o valor relativo de proximidade, melhor.

Tabela 1: Tabela com valores para fuzzificação

Valores para fuzzificação														
l	f	u	l	f	u	l	f	u	l	f	u	l	f	u
0,0	1,0	3,0	1,0	3,0	5,0	3,0	5,0	7,5	5,0	7,5	9,0	7,5	9,0	10,0
	1			2			3			4			5	

Fonte: próprio autor

Para a construção da matriz de decisão difusa normalizada, leva-se em consideração o tipo do critério, seja ele de custo ou benefício, a fim de que os dados tenham a devida proporcionalidade. O cálculo utilizou as fórmulas:

$$r_{ij} = \left(\frac{a_{ij}}{c_j}, \frac{b_{ij}}{c_j}, \frac{c_{ij}}{c_j} \right); c_j^+ = \max_i c_{ij} (\text{Tipo benefício}) \quad \text{Eq.(3)} \quad r_{ij} = \left(\frac{\bar{a}_{ij}}{c_j}, \frac{\bar{a}_{ij}}{b_j}, \frac{\bar{a}_{ij}}{a_j} \right); a_j^+ = \min_i a_{ij} (\text{Tipo custo}) \quad \text{Eq.(4)}$$

Após a normalização, os pesos são atribuídos sobre cada avaliação, colocando-os na proporção de sua importância. Foram utilizados os pesos calculados no método AHP, gerando uma matriz pesificada, a partir da fórmula: $v_{ij} = r_{ij} * w_{ij}$ Eq. (5).

No método FUZZY TOPSIS, baseado na distância entre pontos, cada valor é percebido como um ponto cartesiano. Necessita encontrar as soluções ótimas,

positivas e negativas, para calcular as distâncias entre os pontos e estas soluções. O cálculo da solução ideal positiva (PIS) é feito a partir da fórmula: $PIS A^+ = [v_{11}, v_{12}, v_{13}, \dots, v_{1n}]$ onde $v_j = \max_i \{v_{ij}\}$ e o da solução ideal negativa (NIS) da fórmula: $NIS A^- = (v_{11}, v_{12}, v_{13}, \dots, v_{1n})$ onde $v_j = \min_i \{v_{ij}\}$.

São geradas duas novas matrizes, uma das distâncias dos pontos para a (PIS) A^+ , e/ outra das distâncias da (NIS) A^- . Os valores fuzzy desaparecem, gerando um valor que representa a distância do ponto relativo a alternativa sobre o critério. A fórmula utilizada é a mesma:

$$\begin{aligned} &\text{Dados dois números fuzzy triangulares} && \text{Eq. (6)} \\ &m = (m_1, m_2, m_3) \text{ e } n = (n_1, n_2, n_3) \\ &d_{(m,n)} = \sqrt{\frac{1}{3} * [(m_1 - n_1)^2 + (m_2 - n_2)^2 + (m_3 - n_3)^2]} \end{aligned}$$

Para calcular as distâncias das alternativas, necessita-se totalizar os distâncias de cada ponto das matrizes de distância d^+ e d^- .

$$S^+ = \sum_{i=1}^i d_i^+ \quad \text{e} \quad S^- = \sum_{i=1}^i d_i^-$$

De posse do total das distâncias, calcula-se o coeficiente de proximidade CC, que permite enxergar a classificação e ranqueamento das alternativas, para tanto, usa-se a fórmula:

$$CC = \frac{S^-}{S^- - S^+}$$

Os valores resultantes são ordenados e geram a classificação de maior para menor no grau de importância, assim, quanto maior o valor menor a distância da solução ideal positiva e maior a distância da solução ideal negativa.

3 – Resultados

A delimitação dos critérios diferenciando as necessidades específicas para a formação do primeiro ano no curso técnico de informática, em detrimento dos anos seguintes, foi essencial, uma vez que o foco se manteve sobre a construção da base do conhecimento sobre programação estruturada, pensamento computacional e algoritmos, independentemente das linguagens de produtividade do mercado. Valorizou-se a facilidade, simplicidade e solidez na formação inicial.

Na primeira etapa de definição dos critérios e pesos com o método AHP, participaram três professores do núcleo do IFFluminense de Quissamã. Na segunda etapa, participaram 21 professores de informática do IFFluminense, no julgamento das alternativas à luz de cada critério.

Como resultado deste trabalho, as linguagens/tecnologias foram ranqueadas de forma a priorizar as mais assimiláveis pelos alunos, ou seja, 1º portugol-webstudio.cubos.io; 2º studio.code.org; 3º scratch.mit.edu; 4º JavaScript; 5º Python; 6º C#; 7º Java e 8º Linguagem C.

O plano de aula da disciplina de introdução a programação foi profundamente reestruturado e colocado em prática no IFFluminense de Quissamã, utilizando não apenas a primeira opção do ranque, mas, paralelamente as três primeiras linguagens do ranque no que cada uma poderia oferecer de melhor.

3.1 – Dos Critérios e Pesos

O quadro 3, traz as avaliações resultantes dos critérios sob a ótica dos três decisores, segundo a metodologia AHP. Após confirmada a consistência dos julgamentos, calculou-se a partir dos vetores de pesos de cada matriz, um único vetor de pesos para cada critério.

Os pesos foram compostos, segundo a lógica fuzzy de um valor abaixo (l), sendo este o menor valor entre os pesos do critério correspondente, um valor de referência (peso), calculado pela média aritmética dos valores correspondentes e um valor acima (u), o maior valor dentre as avaliações do critério correspondente.

Quadro 3: Geração de pesos para AHP FUZZY TOPSIS

	Decisor 1	Decisor 2	Decisor 3	Média	l	peso	u	dp	dp/f
C1	1,619	1,513	2,080	1,737	1,513	1,737	2,080	0,246	14,17%
C2	0,978	1,046	0,794	0,939	0,794	0,939	1,046	0,107	11,36%
C3	4,185	4,391	4,174	4,250	4,174	4,250	4,391	0,100	2,34%
C4	0,382	0,372	0,723	0,492	0,372	0,492	0,723	0,163	33,18%
C5	0,329	0,322	0,297	0,316	0,297	0,316	0,329	0,014	4,38%

Fonte: próprio autor

O critério C3 apresentou-se como o de maior relevância e, com os desvios padrões com baixa variação em relação à média, optou-se por considerar os menores valores observados para (l) e os maiores valores para (u), respectivamente; os valores abaixo e os acima de referência para a formação do número *fuzzy*, e, *assim obter* uma maior abrangência sobre o coeficiente de pertinência de cada avaliação.

A matriz contendo a resposta dos 21 decisores, foi reduzida para uma matriz critérios x alternativas (Figura 1).

Figura 1: Matriz reduzida de avaliações em critérios por alternativas

	C1			C2			C3			C4			C5		
	l	f	u	l	f	u	l	f	u	l	f	u	l	f	u
A1	0	5,8	10	0	5,5	10	1	5,7	10	1	6,7	10	1	7,6	10
A2	0	7,4	10	0	7,3	10	3	8,1	10	1	6,9	10	0	5,4	10
A3	0	6,4	10	0	6,9	10	0	5,7	10	0	6,8	10	5	8,6	10
A4	0	7,5	10	0	6,4	10	3	8,4	10	1	7,5	10	0	5,1	10
A5	0	6,1	10	0	5,7	10	3	7,2	10	1	6,6	10	3	8,4	10
A6	0	7,3	10	0	7,3	10	3	8,2	10	1	7,8	10	0	4,3	10
A7	0	6,7	10	0	6,8	10	1	7,1	10	1	7,1	10	5	8,8	10
A8	0	6,7	10	0	6,7	10	1	6,4	10	1	6,5	10	1	8,4	10

A matriz reduzida foi normalizada, de acordo com a metodologia AHP FUZZY TOPSIS resultando na matriz normalizada (Figura 2) para gerar intervalos de 0 a 1.

Figura 2: Matriz normalizada

	C1	C2	C3	C4	C5										
A1	0,00	0,58	1,00	0,00	0,55	1,00	0,10	0,57	0,00	0,10	0,67	1,00	0,10	0,76	1,00
A2	0,00	0,74	1,00	0,00	0,73	1,00	0,30	0,81	1,00	0,10	0,69	1,00	0,00	0,54	1,00
A3	0,00	0,64	1,00	0,00	0,69	1,00	0,00	0,57	1,00	0,00	0,68	1,00	0,50	0,86	1,00
A4	0,00	0,75	1,00	0,00	0,64	1,00	0,30	0,84	1,00	0,10	0,75	1,00	0,00	0,51	1,00
A5	0,00	0,61	1,00	0,00	0,57	1,00	0,30	0,72	1,00	0,10	0,66	1,00	0,30	0,84	1,00
A6	0,00	0,73	1,00	0,00	0,73	1,00	0,30	0,82	1,00	0,10	0,78	1,00	0,00	0,43	1,00
A7	0,00	0,67	1,00	0,00	0,68	1,00	0,10	0,71	1,00	0,10	0,71	1,00	0,50	0,88	1,00
A8	0,00	0,67	1,00	0,00	0,67	1,00	0,10	0,64	1,00	0,10	0,65	1,00	0,10	0,84	1,00

Uma vez normalizada, foi calculada a importância de cada avaliação proporcionalmente aos pesos de cada critério, conforme figuras 62 e 63.

Figura 3: Matriz de pesos consistentes (AHP)

pesos	1,51	1,74	2,08	0,79	0,94	1,05	4,17	4,25	4,39	0,37	0,49	0,72	0,30	0,32	0,33
-------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------

Figura 4: Matriz pesificada

	C1	C2	C3	C4	C5										
A1	0	1,01	2,08	0	0,51	1,05	0,42	2,44	0	0,04	0,33	0,72	0,03	0,24	0,33
A2	0	1,29	2,08	0	0,69	1,05	1,25	3,44	4,39	0,72	0,34	0,72	0	0,17	0,33
A3	0	1,11	2,08	0	0,65	1,05	0	2,43	4,39	0	0,34	0,72	0,15	0,27	0,33
A4	0	1,29	2,08	0	0,60	1,05	1,25	3,56	4,39	0,04	0,37	0,72	0	0,16	0,33
A5	0	1,06	2,08	0	0,54	1,05	1,25	3,08	4,39	0,04	0,32	0,72	0,09	0,26	0,33
A6	0	1,26	2,08	0	0,69	1,05	1,25	3,50	4,39	0,04	0,39	0,72	0	0,14	0,33
A7	0	1,17	2,08	0	0,64	1,05	0,42	3,02	4,39	0,04	0,35	0,72	0,15	0,28	0,33
A8	0	1,17	2,08	0	0,63	1,05	0,42	2,72	4,39	0,04	0,32	0,72	0,03	0,26	0,33

5. Soluções ótimas positivas (PIS) e negativas (NIS) foram encontradas, figura

Figura 5: Matriz com soluções ótimas positivas (PIS) e negativas (NIS)

A ⁺ =	[0	1,29	2,08	0,00	0,69	1,05	1,25	3,56	4,39	0,04	0,39	0,72	0,15	0,28	0,33]
A ⁼	[0	1,01	2,08	0,00	0,51	1,05	0,00	2,43	0,00	0,00	0,32	0,72	0,00	0,14	0,33]

O método AHP FUZZY TOPSIS baseia-se na ideia de visualizar as pontuações como pontos cartesianos. Criou-se duas Matrizes, uma com as distâncias positivas (d^+), contendo as distâncias de cada ponto em relação a A^+ na Figura 65 e/ outra com as distâncias negativas (d^-), contendo as distâncias de cada ponto em relação a A^- na Figura 6.

Figura 6: Matriz com as distâncias dos pontos para PIS e para NIS

	C1	C2	C3	C4	C5		C1	C2	C3	C4	C5	
d^+	A1	0,16	0,10	2,66	0,03	0,07	A1	0,00	0,00	0,24	0,02	0,06
	A2	0,00	0,00	0,07	0,03	0,11	A2	0,16	0,10	2,70	0,02	0,02
	A3	0,11	0,02	0,98	0,04	0,00	A3	0,06	0,08	2,54	0,01	0,12
	A4	0,00	0,05	0,00	0,01	0,11	d^- A4	0,16	0,05	2,72	0,04	0,01
	A5	0,14	0,09	0,28	0,04	0,04	A5	0,03	0,01	2,66	0,02	0,09
	A6	0,02	0,00	0,04	0,00	0,12	A6	0,15	0,10	2,71	0,04	0,00
	A7	0,07	0,03	0,58	0,02	0,00	A7	0,09	0,07	2,57	0,03	0,12
	A8	0,07	0,04	0,68	0,04	0,07	A8	0,09	0,06	2,55	0,02	0,08

As distâncias em cada critério foram somadas para gerar uma única distância da alternativa, tanto para as distâncias positivas em S^+ , quanto para as distâncias negativas em S^- . Calculou-se as distâncias que permitiram encontrar o coeficiente de proximidade de cada alternativa. Este coeficiente foi calculado dividindo-se S^- pelo somatório de S^- com S^+ na Figura 7.

O último procedimento consiste na ordenação dos coeficientes de proximidade, apontando em ordem crescente, do melhor para o pior, ou o primeiro lugar para o de maior pontuação. O ranque resultante está demonstrado na figura 7.

Figura 7: Coeficiente de proximidade e Ordenação de resultados

	S-	S+	S- / (S- + S+)	Ordenação
Linguagem C	0,33	3,03	0,10	8
studio.code.org	3,00	0,21	0,93	3
Java	2,80	1,14	0,71	7
scratch.mit.edu	2,98	0,17	0,95	2
JavaScript	2,82	0,57	0,83	4
portugol- webstudio.cubos.io	3,00	0,17	0,95	1
Python	2,88	0,70	0,80	5
C#	2,80	0,90	0,76	6

4 – Conclusão

O ranque das linguagens foi realizado com dados de qualidade, utilizando metodologias adequadas em todas as etapas, com expressões linguísticas no lugar de valores numéricos para as avaliações nos formulários e com lógica *fuzzy* aplicada.

A metodologia AHP foi essencial para a consistência na geração do vetor de pesos, assim como a utilização de variáveis linguísticas permitiu uma avaliação mais fidedigna no tocante à percepção dos técnicos avaliadores sobre cada alternativa, e, resultando em maior assertividade acerca dos dados fuzzy/nebulosos na aplicação da metodologia AHP FUZZY TOPSIS.

Considera-se ainda que houve boa participação dos professores da área, com 21 respostas, entendendo-se assim, com uma quantidade relevante e necessária para os cálculos realizados.

Referências

Aguado, A.G.; Cantanhede, M.A. **Lógica Fuzzy**. (2010). Disponível em: <<http://www.ft.unicamp.br>>, Acesso em: 12 NOV. 2019.

Chen, C.T. **Extensions of the TOPSIS for group decision-making under fuzzy environment**. *Fuzzy Sets and Systems*, 114, 1–9 (2000).

COSTA, H.G. **Auxílio multicritério à decisão: método AHP**. Rio de Janeiro: ABEPRO, 2006.

Jenkins, T - **On the difficulty of learning to program - ”**, in Proc. of the 3rd Annu. LTSN_ICS Conf., Loughborough University, United Kingdom, pp. 53-58. (2002)

Madi E.N., Garibaldi J.M., Wagner C., **A comparison between two types of FUZZY TOPSIS method**. *Proceedings of the 2015 IEEE International Conference on Systems, Man, and Cybernetics (SMC'15)*, , Hong Kong, China. Piscataway, NJ, USA: IEEE, 2015: 291–297, (2015).

Ministério da Educação. (MEC) - Pesquisa sobre Instituições da Rede Federal. Disponível em: <<http://portal.mec.gov.br/rede-federal-inicial/instituicoes>> Acessado em 25/06/2020.

Parker K.R., Chao J.T., Ottaway T.A., Chang J., **A Formal Language Selection Process for Introductory Programming Courses**, *Journal of Information Technology Education*, vol. 5, (2006).

Prokop, Y., Trofimenko, Severin, N., E., Bukata, N. **An Analysis of Criteria for Choosing a First Programming Language in Universities** - Archived Volume, – researchgate.net DOI: 10.22152/programming-journal. Org/2017/1/18. 16. (2019).