



UNIVERSIDADE ESTADUAL DA PARAÍBA - UEPB
CAMPUS VII – GOVERNADOR ANTÔNIO MARIZ
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS
CURSO DE LICENCIATURA PLENA EM COMPUTAÇÃO

JAQUELINE RODRIGUES CARNEIRO

TER APTIDÃO EM MATEMÁTICA IMPLICA SER BOM EM PROGRAMAÇÃO?
UM ESTUDO DE CASO EM UMA INSTITUIÇÃO PRIVADA DE PATOS-PB

PATOS
2017

JAQUELINE RODRIGUES CARNEIRO

**TER APTIDÃO EM MATEMÁTICA IMPLICA SER BOM EM PROGRAMAÇÃO?
UM ESTUDO DE CASO EM UMA INSTITUIÇÃO PRIVADA DE PATOS-PB**

Artigo apresentado ao Curso de Licenciatura em Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de Licenciado em Computação.

Orientador: Prof. MSc. Pablo Ribeiro Suárez

**PATOS
2017**

C289t Carneiro, Jaqueline Rodrigues.

Ter aptidão em Matemática implica ser bom em Programação? [manuscrito] : um estudo de caso em uma instituição privada de Patos - PB / Jaqueline Rodrigues Carneiro. - 2017

21 p. : il. colorido.

Digitado.

Trabalho de Conclusão de Curso (Graduação em Computação) - Universidade Estadual da Paraíba, Centro de Ciências Exatas e Sociais Aplicadas, 2017.

"Orientação : Prof. Me. Pablo Ribeiro Suárez, Coordenação do Curso de Computação - CCEA."

1. Ensino de Programação. 2. Dificuldades em Programação. 3. Matemática em Programação.

21. ed. CDD 005.711

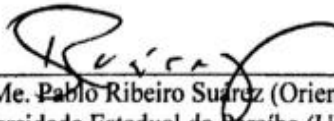
JAQUELINE RODRIGUES CARNEIRO

TER APTIDÃO EM MATEMÁTICA IMPLICA SER BOM EM PROGRAMAÇÃO? UM
ESTUDO DE CASO EM UMA INSTITUIÇÃO PRIVADA DE PATOS-PB

Artigo apresentado ao Curso de Licenciatura em Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de Licenciado em Computação.

Aprovada em: 05/12/2017.

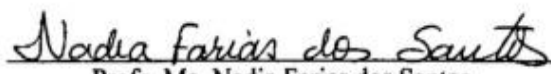
BANCA EXAMINADORA



Prof. Me. Pablo Ribeiro Suárez (Orientador)
Universidade Estadual da Paraíba (UEPB)



Prof. Pablo Roberto Fernandes de Oliveira
Universidade Estadual da Paraíba (UEPB)



Profa. Ma. Nadia Farias dos Santos
Universidade Estadual da Paraíba (UEPB)

Aos meus pais, pela dedicação, companheirismo e amizade, DEDICO.

AGRADECIMENTOS

Agradeço primeiramente a Deus, que me guiou durante todo o caminho, me dando força e sabedoria para chegar até aqui. A ti, meu pai, toda honra e glória.

A minha mãe Janete, que nunca mediu esforços para realizar meus sonhos e sempre foi minha maior incentivadora.

Ao meu pai, José Carneiro (*in memoriam*) por ter sido meu companheiro durante todo meu período acadêmico, dando-me força e apoio. Embora fisicamente ausente, sei que espiritualmente se faz presente e, onde estiver, estará feliz e realizado por mais essa conquista em minha vida.

A minha tia Janeide e a minha avó Maria das Dores (*in memoriam*), que tiveram papel fundamental para essa conquista.

A minha irmã Juliete, que não titubeou em me ajudar sempre que precisei.

Ao meu esposo Luiz Augusto, que foi meu grande mestre, professor e incentivador. Por todo amor, compreensão e apoio prestado durante toda a minha caminhada.

A minha sogra Terezinha, que me estendeu a mão e me acolheu como filha.

Ao meu professor e Orientador Pablo Suárez, pela paciência, presteza e pelas inúmeras contribuições para a construção deste trabalho.

A todos os professores do Curso de Licenciatura em Computação da UEPB, que contribuíram para o meu desenvolvimento acadêmico.

Aos colegas de classe e companheiros de luta: Amélia, Marcela, Klyvia e Rianne. Em especial a Arthur, Domaria e Joelson, a quem tanto admiro e sou grata pelos momentos de amizade e apoio.

A todos os meus familiares e amigos, que mesmo distante sempre torceram pelo meu sucesso. Em especial ao meu padrasto Severino e os meus amigos Cíntia e Filipe.

Sumário

1 INTRODUÇÃO.....	7
1.1 Definição da problemática	8
1.2 Objetivos.....	9
1.2.1 Objetivo Geral.....	9
1.2.2 Objetivos Específicos.....	9
1.3 Justificativa	9
1.4 Estrutura do trabalho.....	10
2 REFERENCIAL TEÓRICO.....	11
2.1 Ensino introdutório de programação.....	11
2.2 Dificuldades dos alunos em aprender a programar.....	12
2.3 Relação entre programação e matemática	13
3 METODOLOGIA.....	14
3.1 Tipo de Pesquisa	14
3.2 Sujeitos.....	15
3.3 Coleta de dados	15
3.4 Análise	15
4 RESULTADOS	15
4.1 Existe relação entre matemática e programação no curso de Sistemas de Informação das FIP?.....	15
4.2 Por que não se encontrou relação entre matemática e programação?.....	18
5 CONCLUSÃO.....	18
REFERÊNCIAS	19

TER APTIDÃO EM MATEMÁTICA IMPLICA SER BOM EM PROGRAMAÇÃO? UM ESTUDO DE CASO EM UMA INSTITUIÇÃO PRIVADA DE PATOS-PB

Jaqueline Rodrigues Carneiro¹

RESUMO

É de conhecimento na literatura que as disciplinas de programação possuem altos índices de reprovação em cursos de Ciência da Computação e afins. Uma possível solução para atenuar o mau desempenho dos estudantes seria identificar com antecedência variáveis que pudessem indicar quais os indivíduos da turma que teriam mais dificuldade na disciplina. Este trabalho explorou a relação entre as notas de matemática do vestibular e o desempenho dos alunos em programação com o intuito de entender se o desempenho em matemática seria um bom preditor neste contexto. Para isso, foi utilizado dados do curso de computação de uma instituição privada de ensino de Patos-PB. Como resultado, descobriu-se que não existe uma correlação significativa entre as duas variáveis a partir dos dados coletados. Portanto, no contexto da instituição explorada, ter aptidão em matemática não implica ser bom em programação.

Palavras-Chave: Ensino de Programação. Dificuldades em Programação. Correlação entre Matemática e Programação.

1 INTRODUÇÃO

Os cursos na área de Computação geralmente são compostos por várias disciplinas que utilizam como ponto central a Programação e comumente essas disciplinas são apresentadas para os alunos desde o início do curso. Nos últimos anos pôde se perceber uma alta taxa de desistência e baixo desempenho por parte dos alunos em disciplinas que exigiam o conhecimento de programação. Por esta razão, vários pesquisadores vêm investigando maneiras para solucionar esses problemas (TAN; TING; LING, 2009).

Os estudos até então realizados podem contribuir para que haja uma identificação prematura de alunos com baixo desempenho, podendo diminuir os problemas citados anteriormente. Pois, ao serem identificados com antecedência, os alunos que possivelmente

¹ Aluna de Graduação em Licenciatura em Computação na Universidade Estadual da Paraíba – Campus VII.

Email: kelly.rodrigues@gmail.com

apresentem fatores que venham interferir no seu rendimento de forma negativa poderão ser assistidos de uma maneira mais efetiva, recebendo um ensino focado em sua individualidade.

Em uma turma iniciante, os professores ainda não conhecem seus alunos e só conseguem identificar o desempenho de cada um após as primeiras avaliações. No entanto, existe um grande espaço de tempo entre o início das aulas e a primeira prova (BERGIN; REILLY, 2005). Esse atraso na identificação dos alunos que estão propensos a reprovação pode prejudicar ou até invalidar a intervenção dos professores a esses alunos, pois eles poderão estar desmotivados ou não conseguirão acompanhar o ritmo das aulas.

Identificar os alunos de risco a partir da experiência em sala de aula ou por meio das avaliações da disciplina pode ser útil em alguns aspectos. Porém, é mais sensato utilizar técnicas estatísticas formais para desenvolver modelos que auxiliem nessa predição (CHAMILLARD, 2006).

Este trabalho tem como finalidade analisar os dados da disciplina de Matemática do processo seletivo para ingresso no ensino superior, somado às notas da disciplina de Programação I dos alunos do curso de Sistemas de Informação, das Faculdades Integradas de Patos, com o intuito de identificar uma possível relação da matemática no desempenho em programação.

Estudos relacionados à predição de desempenho já foram realizados anteriormente. Este trabalho tem por objetivo reproduzir alguns dos resultados encontrados na literatura e apresentar a perspectiva dos dados específicos da instituição.

1.1 Definição da problemática

Ao longo da disciplina de programação, professores experientes conseguem identificar os alunos que estão conseguindo aprender facilmente e os outros que estão com dificuldades. No entanto, essa identificação nem sempre é precoce, o que impossibilita o docente de auxiliar alguns alunos a tempo.

A aprendizagem de programação parece estar ligada a alguns fatores que são fundamentados nas experiências vividas pelos alunos em sua formação básica, como a aptidão em matemática e resolução de problemas (BYRNE; LYONS, 2001). Neste sentido, este trabalho tem como propósito investigar quais são as características intrínsecas da matemática na formação básica dos alunos que podem ter efeitos positivos ou negativos em seu desempenho na disciplina de introdução a programação.

Este trabalho será mais uma evidencia de que a matemática pode ter correlação com a aprendizagem em programação, o que facilitará para a criação de modelos que possam prever quais seriam os alunos que estão ou não propensos a obter um mau desempenho na disciplina e, conseqüentemente, monitorá-los mais cuidadosamente.

1.2 Objetivos

1.2.1 Objetivo Geral

A aprendizagem de programação pode estar vinculada a fatores fundamentados em experiências na formação básica do aluno, entre elas, a matemática. Conhecer esses fatores pode nortear os professores a uma identificação antecipada dos alunos de risco e a partir disso, poderão buscar práticas educativas adequadas que levem à minimização do insucesso em disciplinas iniciais de programação. Este trabalho tem como objetivo investigar se o desempenho em matemática oriundo de avaliações anteriores tem relação com a nota final dos alunos em uma disciplina introdutória de programação do curso Sistemas de Informação das Faculdades Integradas de Patos.

1.2.2 Objetivos Específicos

Decorrente do objetivo geral tem-se como objetivos específicos:

- *Categorizar os estudantes de acordo com seu desempenho*
- *Explorar a relação entre as notas de matemática e programação*
- Discutir sobre os resultados do estudo de caso

1.3 Justificativa

A identificação prematura de alunos com baixo desempenho pode diminuir os problemas citados anteriormente. Ao serem identificados com antecedência, os alunos de risco podem ser assistidos de uma maneira mais efetiva, recebendo um ensino focado em sua individualidade.

Os professores só conseguem identificar o desempenho do aluno a partir das primeiras avaliações. Esse espaço de tempo entre o início das aulas e a primeira avaliação é grande, se pensarmos que o melhor seria possuir um conhecimento “prévio” do histórico do aluno.

O atraso na identificação dos alunos propensos a um baixo desempenho pode invalidar ou prejudicar a intervenção dos professores a esses alunos, pois corre o risco de já

estarem desmotivados ou não conseguirem mais acompanhar o ritmo da disciplina. Identificar os alunos de risco a partir da experiência em sala de aula ou por meio das avaliações da disciplina pode ser útil em alguns aspectos. Porém, é mais sensato utilizar técnicas estatísticas formais para desenvolver modelos que auxiliem nessa predição (CHAMILLARD, 2006).

Este trabalho é eficaz na identificação de preditores que influenciem positivo ou negativamente na aprendizagem dos alunos das disciplinas de linguagens de programação para que seja desenvolvido um modelo estatístico capaz de identificar alunos de risco de uma forma antecipada.

1.4 Estrutura do trabalho

O presente trabalho está dividido em cinco capítulos incluindo este, que apresentou uma breve introdução sobre o tema escolhido, a definição da problemática e os objetivos da pesquisa.

O segundo capítulo é composto pelo referencial teórico, decorrente de uma revisão bibliográfica utilizada como embasamento para a definição do tema e sua relevância para a consolidação da pesquisa realizada. Nele foi mostrado diversos trabalhos teóricos e experimentais realizados por diferentes autores, de modo a compreender a problemática do trabalho em questão. Apresentou também as questões chaves da pesquisa, tal como o ensino introdutório de programação, as dificuldades dos alunos em aprender e a relação entre a matemática e a programação.

No terceiro capítulo foi apresentado a metodologia aplicada. Especificando todo o processo metodológico, o tipo de pesquisa, os sujeitos, a coleta de dados e a análise feita durante a pesquisa.

No quarto capítulo, os resultados e discussões a cerca deste trabalho, dentre eles a avaliação dos dados da instituição e possíveis fatores que velaram o resultado da pesquisa a ir de encontro com o que a literatura apresenta.

Por fim, no quinto capítulo, as conclusões, que de um modo geral apresentou as contribuições da pesquisa, as limitações do estudo, sugestões de trabalhos futuros e considerações finais. Seguido das referências bibliográficas utilizadas.

2 REFERENCIAL TEÓRICO

Nesta seção serão introduzidos conceitos relevantes para o entendimento do trabalho. Dentre os temas que serão abordados, estão a importância do ensino de programação em cursos de computação, a comprovada dificuldade em cursos tanto a nível nacional quanto internacional e, por fim, resultados de trabalhos anteriores que exploraram a relação entre notas prévias em matemática e o desempenho final dos alunos em programação.

2.1 Ensino introdutório de programação

O ensino introdutório de programação existe desde as origens da Ciência da Computação moderna e faz parte do currículo básico de todos os cursos desta área. Essa disciplina tem o propósito de desenvolver nos alunos, além dos conceitos técnicos, a capacidade de compreender e resolver problemas (GOMES; HENRIQUES; MENDES, 2008).

Ambrósio, et al (2011, p. 3) definem programação como uma forma de resolução de problemas e afirmam que a programação pode ser decomposta em quatro fases: “entender o problema, definir ou planejar uma solução, implementar um plano e verificar se está certo ou se é adequado o resultado final atingido. Pensando nestas quatro fases, antecipamos a diversidade de competências e de conhecimentos envolvidos.”

Conforme consta nas Diretrizes Curriculares da área de computação e informática do MEC (1999), a programação é uma atividade voltada à solução de problemas e está relacionada com várias outras atividades, bem como especificação, projeto, validação, modelagem e estruturação de programas e dados, utilizando-se das linguagens de programação como ferramenta. O conteúdo abordado nesta disciplina não se restringe apenas ao ensino de uma linguagem de programação propriamente dita, envolve um conjunto de princípios, técnicas e formalismos a fim de se obter softwares bem estruturados e confiáveis.

Corroborando com essa ideia, Gomes et al (2008, p. 1), afirmam que a programação vai além de um conjunto de linhas de código numa dada linguagem. Para eles, a programação é uma arte e uma ciência. Arte porque possibilita que o usuário use sua criatividade para codificar as instruções, que podem ser feitas de diferentes maneiras. E uma ciência, porque é composta por um conjunto de regras orientadoras; porque há uma necessidade de conhecimento e uso de lógica e, além do mais, possui métodos rigorosos de programar que asseguram a eficiência, economia e utilidade dos programas gerados.

Programação é, sem dúvidas, uma das disciplinas mais importantes para uma boa formação acadêmica nos cursos da área de Computação. Por isso, é necessário que o aluno tenha domínio dos conteúdos abordados, de modo que consiga por em prática suas habilidades, principalmente quando se pretende ter como produto final do trabalho o desenvolvimento de softwares. (PIMENTEL; FRANÇA; NORONHA; OMAR, 2003).

2.2 Dificuldades dos alunos em aprender a programar

A programação é um campo fundamental da Computação e está presente no decorrer de praticamente todos os cursos dessa área. Todavia, em várias instituições de ensino a aprendizagem dos alunos nas disciplinas de programação tem gerado preocupação pelo baixo índice de assimilação nos conteúdos (PIMENTEL; FRANÇA; OMAR, 2003).

Para grande parte dos estudantes o primeiro contato com a programação tem sido uma experiência difícil e desafiadora. Os alunos ingressantes costumam apresentar comportamentos distintos no que diz respeito à aprendizagem na disciplina. Enquanto uns conseguem aprender a programar rapidamente, inclusive com relativa facilidade, outros apresentam enormes dificuldades (LEITHER; LEWIS, 1978 apud AMBRÓSIO et al, 2011).

Alguns autores como Neto e Schuvartz (2007) e Silva et al (2009) consideram que o fato das disciplinas de programação estarem presentes geralmente nos primeiros períodos do curso pode fazer com que o aluno se sinta incapaz de programar devido a um conjunto de habilidades que a programação exige, no qual o aluno ainda não está acostumado por ainda estar em um processo de familiarização do ambiente acadêmico.

Algumas outras dificuldades foram apontadas por pesquisadores como possíveis causas de insucesso dos alunos na disciplina de programação. Dentre elas estão:

- i) A dificuldade dos alunos em compreender, refletir e praticar os conteúdos abordados na disciplina, quando os mesmos estão acostumados com disciplinas em que a memorização e mecanização da aprendizagem são suficientes para garantir sua aprovação (GOMES et al, 2008);
- ii) Falta de compreensão do raciocínio lógico (TOBAR et al. apud SANTOS e COSTA, 2006) , (NETO & SCHUVARTZ, 2007);
- iii) Capacidade para solucionar problemas concretos (GOMES et al, 2008);
- iv) Falta de habilidades matemáticas; (NETO; SCHUVARTZ, 2007)

Diante das dificuldades até então citadas, Costa e Bubliz (2013) fizeram uma análise do ponto central desse problema, que é o fato de alguns alunos não estarem aprendendo programação e quais os fatores estão influenciando para que isso aconteça. Será que o problema está relacionado apenas com o aluno? E o professor? E a instituição? Tendo em vista esses pontos, os autores decidiram analisar separadamente o papel de cada um no processo de ensino-aprendizagem de programação.

Segundo eles, caso o professor seja o responsável pelo problema em questão, um dos fortes indícios é o de que ele pode não conseguir transmitir o conteúdo como deveria, o que pode ser explicado de acordo com as seguintes vertentes: (i) o professor não tem domínio do conteúdo; (ii) o professor não possui uma formação condizente com o cargo que ocupa; (iii) o professor não tem didática em sala de aula; (iv) a instituição não oferece condições necessárias para que o professor realize suas atividades da maneira que deseja. Se pensarmos na última vertente, podemos então tirar a culpa do professor, que encontrou empecilhos impostos pela instituição que acabaram atrapalhando o seu desempenho.

Quando relacionado o problema ao próprio aluno, categorizaram-se algumas hipóteses: (i) O aluno utiliza métodos inadequados de estudo; (ii) Desmotivação na disciplina; (iii) Recursos insuficientes para a aprendizagem fora da sala de aula; (iv) atraso e/ou evasão nas aulas; (v) o aluno não consegue acompanhar o ritmo da disciplina.

Todas as dificuldades citadas na literatura podem ser vistas através do alto índice de evasão e reprovação nas disciplinas introdutórias de programação. Portanto, fica evidente que é importante a identificação dos fatores que têm afetado o rendimento dos alunos nas disciplinas iniciais de programação, a fim de combater esses problemas em sua origem, aumentando assim a possibilidade de reverter esse quadro de reprovações/evasões.

2.3 Relação entre programação e matemática

A resolução de problemas de programação exige do aluno diversas competências, dentre elas, segundo Gomes et al (2008) está o conhecimento matemático. Associada a essa questão, alguns autores como Ambrosio et al (2011) consideram o conhecimento prévio em matemática um indicativo para se obter êxito na disciplina de programação.

Segundo experiências produzidas por Gomes et al (2006) afim de encontrar correlação com o conhecimento matemático e a resolução de problemas de programação, concluiu-se que os alunos que tinham dificuldades em aprender programação apresentavam, em simultâneo, inúmeras dificuldades em diversas áreas, entre elas calculo básico, teoria de

números, conceitos geométricos e trigonométricos simples. Além do mais, os autores frisaram a dificuldades dos alunos em relacionar a descrição textual do problema com a fórmula matemática para resolvê-lo (apud GOMES et al, 2008, p.4).

Acredita-se que a correlação de sucesso entre as disciplinas de programação e matemática esteja ligada com as funções cognitivas mais básicas que são comuns às duas áreas. Assim que o aluno consegue resolver problemas matemáticos, adquire automaticamente habilidades cognitivas similares necessários para a aprendizagem e resolução de problemas de programação (AMBRÓSIO et al, 2001, p. 8).

Pensando nisso, autores como Henderson (1987) defendem a ideia de que a disciplina de algoritmos e linguagens de programação deve ser integrada com a matemática discreta, com a justificativa de que o algoritmo quando ensinado de uma maneira informal, mostrando apenas os rudimentos da lógica booleana faz com que os alunos desenvolvam apenas soluções algorítmicas para problemas simples (apud KOLIVER, 2004, p.3).

Corroborando com essa ideia, Koliver (2004) afirma que a matemática discreta é sim um pré-requisito desejável no processo de ensino-aprendizagem de programação, porém, acredita que não seja absolutamente necessário esse conhecimento prévio no nível introdutório no qual a disciplina de programação está inserida.

De uma forma geral, vimos que conceitos de matemática estão presentes direto ou indiretamente no cotidiano dos alunos de programação. Sendo assim, conclui-se que, de certa forma não pode ser ignorada por quem deseja seguir a carreira acadêmica na área.

3 METODOLOGIA

Neste trabalho, foi realizado um estudo de caso na disciplina de Programação I, das Faculdades Integradas de Patos (FIP), com o objetivo de analisar se havia relação entre a experiência anterior dos alunos em matemática e seu desempenho final em programação.

3.1 Tipo de Pesquisa

O estudo se consistiu em uma pesquisa quantitativa, do tipo ex-post facto, na qual os dados são coletados após a ocorrência dos eventos e o pesquisador não tem controle sobre as variáveis, nem tão pouco como interferir ou manipular os resultados. Segundo Gil (2002, p.8), “o propósito básico desta pesquisa é verificar a existência de relações entre variáveis” e consequentemente obter uma constatação sobre essas relações.

3.2 Sujeitos

Foram analisados os dados de 59 alunos da disciplina de linguagem de programação no curso de Sistemas de Informação nas Faculdades integradas de Patos entre o período de 2015.1 a 2016.1. Foi selecionada uma amostra de 30 dos 59 alunos, tendo em vista que 29 dos 59 alunos estão incluídos em um dos seguintes aspectos: foram reprovados por falta, trancaram a disciplina, são repetentes na disciplina ou entraram no curso através da nota do Exame Nacional do Ensino Médio (ENEM). Deste modo, optou-se por não incluir esses alunos, pois acreditou-se que poderiam interferir nos resultados da pesquisa.

3.3 Coleta de dados

Para a realização deste estudo foi utilizado um conjunto de dados dos alunos do curso de Sistemas de Informação das Faculdades Integradas de Patos, entre o período letivo de 2015.1 a 2016.1. Os dados foram fornecidos pelo Grupo de Apoio à Tecnologia da informação (GATI) e consistem nas notas de matemática (do tipo ordinal) do processo seletivo para ingresso na instituição e as notas da disciplina de programação (do tipo numérico) referentes ao primeiro período do curso.

3.4 Análise

A análise inicial foi descritiva, com o auxílio de dados categorizados em uma tabela e em um gráfico de pontos. Posteriormente, foi feita uma análise inferencial, a partir da correlação entre as variáveis das notas de matemática e de programação. A correlação foi obtida a partir do Ranking de Correlação de Spearman [McDonald, 2009], que deve ser utilizado quando uma das variáveis tem caráter ordinal – como é o caso das notas de matemática, neste contexto – ou quando se tem poucos dados. Ambas as condições foram satisfeitas neste estudo.

4 RESULTADOS

Neste capítulo serão apresentados os resultados e discussões a respeito do trabalho desenvolvido, além das limitações da pesquisa.

4.1 Existe relação entre matemática e programação no curso de Sistemas de Informação das FIP?

O desempenho dos estudantes no t3pico de matem3tica ao prestarem o vestibular foi em m3dia 2 (desvio padr3o de 1), onde a nota variava numa escala de 0 a 4. J3 o desempenho m3dio em programaç3o foi de 7,6 (desvio padr3o de 1,8), num intervalo de 0 a 10. Na Tabela 1, categorizamos as notas de programaç3o em intervalos de 2 em 2 pontos e agrupamos a quantidade de alunos pelo desempenho em matem3tica (vestibular) e programaç3o.

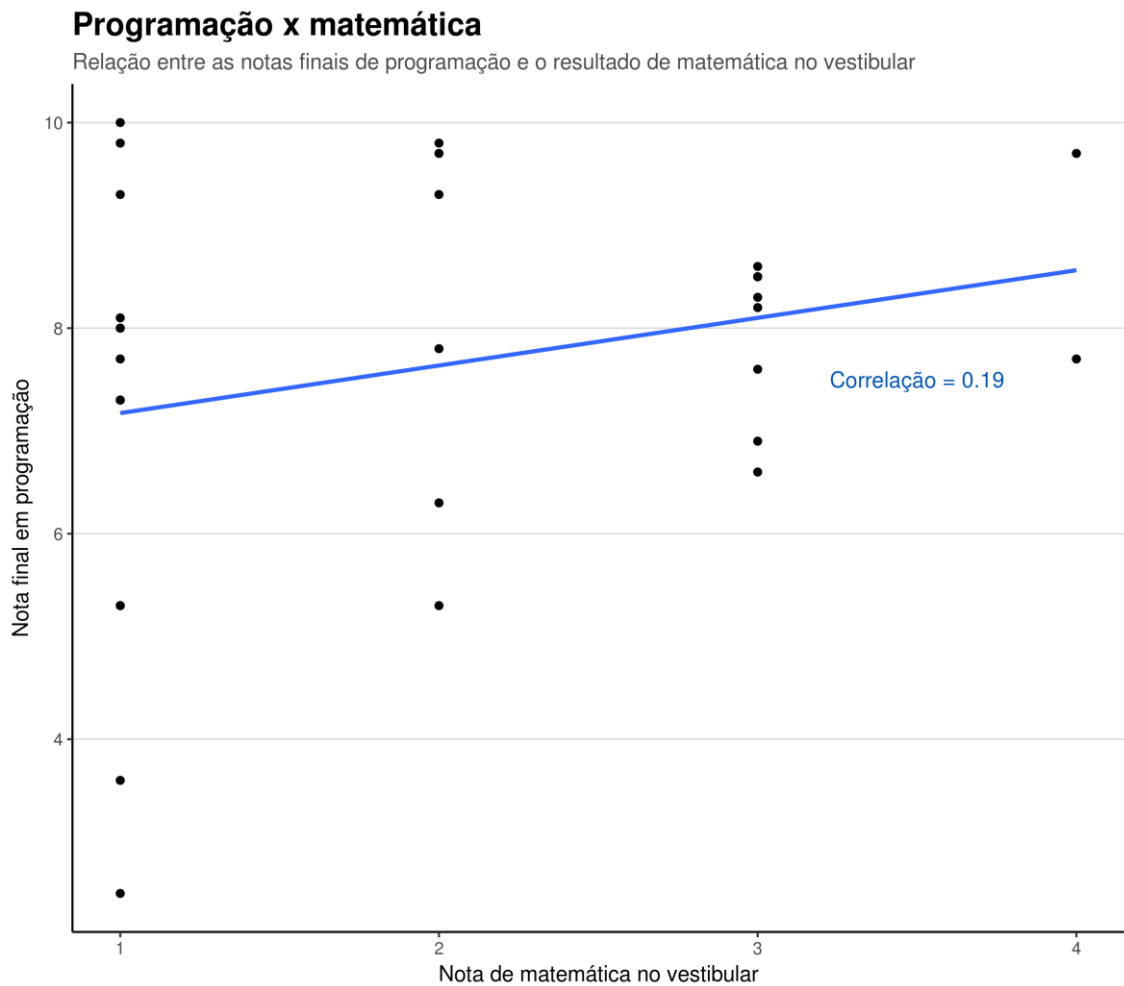
Tabela 1: Quantidades de alunos com relaç3o às notas de matem3tica e programaç3o

Notas Vestibular	Notas Programaç3o	Quantidade de Alunos	Porcentagem
1	(2 a 4]	2	6,9%
1	(4 a 6]	2	6,9%
1	(6 a 8]	6	20,7%
1	(8 a 10]	3	10,3%
2	(4 a 6]	1	3,4%
2	(6 a 8]	2	6,9%
2	(8 a 10]	3	10,3%
3	(6 a 8]	3	10,3%
3	(8 a 10]	5	17,2%
4	(6 a 8]	1	3,4%
4	(8 a 10]	1	3,4%

Fonte: Dados fornecidos pelo Grupo de Apoio à Tecnologia da Informaç3o

À primeira vista, o que mais se destaca é a proporç3o de estudantes que est3o no grupo que obteve nota 1 em matem3tica e entre 6 e 8 em programaç3o. Al3m disso, outro grupo que possui uma quantidade de pessoas ligeiramente maior que os demais é aquele formado por alunos que alcançaram nota 3 no vestibular e entre 8 e 10 na disciplina de programaç3o. Os demais grupos possuem uma distribuiç3o semelhante de alunos.

Figura 1: Relação entre notas de matemática e programação



Fonte: Dados fornecidos pelo Grupo de Apoio à Tecnologia da Informação

Para analisar mais a fundo essa relação que poderia existir entre as notas de matemática e o desempenho final em programação, foi feita uma análise descritiva dos dados por meio de um gráfico de pontos, que pode ser visto na Figura 1. Nele, observa-se que parece haver uma relação positiva, mas bastante fraca entre ter sucesso em matemática e ter um bom desempenho em programação.

A partir dos resultados alcançados, observou-se por meio de um método estatístico se tal relação realmente se configurava. Ao realizar o teste de correlação de Spearman, obteve-se o coeficiente $\rho = 0,19$, que significa uma relação positiva, mas insignificante. Desta forma, apesar de perceber na análise descritiva certa relação entre as variáveis, não se pode considerar que ela tenha significância estatística.

4.2 Por que não se encontrou relação entre matemática e programação?

A literatura já possui diversos trabalhos que tratam o tema abordado neste estudo. Existem inúmeros resultados que levam a crer que o desempenho em matemática pode auxiliar os professores a preverem com antecedência quais dos alunos terão dificuldade em aprender a programar (BYRNE, 2001). No entanto, neste caso, não conseguimos encontrar uma relação significativa entre as notas de matemática e programação. Discutiremos quais fatores podem ter levado a este resultado.

Primeiramente, uma limitação deste trabalho é ter uma amostra pequena dados. Embora tenhamos coletado dados de 3 períodos, cada turma possuía um número bem reduzido de alunos. Essa pouca quantidade de dados pode limitar a eficácia de testes estatísticos, mesmo aqueles como o Ranking de Correlação de Spearman, que aceita um número relativamente pequeno de observações. Este motivo, portanto, poderia esconder possíveis relações que houvessem nos dados.

Outro fator que pode ter causado confusão na análise foi o fato da dinâmica da disciplina ter mudado ao longo dos períodos. Os critérios de avaliação da disciplina de programação evoluíram entre 2015.1 e 2016.1, o que pode ter contribuído para tornar implícita qualquer relação que houvesse entre os dados.

5 CONCLUSÃO

Este trabalho teve como objetivo explorar uma possível relação entre o desempenho prévio em matemática e as notas finais da disciplina de programação no curso de Sistemas de Informação. Descobriu-se que, apesar de existir uma tendência positiva entre as variáveis, este resultado não possui significância estatística. Portanto, no contexto deste estudo de caso, ter aptidão em matemática não implica ser bom em programação.

Como limitações deste trabalho, estão o tamanho da amostra, que corresponde a notas de 3 períodos da disciplina de programação e o fato da disciplina ter mudado ao longo do tempo. Estes dois aspectos devem ser levados em consideração como possíveis ameaças à validade, mas não desmerecem o rigor científico empregado no estudo.

Não ter sido encontrado relação entre o desempenho entre matemática e programação vai de encontro com o que existe na literatura. Portanto, apesar das ameaças à validade citadas

neste trabalho, possuímos uma contribuição importante, já que esta abre caminho para que se tente entender o que levou os alunos de Sistemas de Informação a apresentarem um comportamento tão peculiar.

Como trabalhos futuros, pretende-se aumentar o universo de dados e utilizar artifícios para controlar parcialmente mudanças que tenham acontecido entre períodos. Além disso, como forma de aumentar a validade externa, utilizar dados similares de outras instituições de ensino, para tornar mais sólidos os resultados.

HAVING GOOD GRADES IN MATH IMPLIES HAVING GOOD PERFORMANCE IN COMPUTER PROGRAMMING? A CASE STUDY

ABSTRACT

There is vast evidence in literature showing that computer programming courses have high failure rates. A possible solution would be to detect in advance predicting variables for students' low performance so that instructors could support risk groups during early stages. This study explored the effect of students' grades in math to their performance in programming. The work aimed to understand if both variables correlate to each other in order to use math grades as predictors of programming performance. We used data from a Computer Science course in a private college in Patos-PB. We discovered that there is no significant correlation between the variables. Thus, having good grades in math does not imply having good performance in programming considering the context of this work.

Keywords: Teaching Programming. Difficulties in Learning Programming. Correlation between Math and Programming.

REFERÊNCIAS

AMBRÓSIO, Ana Paula L. et al. **Programação de computadores:** compreender as dificuldades de aprendizagem dos alunos. 2011.

BERGIN, Susan; REILLY, Ronan. **Programming:** factors that influence success. In: ACM SIGCSE Bulletin . ACM, 2005. p. 411-415.

BYRNE, Pat; LYONS, Gerry. **The effect of student attributes on success in programming.** In: ACM SIGCSE Bulletin. ACM, 2001. p. 49-52. TAN, Phit-Huan;

CHAMILLARD, A. T. **Using student performance predictions in a computer science curriculum.** In: ACM SIGCSE Bulletin . ACM, 2006. p. 260-264. Computer Programming, p.207-235. Hillsdale, NJ: Lawrence ErlbaumAssociates.

COSTA, Túlio H.; BUBLITZ, Frederico M. **Análise dos Principais Problemas que Afetam Alunos de Programação:** uma investigação empírica no Estado da Paraíba. 2013.

DA ÁREA, Diretrizes Curriculares de Cursos de Computação e Informática. **COMISSÃO DE ESPECIALISTAS DE ENSINO DE COMPUTAÇÃO E INFORMÁTICA–CEEInf.** MEC–Secretaria de Educação Superior, 1999.

DOS SANTOS, Rodrigo Pereira; COSTA, Heitor Augustus Xavier. **Análise de Metodologias e Ambientes de Ensino para Algoritmos, Estruturas de Dados e Programação aos iniciantes em Computação e Informática.** INFOCOMP Journal of Computer Science, v. 5, n. 1, p. 41-50, 2006.

GIL, Antônio Carlos. **Como classificar as pesquisas.** v. 4, p. 44-45, 2002.

GOMES, Anabela; HENRIQUES, Joana; MENDES, António. **Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores.** Educação, Formação & Tecnologias-ISSN 1646-933X, v. 1, n. 1, p. [93-103], 2008.

KOLIVER, Cristian; DORNELES, Ricardo Vargas; CASA, Marcos Eduardo. **Das (muitas) dúvidas e (poucas) certezas do ensino de algoritmos.** In: XII Workshop de Educação em Computação. 2004.

MCDONALD, John H. **Handbook of biological statistics.** Baltimore, MD: Sparky House Publishing, 2009.

NETO, Wilson Castello Branco; SCHUVARTZ, Aguinaldo Antonio. **Ferramenta Computacional de Apoio ao Processo de Ensino-Aprendizagem dos Fundamentos de Programação de Computadores.** In: Anais do Simpósio Brasileiro de Informática na Educação. 2007. p. 520-528.

PIMENTEL, Edson P. et al. **Avaliação contínua da aprendizagem, das competências e habilidades em programação de computadores.** In: Anais do Workshop de Informática na Escola. 2003. p. 533-544.

SILVA, IFA; SILVA, Ivanda Maria Martins; SANTOS, Marizete Silva. **Análise de problemas e soluções aplicadas ao ensino de disciplinas introdutórias de programação**. Universidade Federal Rural de Pernambuco, Recife–PE, 2009.

TING, Choo-Yee; LING, Siew-Woei. **Learning Difficulties in Programming Courses: Undergraduates' Perspective and Perception**. In: Computer Technology and Development, 2009. ICCTD'09. International Conference on . IEEE, 2009. p. 42-46.