

Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Coordenação de Pós-Graduação em Ciência da Computação

Avaliação de uma Abordagem Personalizada de Ensino no Contexto de Programação Introdutória

Luiz Augusto de Macêdo Moraes

Dissertação submetida à Coordenação do Curso de Pós-Graduação em
Ciência da Computação da Universidade Federal de Campina Grande -
Campus I como parte dos requisitos necessários para obtenção do grau
de Mestre em Ciência da Computação.

Área de Concentração: Ciência da Computação

Linha de Pesquisa: Engenharia de Software

Dalton Dario Serey Guerrero e Jorge César Abrantes de Figueiredo
(Orientadores)

Campina Grande, Paraíba, Brasil

©Luiz Augusto de Macêdo Moraes, 21 de Setembro de 2015

Resumo

Nas disciplinas introdutórias de programação existem alunos que aprendem em ritmos diferentes. Apesar disso, muitos professores permanecem adotando abordagens nas quais todos são tratados da mesma forma. Infelizmente, aqueles que não conseguem acompanhar o ritmo da disciplina acabam aprendendo o conteúdo superficialmente, pois precisam estudar novos conceitos antes de dominar os anteriores. Pensando nisso, professores da Universidade Federal de Campina Grande (UFCG) resolveram aprimorar a abordagem de ensino de Programação I, utilizando elementos do *Mastery Learning* — uma filosofia de aprendizagem baseada no princípio de que todos podem aprender, desde que sejam oferecidas as condições adequadas. O intuito desta mudança foi permitir que cada aluno pudesse aprender em seu próprio ritmo e, assim, dominar os conteúdos antes de estudar os próximos. A nova abordagem foi implementada na disciplina e bem aceita entre os professores. No entanto, não havia evidência científica de que ela tivesse trazido benefícios ao processo de ensino-aprendizagem. Este trabalho surgiu como uma oportunidade para investigar os efeitos da mudança de método nos estudantes e descobrir os aspectos positivos e negativos da nova abordagem. Descobrimos que o *Mastery Learning* tornou os alunos mais focados e possibilitou que eles levassem menos dúvidas adiante. Além disso, à medida que o método evoluiu na disciplina, houve uma melhora considerável no desempenho dos estudantes. Este trabalho reuniu evidências de que a introdução do *Mastery Learning* em Programação I melhorou vários aspectos do processo de ensino-aprendizagem e reuniu lições aprendidas pelos professores ao longo da mudança de método.

Abstract

Students in introductory programming classes learn at different paces. Nevertheless, many professors remain adopting approaches in which all are treated equally. Unfortunately, students who fail to keep pace with the course end up learning superficially, because they study new concepts before mastering the previous ones. Professors from the Federal University of Campina Grande decided to improve the teaching method used in their CS1 course using elements of Mastery Learning — a philosophy of learning based on the principle that everyone can learn under adequate conditions. They changed the teaching approach hoping to allow every student to learn at his/her own pace and master the content before studying the next one. The new method was implemented in the course and well accepted among instructors. However, there was no scientific evidence to support its benefits in the teaching-learning process. This work is an opportunity to investigate the effects of Mastery Learning in students and find out the positive and negative aspects of its implementation. We found that students were more focused on their difficulties and had less doubts. In addition, as the method has evolved in the course, there was considerable improvement in student performance. This work gathered evidence that the use of the Mastery Learning in CS1 improved various aspects of the teaching-learning process and brought together lessons learned by instructors throughout the change of method.

Agradecimentos

Agradeço a meus pais, Luiz e Terezinha, por todo amor, compreensão e apoio que me deram até agora. Sou imensamente grato e realizado por ser seu filho e lhes dedico todas as minhas conquistas, passadas e vindouras.

Agradeço a Jaqueline — meu grande amor — pelas horas de terapia ao telefone, pela paciência que teve em ouvir minhas lamentações sobre o mestrado e por sempre me apoiar e amar.

Agradeço a Tia Lourdes, que me acolheu em seu lar por dois anos, fazendo eu me sentir como se nunca tivesse saído de casa. Nunca vou esquecer dos seus “rubacões” com carne de sol.

Agradeço ao curso de pós-graduação em Ciência da Computação, da UFCG, que me acolheu de braços abertos e me deu a oportunidade de ver a ciência com outros olhos.

Agradeço a Dalton e Jorge — meus orientadores — que me guiaram e lapidaram durante estes últimos anos. Vou levar seus ensinamentos para a vida e, principalmente, para a academia.

Agradeço a Izabela (Iza) por sempre me ajudar nas mais diversas situações, sejam relacionadas ao mestrado ou a Campina Grande. Sua amizade transpassou as paredes do SPLAB e da UFCG.

Agradeço aos companheiros do SPLAB, principalmente Matheus, Taciano e Kláudio pelas discussões sobre minha pesquisa e pelas conversas agradáveis durante a pausa para o café.

Conteúdo

1	Introdução	1
1.1	Contextualização	1
1.2	Iniciativa na UFCG	3
1.3	Foco do trabalho	4
1.4	Design da pesquisa	5
1.5	Resultados	6
1.6	Limitações	7
1.7	Terminologia	8
1.8	Estrutura do trabalho	9
2	Fundamentação teórica	10
2.1	<i>Mastery Learning</i>	10
2.1.1	Modelo de aprendizagem escolar	10
2.1.2	A filosofia posta em prática	12
2.1.3	Terminologia	13
2.2	<i>Learning for Mastery</i>	13
2.2.1	História	13
2.2.2	Características	14
2.2.3	Eficácia	16
2.2.4	Desvantagens	17
2.3	<i>Personalized System of Instruction</i>	18
2.3.1	História	18
2.3.2	Características	19
2.3.3	Eficácia	21

2.3.4	Desvantagens	21
3	O <i>Mastery Learning</i> na UFCG	22
3.1	Primeiros passos	23
3.2	Evolução	25
3.3	Design atual	27
3.3.1	Aprendizagem	28
3.3.2	Avaliação	29
3.3.3	Correção e Aperfeiçoamento	30
3.4	Conclusões	31
4	Os efeitos do <i>Mastery Learning</i> no comportamento acadêmico	33
4.1	Metodologia	33
4.1.1	<i>Dataset</i>	34
4.1.2	Métricas	34
4.1.3	Análise	39
4.2	Dispersão de estudos	39
4.3	Rotina de estudos	41
4.4	Cobertura do conteúdo	42
4.5	Discussões	44
4.5.1	Aprovados por média	44
4.5.2	Aprovados na final	45
4.5.3	Reprovados na final	46
4.5.4	Reprovados por média	47
4.6	Conclusões	47
5	O <i>Mastery Learning</i> e o desempenho acadêmico	50
5.1	Metodologia	51
5.1.1	<i>Dataset</i>	51
5.1.2	Métricas	51
5.1.3	Análise	52
5.2	Efeitos na aprovação e reprovação	53

5.3	A evolução do método e o desempenho	54
5.4	Discussões	56
5.4.1	Domínio x cobertura	56
5.4.2	Aprendendo com os erros	58
5.5	Conclusões	59
6	O <i>Mastery Learning</i> sob a perspectiva dos estudantes	61
6.1	Metodologia	61
6.1.1	Participantes	62
6.1.2	Coleta de dados	63
6.1.3	Análise	63
6.2	Aceitação	63
6.2.1	Satisfação pelo <i>Mastery Learning</i>	64
6.2.2	Aprovação das características	64
6.3	Preferência	70
6.3.1	Continuação nos próximos períodos	70
6.3.2	Continuação após reprovação	72
6.3.3	Expansão para outra disciplina	73
6.4	Prós e contras	75
6.4.1	Vantagens	75
6.4.2	Desvantagens	77
6.5	Conclusões	78
7	Discussões	81
7.1	E aí, o <i>Mastery Learning</i> deu certo?	81
7.2	Por que todos não usam o <i>Mastery Learning</i> ?	83
7.3	Lições aprendidas	84
7.4	O <i>Mastery Learning</i> vale a pena?	86
8	Considerações finais	88
A	Questionário sobre a aprovação do <i>Mastery Learning</i> pelos estudantes	97
A.1	Domínio de unidades	97

A.2 Ritmo independente	97
A.3 Materiais instrucionais	98
A.4 Feedback	98
A.5 Tutoria	98
A Plano de Curso de Programação I - 2014.2	100

Lista de Figuras

2.1	Processo de instrução no <i>Learning for Mastery</i> , segundo Guskey [32]	15
3.1	Dinâmica das unidades no <i>Mastery Learning</i> em Programação I	24
3.2	Mudanças no método de Programação I, ao longo dos períodos	27
4.1	Exemplo de dispersão de estudo	35
4.2	Exemplo de rotina de estudos	37
4.3	Exemplo da cobertura de conteúdos	38
4.4	Dispersão dos diferentes perfis de alunos em cada método	40
4.5	Dispersão dos alunos aprovados e reprovados em cada período	40
4.6	Rotina dos alunos em cada método	41
4.7	Cobertura do conteúdo em cada método, separada pela situação final dos alunos	43
4.8	Cobertura das unidades pelos alunos aprovados e reprovados em cada período	43
5.1	Distribuição das taxas de aprovação dos últimos 6 anos em P1	53
5.2	Taxas de aprovação e reprovação em cada período do <i>Mastery Learning</i>	54
5.3	Distribuição da nota dos minitestos práticos a cada período	55
6.1	Preferência pela continuação do <i>Mastery Learning</i> em P1	71
6.2	Preferência pela continuação do <i>Mastery Learning</i> em turmas de alunos que reprovaram	73
6.3	Preferência pela expansão do <i>Mastery Learning</i> para P2	74

Lista de Tabelas

2.1	Diferenças entre <i>Learning for Mastery</i> e <i>Personalized System of Instruction</i> , segundo [49]	12
6.1	Caracterização dos alunos matriculados em Programação I	62
6.2	Satisfação pelo <i>Mastery Learning</i> entre períodos	64
6.3	Aprovação do domínio de unidades, separada por períodos	65
6.4	Aprovação do domínio de unidades em 2014.1	65
6.5	Aprovação do ritmo individual, separada por períodos	66
6.6	Aprovação do ritmo individual em 2014.1	66
6.7	Aprovação dos roteiros, separada por períodos	67
6.8	Aprovação dos materiais instrucionais em 2014.1	68
6.9	Aprovação do <i>feedback</i> dos testes automáticos, separada por períodos . . .	68
6.10	Aprovação do <i>feedback</i> em 2014.1	69
6.11	Aprovação da tutoria, separada por períodos	69
6.12	Aprovação da tutoria em 2014.1	70

Capítulo 1

Introdução

1.1 Contextualização

Existe um fato que é conhecido entre muitos professores de programação introdutória: os alunos parecem aprender em ritmos diferentes. É comum se deparar com estudantes resolvendo problemas sem dificuldade nas primeiras semanas, enquanto outros ainda lutam para entender conceitos simples de entrada e saída, por exemplo. Essa diferença pode estar ligada ao fato das turmas introdutórias de programação possuírem, pelo menos, dois ou três perfis de indivíduos [21], cujas características (e.g. experiência anterior, desempenho em disciplinas correlatas, QI, etc.) contribuem para seu sucesso ou fracasso. Enquanto alguns conseguem acompanhar o ritmo do professor e aprender facilmente a programar, outros precisam de mais tempo.

Os alunos que não acompanham o ritmo do professor acabam levando dúvidas adiante e sentem mais dificuldade em aprender novos conceitos. Segundo Robins [47], isso acontece porque o currículo de programação é altamente integrado e cumulativo: quando o sujeito estuda um conceito, ele precisa saber utilizá-lo ao longo da disciplina. Além disso, os indivíduos aprendem novos assuntos a partir do conhecimento que eles já possuem. Se um estudante não dominou expressões lógicas, por exemplo, ele certamente terá dificuldades em aprender estruturas condicionais. Portanto, quando os alunos estudam mas não dominam o conteúdo, eles tendem a acumular dúvidas à medida que a disciplina avança.

Embora muitos dos estudantes que sentem dificuldades continuem frequentando as aulas, é difícil para eles aprender novos conceitos, já que lhes faltam uma base. Infelizmente,

grande parte dos estudantes que se encontra nesta situação não sabe como se recompor e, por conta disso, fica desmotivada. De acordo com Rocha et al. [48], quando os alunos não conseguem aprender no ritmo do professor, eles podem passar a ver a disciplina como algo aversivo e terminam se preocupando apenas em conseguir a nota mínima para serem aprovados. No final, eles acabam com grandes lacunas em seu conhecimento, pois aprendem o conteúdo apenas superficialmente.

O fato dos alunos não dominarem o conteúdo de programação e acumularem dúvidas ao longo da disciplina também pode ocasionar prejuízos a longo prazo. O estudante que aprende um conteúdo de forma superficial tende a apresentar dificuldades em reaprendê-lo futuramente [27]. Para justificarmos tal afirmação, precisamos entender como os indivíduos aprendem um novo conceito. De acordo com Gentile e Lalley [27], a aprendizagem inicial ocorre em 3 fases:

Prontidão Na primeira etapa, o aluno tenta criar conexões entre os conceitos antigos e o novo, a fim de compreendê-lo. A prontidão está intimamente relacionada à definição de ritmo, abordada neste trabalho. Enquanto alguns sujeitos ficam “prontos” mais cedo para aprender um conteúdo (pois já possuem a base necessária para isto), outros — cuja habilidade ou experiência anterior não existe — precisam de mais tempo para adquirir os requisitos necessários a sua compreensão.

Domínio inicial Na segunda fase, o indivíduo passa, de fato, a aprender. Isso acontece porque ele já possui os pré-requisitos necessários para compreender os novos conceitos. Neste modelo, a aprendizagem ocorre em função do tempo (que pode variar, de sujeito para sujeito). Desta forma, o aluno domina um conteúdo quando ele dedica tempo suficiente para estudá-lo.

Esquecimento Esta última etapa é inevitável no processo de aprendizagem. Mesmo se o aluno conseguir dominar 80% a 100% do conteúdo, ele irá esquecê-lo, à medida que for deixando de estudar os conceitos. Mas esquecer, neste caso, não significa apagar do cérebro tudo o que acabou de ser aprendido. Quanto maior o domínio do conteúdo, mais lento e fraco será o esquecimento.

Quando dominamos um conteúdo, nosso cérebro cria conexões fortes, que nos permitem reaprender mais rápido o que vai sendo esquecido. Por outro lado, ao aprendermos superfi-

cialmente, acabamos criando conexões fracas, que dificultam a recuperação de informação e tornam a reaprendizagem muito mais lenta. É por causa disso que muitos alunos reprovados sentem a mesma dificuldade ao pagarem a disciplina pela segunda vez e acabam dizendo a famosa frase: “eu não me lembro de ter visto isso no semestre anterior”. Como eles não dominaram o conteúdo, precisam reaprendê-lo praticamente do zero. Por conta disso, muitos falham por sucessivos períodos e acabam desenvolvendo um sentimento de impotência [27].

Em suma, é preciso entender que a turma de programação possui alunos com os mais diversos ritmos de aprendizagem. Se os professores não estabelecerem medidas para auxiliar a aprendizagem de todos — independente do ritmo da disciplina — aqueles que não conseguirem acompanhá-los, possivelmente ficarão para trás e poderão se prejudicar em curto ou longo prazo.

1.2 Iniciativa na UFCG

Na Universidade Federal de Campina Grande (UFCG), os professores de Programação I perceberam que, apesar do ensino de qualidade e da estrutura do curso, parte dos alunos não conseguia acompanhar o ritmo das aulas e se prejudicava em função disso. A partir desta problemática, eles resolveram desenvolver uma estratégia que auxiliasse nos estudos de todos, independentemente do ritmo da disciplina.

Foram criados um banco de questões sobre os conceitos de programação introdutória e um sistema de correções automáticas de exercícios (TST). Os professores esperavam que este material ajudasse tanto os alunos que aprendiam mais rápido — porque eles poderiam praticar bastante, inclusive conteúdos que ainda não tinham sido abordados — quanto aqueles que não conseguiam acompanhar o ritmo da disciplina — já que eles teriam uma série de questões para estudar e receber *feedback* imediato sobre a corretude de suas respostas.

Com a implantação do TST, os alunos passaram a ter a oportunidade de estudar qualquer conceito por conta própria. No entanto, parte deles ainda não conseguia aprender a tempo o que era ministrado nas aulas e acabava perdendo o foco sobre o que precisava ser estudado. Eles agora tinham o material para aprender num ritmo independente, mas não sabiam como aproveitá-lo da melhor maneira. Era necessária uma mudança maior, que afetasse a rotina de estudo deles. Era necessária uma mudança no método de ensino da disciplina.

Os professores procuraram na literatura por abordagens educacionais que atendessem as necessidades dos diferentes tipos de alunos. Acabaram se identificando com o *Mastery Learning*, uma filosofia educacional idealizada na década de 60 por Benjamin Bloom [11], que tenta proporcionar uma experiência próxima ao ensino de um-para-um. Para Bloom, a situação ideal de aprendizagem se traduz na interação entre o tutor e um único aprendiz [30]. A partir deste pressuposto, ele selecionou os elementos mais importantes desta modalidade e idealizou o que ficaria conhecido como o *Mastery Learning*. Esta filosofia de aprendizagem prega que todos os alunos conseguem aprender, caso sejam oferecidas as condições instrucionais apropriadas. Portanto, em tese, o professor que seguir estes preceitos, permitirá aos alunos de diferentes perfis uma aprendizagem efetiva do conteúdo.

O *Mastery Learning* foi aplicado na prática por meio de duas abordagens: o Personalized System of Instruction (PSI) [39] e o Learning for Mastery (LFM) [11]. A primeira foi amplamente utilizada no ensino superior — em cursos de diversas naturezas — e tem o foco na aprendizagem individualizada; a segunda foi mais aplicada no ensino básico e é focada na aprendizagem em grupos. Os dois métodos têm em comum os elementos básicos do *Mastery Learning*: foco no domínio¹ do conteúdo, *feedback* constante e atividades corretivas ou de aperfeiçoamento. Todas estas características contribuem para que os alunos de diferentes perfis possam avançar na disciplina sem levar dúvidas adiante e ter uma aprendizagem mais sólida e focada².

Ao utilizarem os princípios do *Mastery Learning*, os professores teriam a garantia de que nenhum aluno avançaria na disciplina sem ter dominado cada conceito. O método adotado na UFCG foi baseado no PSI e no LFM, pois eles foram planejados para não interferir no currículo típico da sala de aula [10], o que tornaria mais fácil a adaptação dos professores e alunos.

1.3 Foco do trabalho

Desde 2013, o *Mastery Learning* vem sendo utilizado em Programação I na UFCG e, aparentemente, tem proporcionado resultados positivos à disciplina. Apesar dos efeitos da mudança

¹Dominar, neste sentido, significa aprender efetivamente o conteúdo; não levar dúvidas adiante

²Vários estudos foram expostos na literatura a fim de comprovar os efeitos positivos do *Mastery Learning* nos diversos níveis de ensino [10, 35, 36, 42, 52].

de método terem sido notados pelos professores, nenhuma prova concreta de sua eficácia foi explorada até agora. A satisfação dos professores conta como um ponto positivo para a continuação do método, mas é necessário reunir evidências científicas que indiquem se a abordagem realmente melhorou de alguma forma o processo de ensino-aprendizagem da disciplina.

Este trabalho não propôs uma solução para os problemas discutidos neste capítulo, muito menos encontrou uma substituição definitiva para o método tradicional. O objetivo desta pesquisa foi analisar a experiência da utilização do *Mastery Learning* na UFCG e discutir os aspectos positivos e negativos que foram vivenciados durante os primeiros anos de sua implementação em Programação I. Como contribuições desta pesquisa estão a avaliação do método na UFCG, as métricas criadas para avaliar o comportamento dos alunos e as lições aprendidas pelos professores durante os primeiros dois anos de utilização do *Mastery Learning* na disciplina.

Para avaliarmos os efeitos do *Mastery Learning* em Programação I, analisamos o comportamento e o desempenho dos alunos durante o método tradicional³ e no *Mastery Learning*. Além disso, investigamos a evolução do último método ao longo dos períodos e a consequência disso no desempenho das turmas. Por fim, exploramos a aceitação dos alunos quanto à mudança de abordagem. No final deste trabalho, teremos respondido as seguintes questões de pesquisa:

1. Existem diferenças entre a rotina de estudo dos alunos de Programação I no método tradicional e no *Mastery Learning*?
2. Há indícios de que o *Mastery Learning* influenciou o desempenho dos alunos de Programação I, em relação àqueles que passaram pelo método tradicional?
3. O método foi aceito pelas turmas de Programação I que passaram por ele?

1.4 Design da pesquisa

A avaliação do método foi feita a partir de duas perspectivas: (1) *entre métodos*, onde analisamos se o *Mastery Learning* apresentou benefícios em relação ao método tradicional; e

³Neste trabalho, quando falamos em método tradicional, estamos nos referindo à abordagem que era utilizada antes do *Mastery Learning*, que consistia em aulas expositivas e avaliação puramente somativa.

(2) *no método*, onde investigamos se os sucessivos aprimoramentos ao método trouxeram benefícios à turma. Foram realizados 3 estudos: dois quantitativos, sobre a influência do *Mastery Learning* no desempenho e na rotina de estudo dos alunos; e um misto, relativo à aceitação dos alunos ao método e as possíveis vantagens e desvantagens de sua utilização. Os dois primeiros estudos contemplaram as duas perspectivas citadas anteriormente (entre métodos e no método) e avaliaram a influência do *Mastery Learning* na disciplina. Já o último, explorou a percepção dos alunos em relação à utilização desta nova abordagem.

A primeira pesquisa foi do tipo *ex post facto* e analisou os dados das submissões de exercícios em Programação I entre 2011 e 2014. Foram exploradas relações entre a utilização do *Mastery Learning* e o comportamento dos estudantes na disciplina. Os fatores investigados foram: a dispersão de estudo dos alunos, sua rotina de estudo e a cobertura do conteúdo pelos diferentes perfis de sujeitos.

O segundo estudo *ex post facto* [17] reuniu dados sobre o desempenho e as taxas de aprovação e reprovação das turmas de Programação I, entre os anos de 2009 e 2014. Foi analisada a diferença entre as taxas de aprovação nas turmas que passaram pelo método tradicional e as que vivenciaram o *Mastery Learning*. Além disso, foi investigada uma possível relação entre a evolução do *Mastery Learning* e o desempenho das turmas, representado pelas notas dos minitests e taxas de aprovação e reprovação.

No terceiro estudo, foi investigada a aceitação dos alunos em relação ao *Mastery Learning* e suas percepções quanto às vantagens e desvantagens da implementação do método em Programação I. Primeiramente foi realizado um *survey* [19] com alunos de 2013.2 e 2014.1 para explorar sua aceitação e percepções em relação ao método. Num segundo momento, foram investigados os motivos das respostas do *survey* por meio da análise temática das entrevistas feitas com alunos em 2014.1.

1.5 Resultados

Após a implantação do *Mastery Learning*, os estudantes tiveram a oportunidade de aprender o conteúdo em seu próprio ritmo. E essa mudança na disciplina parece ter influenciado positivamente a rotina de estudo dos diferentes tipos de alunos. No método tradicional, eles possuíam uma rotina de estudos dispersa — estudando entre 1 e 3 conceitos por semana — e

pouco se dedicavam (entre 17% a 35% do tempo) em revisar os conteúdos antes de avançarem para os próximos. No *Mastery Learning*, os estudantes se focaram em menos unidades por semana (uma média de 1 conteúdo por semana) e passaram mais tempo revisando os conceitos antes de avançar (entre 35% e 44% do tempo), o que parece ter ajudado a diminuir as dificuldades em conceitos anteriores. Portanto, as turmas que passaram pelo *Mastery Learning* parecem ter apresentado uma rotina mais organizada e focada no conteúdo que realmente precisava ser aprendido.

A rotina de estudos no *Mastery Learning* é bastante diferente daquela que os alunos vivenciam em outras disciplinas. Nós investigamos como eles reagiram à dinâmica imposta pelo método e se eles ficaram satisfeitos com o novo formato da disciplina. Descobrimos que os alunos ficaram bastante satisfeitos com a nova abordagem (cerca de 86% de aprovação) e a maioria deles (entre 86% a 97% dos entrevistados) foi a favor de sua continuação em Programação I e sua expansão para programação II. Além disso, quase todos os problemas apontados pelos alunos que envolviam o *Mastery Learning* eram relacionados à forma como ele tinha sido implementado na disciplina, sendo que muitos já foram resolvidos em suas últimas versões.

Por fim, investigamos os efeitos do *Mastery Learning* no desempenho das turmas. Descobrimos que a transição do método tradicional para a nova abordagem não afetou a taxa de aprovação na disciplina, que continuou em cerca de 73%. Entretanto, vale salientar que as notas das turmas vêm aumentando desde que o método foi adotado e a proporção de aprovados por média tem crescido a cada semestre. Além disso, a versão final, implementada em 2014.2, trouxe resultados animadores em relação ao desempenho dos alunos, quando comparada às versões anteriores.

1.6 Limitações

Os estudos *ex post facto* possuem dados sobre turmas que cursaram a disciplina com o método tradicional e outras com o *Mastery Learning*. Devemos deixar claro que a implementação desses métodos sofreu alterações a cada novo semestre. Portanto, precisamos levar em consideração que poderá haver uma variação de desempenho mesmo entre as turmas que cursaram a disciplina com o mesmo método.

Deixamos claro que esse trabalho é apenas um estudo de caso e não possui generalidade. Nós não garantimos que os resultados alcançados na UFCG sejam observados em outras instituições. Entretanto, esta pesquisa nos proporcionou fortes indícios de que o *Mastery Learning* realmente melhorou a rotina de estudo dos alunos e fez com que eles apresentassem um desempenho satisfatório.

1.7 Terminologia

Este trabalho teve contato com duas áreas distintas: Ciência da Computação e Educação. Por isso, muitos dos termos vistos no texto possivelmente não fazem parte do vocabulário dos cientistas da Computação ou têm outro significado em seu contexto. Portanto, nós sentimos a necessidade de explicar antecipadamente alguns termos relacionados ao *Mastery Learning*.

Unidades (*unit*) são partes estruturadas de conteúdo que, juntas, formam o currículo da disciplina. No *Mastery Learning*, as unidades são formadas por poucos conceitos e possuem objetivos de aprendizagem bem definidos. Em programação I, por exemplo, cada estrutura da linguagem (*if*, *for*, *while*, etc.) é abordada em uma unidade diferente.

Domínio de unidades (*unit mastery*) é uma das principais características do *Mastery Learning*. Ela define que todo aluno precisa dominar uma unidade — isto é, alcançar os critérios de domínio estabelecidos pelo professor — antes de avançar para a próxima. Essa medida tem como objetivo impedir que os estudantes levem dúvidas adiante.

Critério(s) de domínio (*mastery criteria*) são regras que garantem ao professor que os alunos conseguiram dominar o conteúdo da unidade avaliada. Cada aluno precisa alcançar este(s) critério(s) para conseguir passar de unidade. Um critério comum é estabelecer que todo aluno precisa alcançar uma nota entre 8,0 e 10,0. Em Programação I, por exemplo, o critério adotado em 2014.2 foi resolver corretamente duas questões dentro de uma hora e meia.

Ritmo individual (*self-pacing*) outra característica do *Mastery Learning*, que tem como objetivo permitir a cada aluno aprender em seu próprio ritmo. Em Programação I, os alunos tem ritmos individuais porque podem estudar unidades diferentes, sem precisar acompanhar o ritmo do professor.

Atividades de aperfeiçoamento (*enrichment activities*) são estratégias voltadas aos alunos que conseguem demonstrar domínio em uma unidade. Elas servem para incentivá-los a explorar conceitos mais complexos a respeito do conteúdo que acabaram de estudar.

Atividades de correção (*corrective activities*) são estratégias direcionadas aos alunos que não conseguem dominar uma unidade. Elas servem para que eles reforcem suas dificuldades antes de progredirem na disciplina. Essas atividades podem variar: um encontro com o tutor, estudar por materiais alternativos ou até assistir uma aula de “tira-dúvidas” com o professor.

Tutoria (*tutoring*) é um dos elementos do *Mastery Learning* que serve para personalizar a experiência de aprendizagem. Na tutoria, cada tutor — que pode ser aluno ou contratado — fica responsável por acompanhar e auxiliar um grupo pequeno de estudantes durante todo o semestre.

1.8 Estrutura do trabalho

No capítulo 2, fundamentaremos os conceitos referentes à filosofia do *Mastery Learning* e de seus dois principais métodos: PSI e LFM. No capítulo 3, acompanharemos a evolução do *Mastery Learning* em Programação I e relataremos as experiências e lições aprendidas a partir de sua implementação na UFCG. Nos capítulos 4, 5 e 6, abordaremos os resultados das pesquisas em relação à rotina de estudos, ao desempenho e à aceitação dos alunos com o *Mastery Learning*, respectivamente. No capítulo 7, discutiremos se o método realmente deu certo e se vale a pena adotá-lo em outros contextos. O capítulo 8 faz uma breve conclusão.

Capítulo 2

Fundamentação teórica

Neste capítulo, abordaremos em mais profundidade as bases teóricas que norteiam o método de ensino avaliado nesta dissertação. Aprenderemos como ele surgiu, quais os resultados já apresentados na literatura e quais os desafios que o modelo ainda apresenta.

2.1 *Mastery Learning*

O *Mastery Learning* é uma filosofia de ensino-aprendizagem cujo objetivo é proporcionar a todos os perfis de alunos a chance de dominar o conteúdo. De acordo com essa corrente educacional, todos os estudantes (ou pelo menos 90% deles) são capazes de aprender qualquer conceito, desde que sejam oferecidas as condições instrucionais apropriadas para sua aprendizagem [10].

2.1.1 Modelo de aprendizagem escolar

Algumas características presentes no *Mastery Learning* já tinham sido discutidas por estudiosos como Comenius, Pestalozzi e Herbart [31], mas a definição teórica moderna desta filosofia foi estabelecida a partir do trabalho de Bloom [11], que se baseou no Modelo de Aprendizagem Escolar¹ proposto por Carroll [16].

De acordo com Carroll, o grau de aprendizagem dos alunos é definido pela relação entre o tempo gasto estudando e o tempo necessário para aprender (veja a Fórmula 2.1). Desta maneira, ele propõe que se o professor disponibilizasse para cada aluno o tempo necessário para

¹*Model for School Learning* em Inglês.

sua aprendizagem e, conseqüentemente, os discentes aproveitassem este tempo estudando, provavelmente todos conseguiriam dominar o conteúdo [10].

$$\text{Aprendizagem} = f \left(\frac{\text{Tempo gasto}}{\text{Tempo necessário}} \right) \quad (2.1)$$

Carroll [16] acreditava que a aprendizagem não é definida apenas pelas características inerentes aos alunos, mas também pela forma como a instrução é realizada. Com base nisso, ele decompôs as variáveis apresentadas na fórmula 2.1 da seguinte maneira:

$$\begin{aligned} \text{Aprendizagem} &= f \left(\frac{\text{Tempo gasto}}{\text{Tempo necessário}} \right) \\ &= f \left(\frac{\text{Tempo dedicado aos estudos ou Tempo permitido}}{\text{Tempo necessário (dominar assunto) + Tempo necessário (instrução)}} \right) \\ &= f \left(\frac{\text{Perseverança ou Oportunidade para aprender}}{\text{Aptidão + [Qualidade da instrução} \times \text{Abilidade de entender instrução]}} \right) \end{aligned} \quad (2.2)$$

Na versão completa do modelo de Carroll (fórmula 2.2), o tempo gasto é representado pela perseverança (período que o aluno está disposto a dedicar aos estudos) ou pela oportunidade de aprender (tempo disponibilizado para estudar determinado assunto). Já o tempo necessário é composto pela aptidão (tempo necessário para aprender um assunto sob condições instrucionais apropriadas), qualidade da instrução e habilidade para entender a instrução.

Baseado no modelo de Carroll, Bloom [11] propôs o seguinte raciocínio: se a aptidão representa apenas o tempo que um estudante demora para aprender — e não necessariamente o quanto pode ser aprendido — então todos poderiam obter sucesso se recebessem uma instrução de qualidade por tempo suficiente. Por isso, ao desenvolver o *Mastery Learning*, o autor definiu que o grau de aprendizagem seria fixo — por meio de um critério de domínio — e os fatores controlados pela instrução (qualidade da instrução e oportunidade para aprender) deveriam variar de acordo com a necessidade de cada aluno [10].

2.1.2 A filosofia posta em prática

Os conceitos que norteiam a filosofia do *Mastery Learning* foram postos em prática por meio de duas abordagens: *Learning for Mastery* (LFM) de Bloom [11] e *Personalized System of Instruction* (PSI) de Keller [39]. A primeira, desenvolvida no campo da Educação, é baseada em grupos, segue o ritmo do professor e tem sido amplamente utilizada no nível Básico de ensino. Já a segunda, concebida a partir da Psicologia, tem sido mais executada no ensino Superior e promove uma aprendizagem individual, no ritmo do aluno [10]. Veja a comparação entre as duas abordagens feita por Sheng e Lifeng [49] na Tabela 2.1.

Tabela 2.1: Diferenças entre *Learning for Mastery* e *Personalized System of Instruction*, segundo [49]

	LFM	PSI
Ambiente	Baseado em grupos	Individual
Ritmo	Ritmo do professor	Ritmo individual
Instrução	Ministrada pelo professor	Materiais escritos
Correção	Tutoriais individuais ou em grupo	Materiais de re-estudo
Duração	2-108 semanas	1 semestre
Critério de domínio	Menor	Alto
Nível de ensino	Básico	Superior

LFM: Learning For Mastery, PSI: Personalized System of Instruction

Embora o LFM e o PSI possuam diferenças, ambos promovem um processo de ensino-aprendizagem no qual os alunos têm a possibilidade de dominar o conteúdo. Gentile [27] selecionou os elementos que são considerados essenciais a todas as abordagens baseadas no *Mastery Learning*:

1. Objetivos instrucionais explícitos, que sejam sequenciados hierarquicamente e possam ser alcançados por todos os estudantes.
2. Avaliação com o objetivo de verificar a aprendizagem e fornecer *feedback* ligado aos objetivos instrucionais.
3. Instrução corretiva para os alunos que não conseguem atingir o nível de desempenho determinado.

O autor ainda afirma que nas melhores implementações do *Mastery Learning* existe um elemento adicional: o processo de aperfeiçoamento, que é uma oportunidade oferecida aos

alunos que dominam os conceitos na primeira avaliação. Eles são incentivados a explorar conceitos mais avançados, que exigem um nível maior de raciocínio.

2.1.3 Terminologia

Existe uma confusão na literatura em relação ao termo *Mastery Learning*, já que às vezes os autores estão se referindo à filosofia e em outras, ao modelo prático. Esta mistura aconteceu porque em seu artigo original, Bloom [11] propôs a teoria, mas deu o nome de *Learning for Mastery* à sua implementação. Posteriormente [12], o autor definiu que o termo *Mastery Learning* poderia se referir tanto à filosofia, quanto ao seu modelo prático.

Atualmente, o termo *Mastery Learning* está relacionado a uma série de programas educacionais e currículos que têm como base a ideia de que todos os alunos conseguem e precisam dominar os conteúdos ministrados. Guskey e Gates [35] declararam que muitas destas abordagens possuem pouca ou nenhuma semelhança com os princípios descritos por Bloom em seu trabalho original. Concluimos que *Learning for Mastery* e *Personalized System of Instruction* são considerados(as) modelos/versões/implementações do *Mastery Learning*. Nos próximos capítulos, utilizaremos o termo *Mastery Learning* para se referir à abordagem utilizada na UFCG.

2.2 *Learning for Mastery*

O LFM foi desenvolvido com o intuito de promover uma aprendizagem efetiva para a maioria dos alunos por meio de avaliações formativas e estratégias de recuperação. Neste método, o ritmo é imposto pelo professor e as atividades são feitas, em sua maioria, por meio da cooperação entre alunos [10].

2.2.1 História

Em 1968, Bloom [11] propôs a definição teórica moderna do *Mastery Learning* e, baseado nesta teoria, criou o *Learning for Mastery*. De acordo com Guskey [30], a teoria de Bloom foi aplicada em sala de aula e trouxe resultados positivos na aprendizagem dos alunos, o que gerou um grande interesse e entusiasmo entre educadores ao redor do mundo.

A partir do trabalho inicial de Bloom, o LFM passou a ser amplamente aplicado nos EUA e em menor frequência na Europa ocidental, Austrália, Ásia e América do Sul [38]. Além disso, o método foi utilizado nas mais diferentes áreas (matemática, ciências, línguas, ciência social, educação médica, etc.), níveis de ensino (fundamental, médio e superior) e instituições (hospitais, agências militares, prisões e empresas) [49].

2.2.2 Características

O objetivo do LFM é permitir que todos os alunos (ou pelo menos a maioria) consigam dominar os conceitos da disciplina dentro do prazo normal de aulas. Para Bloom [11], isto pode ser alcançado caso seja fornecido *feedback* apropriado e condições para que os alunos possam corrigir suas dificuldades antes estudarem novos conceitos.

No *Learning for Mastery*, os conceitos e habilidades necessários ao longo da disciplina são organizados em unidades de aproximadamente duas semanas. A cada nova unidade, os alunos assistem aulas e, em seguida, são submetidos a uma avaliação formativa, que serve para identificar os sujeitos que conseguiram ou não dominar o conteúdo. No entanto, Guskey [34] deixa claro que “avaliações formativas, sozinhas, pouco melhoram o desempenho dos alunos ou a qualidade do ensino”. Por esta razão, os estudantes também passam por procedimentos de *feedback*, *correção* e *aperfeiçoamento*², de acordo com suas necessidades [32].

No LFM, os alunos que conseguem atingir o critério de domínio são encaminhados para atividades de aperfeiçoamento, cujo objetivo é enriquecer o conhecimento recém adquirido. Por outro lado, aqueles que não conseguem dominar o conteúdo, recebem medidas corretivas, que variam desde a assistência de tutores, a disponibilização de materiais alternativos ou até atividades de cooperação [34]. Os estudantes que passam pelo processo de correção são submetidos a uma nova avaliação formativa no final da unidade. Entenda melhor o processo como um todo na Figura 2.1.

Além das medidas de *feedback*, correção e aperfeiçoamento, o *Learning for Mastery* exige que a disciplina possua um alinhamento instrucional, isto é, que todos os componentes do processo de ensino-aprendizagem sejam consistentes entre si. Veja as características do LFM com mais detalhes na próxima seção.

²Tradução adaptada da expressão *feedback, correctives and enrichment*

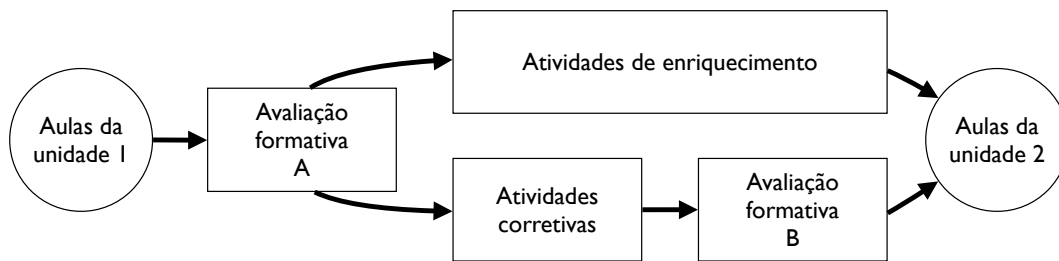


Figura 2.1: Processo de instrução no *Learning for Mastery*, segundo Guskey [32]

Feedback, correção e aperfeiçoamento

Em seu artigo *The sigma 2 problem*, Bloom [14] explica que a instrução de um para um provoca resultados excepcionais na aprendizagem da maioria dos estudantes. No entanto, o autor afirma que esta abordagem é muito custosa para os moldes educacionais, nos quais as turmas possuem dezenas (ou até centenas) de alunos. Bloom desenvolveu os processos de *feedback*, correção e aperfeiçoamento do LFM para tentar suprir este problema.

O *feedback* no LFM vai além da divulgação de notas e assume duas funções: ele é diagnóstico e prescritivo. Nesta abordagem, ao final da avaliação formativa, o aluno deve saber o que ele deveria ter aprendido, o que ele dominou e o que ele ainda precisa aprender. Além disso, o professor precisa prescrever ao aluno alternativas de como reforçar os conceitos que ele ainda possui dificuldades [32].

Embora o *feedback* exerça um papel essencial no LFM, ele deve ser seguido por medidas de correção. De acordo com Guskey [34], estes procedimentos são exatamente o que tornam a tutoria individual — instrução de um para um — tão eficiente, porque quando um aluno erra, seu tutor lhe mostra o erro e como consertá-lo. Bloom [11] recomenda que o professor forneça vários recursos educacionais alternativos para que os alunos possam escolher o que melhor se adapte ao seu estilo de aprendizagem. Alguns exemplos de recursos são a tutoria, os grupos de estudo, materiais multimídia, websites, etc.

Da mesma forma que os alunos com dificuldades recebem apoio no LFM, aqueles que conseguem dominar os conceitos são direcionados para atividades de aperfeiçoamento. Elas

têm como objetivo aperfeiçoar o que foi aprendido e incentivar os alunos a alcançarem níveis mais altos na taxonomia de Bloom [13] por meio de problemas mais desafiadores. De acordo com Guskey [34], os estudantes que conseguem dominar os objetivos de aprendizagem devem explorar tópicos que vão além do conteúdo restrito e superficial das unidades instrucionais.

Alinhamento instrucional

Bloom acreditava que o processo de ensino-aprendizagem fosse composto por 3 componentes: objetivos de aprendizagem, instrução e avaliação. Embora o *Learning for Mastery* seja essencialmente neutro a respeito destes três elementos, o método exige que eles sejam consistentes entre si [29]. Esta característica ficou conhecida como alinhamento instrucional [33].

Para entendermos melhor o alinhamento instrucional, vamos imaginar a seguinte situação: um professor de programação introdutória planeja implementar um curso baseado em LFM. Ele pretende ensinar os conceitos de programação estruturada como entrada e saída, condições, laços e funções. Para isso, ele precisa dividir o conteúdo em unidades e estabelecer quais os seus objetivos de aprendizagem em cada etapa. Em seguida, o professor precisa planejar suas aulas de modo que ele cubra todos os objetivos que foram definidos anteriormente. Por fim, o docente deverá planejar as avaliações para que todos os objetivos e conteúdos vistos em sala sejam testados adequadamente. De acordo com Guskey [33], os professores de todos os níveis já realizam este planejamento, mas no LFM eles precisam tomar decisões de uma maneira mais pensada e com propósitos claros.

2.2.3 Eficácia

Desde o final da década de 60, quando o *Learning for Mastery* foi proposto, centenas de estudos já foram realizados a respeito da sua eficácia no desempenho das turmas, em cursos das mais diversas naturezas [10]. Anderson [7] fez uma síntese dos principais trabalhos que resumem a eficácia do *Mastery Learning* e, consequentemente, do *Learning for Mastery*:

- Kulik et al. [42] analisaram 103 estudos que comparavam o desempenho de turmas baseadas em LFM ou PSI com outras que utilizavam a abordagem tradicional. De

acordo com as conclusões, 93% dos trabalhos apresentaram resultados positivos, sendo que o LFM apresentou maiores ganhos em relação ao PSI.

- Slavin [52] explorou 17 estudos e, por outro lado, encontrou um efeito pequeno do LFM no desempenho das turmas em testes padronizados.
- Guskey e Pigott [36] analisaram 46 trabalhos sobre LFM e reportaram que 89,1% deles possuíam resultados positivos.
- Guskey e Gates [35] investigaram 38 estudos sobre LFM e concluíram que 92,2% dos trabalhos apresentaram resultados positivos, sendo que o tamanho do efeito variam entre médio e grande.
- Block e Burns [10] analisaram 51 estudos sobre LFM e PSI e obtiveram resultados positivos em 89% dos casos, os quais apresentaram um efeito considerado grande.

Além de ter um efeito positivo no desempenho dos alunos, o LFM influenciou a atitude deles diante da disciplina. De acordo com 6 estudos [10, 20, 22, 36, 42, 55], a maioria dos 93 trabalhos analisados mostrou que os alunos no *Learning for Mastery* ficaram mais satisfeitos e tiveram atitudes mais positivas em relação aos estudantes de outros métodos de ensino.

2.2.4 Desvantagens

Apesar do LFM influenciar positivamente o desempenho e as atitudes dos alunos, alguns estudiosos apontaram que esse método não é perfeito. Cotton e Savard [20] afirmaram que um dos fatores limitantes do *Learning for Mastery* é o custo-benefício de sua implementação, já que é necessário um esforço maior para o gerenciamento da classe e assistência aos alunos.

Um outro ponto no LFM que recebeu críticas foi a questão dos alunos mais rápidos precisarem esperar seus colegas mais lentos para poderem avançar na disciplina. Sheng e Lifeng [49] explicaram que os estudantes com maior aptidão tendem a se beneficiar menos com o método porque eles precisam esperar seus colegas dominarem os conceitos.

2.3 *Personalized System of Instruction*

O PSI é uma abordagem individualizada de ensino, baseada nos princípios do *Mastery Learning*, que surgiu a partir da insatisfação de um grupo de professores brasileiros e norte americanos acerca dos métodos convencionais de ensino na década de 60. Essa abordagem foi idealizada com o intuito de recompensar os alunos ao invés de penalizá-los, além de promover o domínio dos conteúdos e aumentar a comunicação interpessoal em sala de aula [26].

2.3.1 História

O PSI foi idealizado em 1962 por Fred Keller, Gil Sherman, Rodolfo Azzi e Carolina Martuscelli Bori ao planejarem o primeiro curso do Departamento de Psicologia da Universidade de Brasília. Sua primeira versão foi aplicada num curso de curta duração na *Columbia University* em 1963 e o estudo piloto foi realizado em Brasília no ano seguinte [39].

Após as iniciativas iniciais, os professores Keller e Sherman continuaram executando o Plano de Keller³ na *Arizona State University*, onde ministraram um curso de análise de comportamento em 1965. Os resultados desse estudo foram expostos no trabalho “Good-bye, Teacher...”, que tem sido considerada a obra fundamental do PSI [40].

O método gradualmente foi sendo reconhecido além das fronteiras do Brasil e Estados Unidos e foi aplicado em países dos mais diversos continentes [40]. Apenas 10 anos após a publicação do “Good-bye, Teacher...”, já havia cerca de 3000 publicações relacionadas ao tema, sendo que os estudos abrangiam sua aplicação em diferentes cursos e níveis de educação [26].

O PSI vem sendo aplicado nos cursos das mais variadas áreas, como psicologia, química, engenharia, física, anatomia, música, ciência da computação, etc. [25]. O artigo de Rocha et al. [48] mostra experiências com o uso do PSI em uma disciplina introdutória de programação. Os resultados obtidos no trabalho são bastante positivos, já que o desempenho dos alunos aumentou significativamente em relação aos períodos anteriores, além haver uma queda brusca no número de desistências. Nilsen e Larsen [46] também relataram em seu trabalho que o número de notas altas aumentou após a utilização do método.

³O método PSI também ficou conhecido como Plano de Keller em homenagem a seu criador.

2.3.2 Características

Com o intuito de melhorar o desempenho dos alunos, o PSI possui uma série de características que substituem a longa tradição de punição, presente no método tradicional, por consequências positivas para o aprendizado [28].

As características do Plano de Keller são descritas como: (a) domínio de unidades; (b) ritmo individual⁴; (c) ênfase em materiais instrucionais escritos; (d) uso de tutores para auxiliar e avaliar os alunos; e (e) aulas com caráter motivacional.

Domínio de unidades

Cursos baseados no PSI são divididos em pequenas unidades que duram cerca de uma semana. Para o aluno avançar de módulo, ele precisa demonstrar domínio na unidade atual.

Para comprovar o domínio sobre uma unidade, os estudantes precisam atingir uma nota mínima, que normalmente varia entre 80% e 100% de acertos. Os alunos que não passam na avaliação podem repeti-la várias vezes, sem nenhuma penalização.

Fornecer oportunidades de recuperação aos estudantes elimina substancialmente o estigma de falha na disciplina. Com isso, as avaliações deixam de ser vistas como uma forma de punição e passam a ter um caráter positivo, de incentivar o domínio dos conteúdos [28].

Ritmo individual

Com o ritmo individual nenhum estudante é prejudicado devido ao ritmo empregado pelo professor nas aulas [44]. Quando o ritmo da disciplina é imposto pelo professor, os estudantes que possuem maior dificuldade não conseguem assimilar completamente os conceitos vistos na aula. Mas se o professor tentar diminuir o ritmo, isso pode prejudicar os alunos que estão mais avançados no curso.

Quando cada estudante pode ter seu próprio ritmo, ele pode avançar a disciplina rapidamente, ou passar mais tempo revisando tópicos mais complicados. Essa característica possibilita que os estudantes, mesmo com necessidades diferentes, atinjam os objetivos propostos no curso, caso tenham tempo suficiente. Entretanto, como os cursos possuem uma

⁴Os termos *unit mastery* e *self-pacing* foram traduzidos, respectivamente, para domínio de unidades e ritmo individual

duração, deve-se pensar em estratégias para eliminar a possibilidade de procrastinação por consequência do ritmo individual [37].

Ênfase em materiais escritos

Nos primeiros cursos baseados no Plano de Keller, o conteúdo instrucional era apresentado principalmente em forma escrita e não havia aulas expositivas. A disciplina possuía um roteiro de estudos o qual era planejado para auxiliar a aprendizagem dos estudantes. Estes roteiros possuem objetivos e questões de estudo para que os alunos se foquem no que precisam aprender [28].

Com a facilidade ao acesso de tecnologias da informação e comunicação, os materiais escritos não são mais a única alternativa para aprender. Hoje, com a Internet, o curso pode ser transmitido em vários formatos como áudio, vídeo, páginas da web, entre outros [26].

Tutores

A utilização de tutores nos cursos baseados em PSI possibilita que haja tanto um *feedback* imediato aos alunos, quanto avaliações contínuas. Além disso, de acordo com Keller [39], a presença de tutores aprimora o aspecto pessoal-social do processo educacional. Alves et al. [5] afirmam que o *feedback* é fundamental para a aprendizagem e que são necessárias avaliações formativas para que os estudantes possam revisar e melhorar sua qualidade de pensamento e compreensão.

Aulas de caráter motivacional

As aulas tradicionais vêm sendo a principal forma de transmissão do conhecimento desde as universidades medievais. Elas têm a vantagem de serem econômicas ao lidar com um grande número de estudantes, mas não conseguem proporcionar uma aprendizagem personalizada [41]. Por este motivo, as aulas expositivas não são prioridade no PSI. Keller [39] afirma que as aulas só devem ser ministradas quando os alunos já demonstrarem domínio do conteúdo, pois — no PSI — elas têm um papel puramente motivacional e funcionam como uma medida de aperfeiçoamento.

2.3.3 Eficácia

A grande maioria dos estudos que avaliaram o *Mastery Learning* nos últimos anos mostraram que os estudantes em cursos baseados no PSI aprendem o conteúdo melhor, lembram-se dele por mais tempo e aprovam a experiência de uma forma mais positiva, quando comparados a alunos no método tradicional [26].

Kulik et al. [43] realizaram uma meta-análise de 75 estudos que comparam o PSI com o método tradicional. De acordo com seus resultados, o Plano de Keller possui superioridade na maioria dos estudos analisados, mesmo entre os mais diferentes designs de curso.

Na literatura também existem estudos que comprovam a eficácia de componentes do PSI, como é o caso dos trabalhos de Boutell e Clifton [15] e Tatum e Lenel [53], que analisaram a relação entre os cursos baseados no ritmo individual e o desempenho dos estudantes. Ambos mostram que as notas nesses cursos são superiores, quando comparadas ao método tradicional. Outro componente estudado isoladamente foi o uso de tutores. De acordo com os resultados de Farmer et al. [24], estudantes que são auxiliados por tutores possuem notas finais melhores que aqueles que não possuem apoio.

2.3.4 Desvantagens

Embora o Plano de Keller possua efeitos positivos no desempenho [43] e na aceitação dos alunos [50], seu uso diminuiu bastante desde o final da década de 1970. De acordo com Eyre [23], muitas razões parecem ter contribuído para o declínio do PSI. Alguns dos motivos citados na literatura são o custo e esforço para implementação dos cursos baseados em PSI, a dificuldade para adaptar o calendário escolar ao ritmo individual dos alunos e o domínio de unidades, a hesitação de alguns professores em mudar de uma abordagem centrada no professor para outra centrada nos alunos e a resistência dos administradores em permitir um método onde os professores não ministram aulas [26].

Capítulo 3

O Mastery Learning na UFCG

No final de 2012, os professores de Programação I se reuniram para planejar o método de ensino do próximo semestre. Eles já estavam utilizando o TST — sistema de correções automáticas — há dois anos e conseguiam avaliar os exercícios e minitestes de forma ágil. A disciplina estava se voltando cada vez mais para os alunos, oferecendo-os um processo de aprendizagem mais independente dos professores. No entanto, alguns elementos ainda pareciam não se encaixar.

Por um lado, os alunos de Programação I podiam resolver exercícios de qualquer unidade e ter *feedback* imediato sobre suas soluções (graças ao TST, juntamente com o banco de questões que era disponibilizado desde os primeiros dias de aula), mas, por outro, toda a turma precisava assistir às mesmas aulas e fazer as mesmas avaliações. Embora parte da disciplina oferecesse aos alunos uma experiência única e personalizada, a outra parte continuava a tratá-los de uma forma genérica, como se “um tamanho servisse para todos”¹. Foi com a adoção do *Mastery Learning* que as coisas começaram a mudar.

No capítulo anterior, vimos que a utilização do *Mastery Learning* pode promover uma aprendizagem mais personalizada aos diferentes tipos de alunos, já que cada um consegue dominar o conteúdo em seu próprio ritmo. Durante este capítulo, acompanharemos como a implementação do método evoluiu em Programação I e quais as características da abordagem atual, que foi baseada nas pesquisas deste trabalho. Ao final, serão apresentadas as perspectivas sobre o que ainda falta ser aprimorado.

¹Referência ao termo em Inglês “one size fits all”, que é comumente utilizado para se referir a situações onde uma abordagem é utilizada para resolver todos os casos.

3.1 Primeiros passos

Os professores estavam frustrados por presenciarem recorrentes casos de sujeitos que não conseguiam acompanhar o ritmo das aulas e acabavam ficando para trás (ou desistindo). Eles já vinham adotando medidas para tornar os alunos cada vez mais independentes (como é o caso do banco de questões e do TST) e decidiram que, em 2013.1, a disciplina mudaria ainda mais nesta direção. Assim, eles começaram a utilizar elementos do *Mastery Learning* em Programação I, se baseando principalmente no artigo *Good-bye teacher...*, de Keller [39], onde o autor descreveu como deveria ser um curso baseado no ritmo dos alunos. Naquela época, o foco principal da nova abordagem foi assegurar que os sujeitos só estudariam novos conceitos depois que dominassem os anteriores.

Na versão inicial do *Mastery Learning*, o conteúdo era dividido em 10 unidades, mais o projeto final da disciplina² (veja a Figura 3.1). Os sujeitos estudavam 6 etapas básicas, que envolviam entrada e saída, condições e laços; e 4 avançadas, que abordavam a definição de funções e estruturas de dados da linguagem. Durante as unidades básicas, os alunos precisavam estudar os conteúdos em sequência, um de cada vez. Após a unidade 6, eles podiam escolher em que ordem estudariam o resto dos conceitos. Ao longo de cada semana, os alunos aprendiam por meio dos materiais instrucionais (livro, roteiros e banco de questões) e pelas aulas direcionadas à sua unidade atual.

O esquema de aulas direcionadas permitiu, em parte, que os alunos aprendessem em seu próprio ritmo. Como a disciplina possuía 3 professores, cada um era alocado para ensinar um conteúdo diferente por semana. A turma era dividida em 3 grupos — formados por alunos que estavam estudando unidades iguais — e cada professor ficava responsável por ensinar um dos grupos. Quando mais de 3 unidades começavam a ser exploradas pelos alunos, os docentes passavam a se focar nas últimas três ou abordavam os conteúdos de duas etapas na mesma aula. Isto é, os grupos passavam a ser formados por alunos que estavam em unidades próximas. Ao final de cada semana, os estudantes eram avaliados.

O processo de avaliação foi o que mais mudou após a adoção do *Mastery Learning*. Antes, ele era puramente somativo e composto por provas, minitestes e exercícios. Seu único

²O projeto só era destinado aos alunos que conseguissem passar em, pelo menos, 6 unidades. Nele, os alunos tinham a oportunidade de estudar por conta própria conceitos de orientação a objetos e explorar bibliotecas do Python.

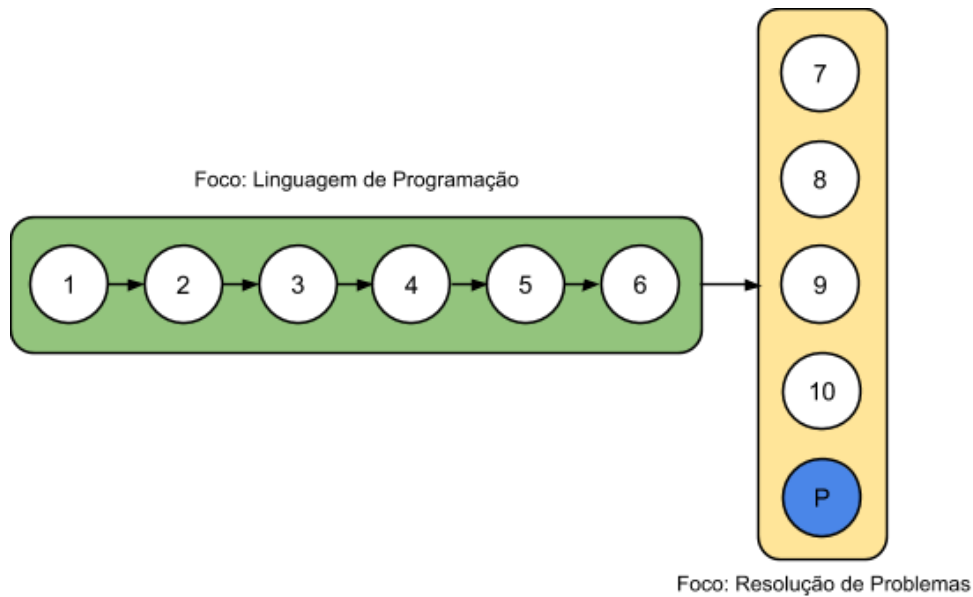


Figura 3.1: Dinâmica das unidades no *Mastery Learning* em Programação I

propósito era verificar se os alunos tinham aprendido ou não determinados conteúdos. A partir de 2013, o processo ganhou um aspecto mais formativo e passou a servir também como um *feedback* para os estudantes. Entre as avaliações, os alunos recebiam dos professores informações sobre o que tinham errado e o que precisava ser melhorado. Assim, eles poderiam reforçar suas dificuldades a partir dos materiais instrucionais ou tirar dúvidas com seus tutores³.

Antes do *Mastery Learning*, a disciplina já possuía minitests teóricos (MTTs) — que avaliavam se os alunos haviam entendido a sintaxe e a rotina de execução dos programas — e práticos (MTPs) — que avaliavam a habilidade em resolver problemas, utilizando os conceitos da unidade. No entanto, após a mudança de abordagem, os minitests se tornaram o principal componente de avaliação⁴ da disciplina. Para serem aprovados em uma unidade, os alunos precisavam alcançar os critérios de domínio. No período piloto, isso significava obter 7,0 no MTT (que não fazia parte da nota, mas servia como pré-requisito para liberar o MTP) e 6,0 em dois MTPs. Se os alunos não conseguissem atingir esse critério, eles teriam mais uma semana para tentar aprender os conceitos da unidade, antes de serem submetidos a uma nova avaliação. Este processo continuava até que eles conseguissem passar de unidade,

³Na tutoria um grupo pequeno de alunos é acompanhado por um tutor durante todo o semestre. Em Programação I, os tutores são graduandos ou pós-graduandos que cursaram a disciplina com notas elevadas.

⁴O principal critério de avaliação antes do *Mastery Learning* eram as provas. Os minitests, juntamente com os exercícios, serviam apenas como um complemento para a nota.

ou seja, atingissem o critério de domínio.

Como veremos nos capítulos 4 e 5, o período piloto do *Mastery Learning* causou uma mudança positiva na rotina de estudo dos diferentes perfis de alunos, pois proporcionou que eles tentassem superar suas dificuldades antes de progredir na disciplina. Por outro lado, a estratégia de avaliação acabou dificultando o progresso de boa parte da turma. Vários alunos passaram muito tempo presos nas unidades iniciais e não conseguiram cobrir todo o conteúdo antes do final do semestre. Por conta disso, os professores tiveram que aprimorar o processo de avaliação no período seguinte.

Na próxima seção, acompanharemos os sucessivos aprimoramentos ao método de Programação I, ao longo dos últimos dois anos. Parte das mudanças que aconteceram foram possíveis graças aos resultados deste trabalho, já que a pesquisa deste mestrado permitiu aos professores tomar decisões que melhoraram o método de forma significativa.

3.2 Evolução

Para tentar evitar o problema da falta de cobertura do conteúdo, que aconteceu em 2013.1, os professores adotaram no semestre seguinte critério de domínio mais flexível. Os alunos ainda teriam que passar em um miniteste teórico e dois práticos, mas eles poderiam realizar minitests de duas unidades ao mesmo tempo. Isso ajudou a diminuir os atrasos na turma, já que os sujeitos tinham a oportunidade de fazer mais avaliações em relação ao período anterior. Mesmo assim, boa parte da turma ainda ficou presa e foi prejudicada.

A constante retenção de alunos nas primeiras unidades — ocasionada pela restrição do domínio de unidades — levou os professores a mudar novamente os critérios de avaliação em 2014.1. Eles decidiram abolir os minitests teóricos e exigiram que os alunos passassem apenas em dois minitests práticos. Mas, diferente de 2013.2, os sujeitos teriam que demonstrar domínio em uma unidade de cada vez. Esta estratégia melhorou bastante o desempenho dos alunos em relação aos períodos anteriores, mas um problema ainda persistia: os estudantes que passavam muitas semanas reforçando as unidades iniciais não tinham a oportunidade de recuperar o ritmo e acompanhar o resto da turma.

O período de 2014.2 marcou o início de uma nova estratégia de avaliação. Como forma de solucionar o problema dos atrasos na disciplina, os professores decidiram que os alunos

poderiam fazer minitests de várias unidades no mesmo dia. Neste esquema, os estudantes ainda precisariam passar em dois minitests de cada unidade, mas teriam 1 hora e meia para fazerem quantos testes conseguissem⁵. Com a mudança na dinâmica das avaliações, os alunos que ficavam presos ao longo da disciplina conseguiam recuperar seu ritmo em apenas uma semana, caso se esforçassem. Como consequência dessa nova estratégia, grande parte dos alunos conseguiu completar a disciplina a tempo e foi aprovada (veremos mais detalhes no capítulo 5).

A forma como as aulas eram conduzidas em Programação I também mudou em 2014.1. Por causa da redução no quadro de professores, a disciplina deixou de ter aulas direcionadas e voltou a funcionar no estilo tradicional, onde o professor ensinava a todos os alunos o mesmo conteúdo. Infelizmente, a escolha de dar aulas no modelo tradicional atrapalhou bastante a vida dos alunos. Em conversas informais, notamos que eles reclamavam muito de estarem estudando os conceitos da unidade 6, por exemplo, mas precisarem fazer minitests da unidade 4, pois ainda não haviam dominado o conteúdo. O modelo de aulas adotado em 2014.1 foi de encontro à filosofia do *Mastery Learning*, pois não oferecia uma aprendizagem personalizada. Era necessário encontrar uma maneira de ensinar, mas, ao mesmo tempo, garantir que o conteúdo fosse relevante à situação atual de cada aluno.

Para solucionar o problema da incompatibilidade das aulas tradicionais, os professores adotaram, em 2014.2, um modelo de sala de aula invertida [9]. Nesta nova abordagem, eles permitiam que os alunos aprendessem novos conceitos por meio de diversos materiais instrucionais e depois reforçassem suas dificuldades em aulas focadas na resolução de problemas. Com a adoção da sala de aula invertida, as aulas deixaram de ter o papel de transmitir novos conceitos e passaram a servir como um momento para o professor auxiliar os alunos a superarem suas dificuldades. Os docentes passaram a utilizar a sala de aula como ambiente de discussão e troca de conhecimento. Os alunos que estavam em unidades adjacentes começaram a participar das mesmas aulas e passaram a interagir para solucionar problemas relacionados às suas dificuldades.

Nesta seção, acompanhamos como foi a evolução do *Mastery Learning* em Programação I, durante os primeiros 2 anos de sua adoção (veja um resumo na Figura 3.2). Nos primeiros

⁵À medida que eles iam passando em uma unidade, eram liberados para fazer mais dois minitests da próxima unidade.

3 períodos, as mudanças aconteceram a partir da experiência dos professores em sala de aula e de suas necessidades para a disciplina. Ao longo deste tempo, realizamos pesquisas com o intuito de avaliar se a mudança de método tinha afetado de alguma forma o comportamento e o desempenho dos alunos. A partir de 2014.2, os professores puderam aprimorar o método com base nos resultados deste trabalho e, como veremos no capítulo 5, alcançaram resultados animadores. Na próxima seção, veremos em detalhes o design da abordagem utilizada em

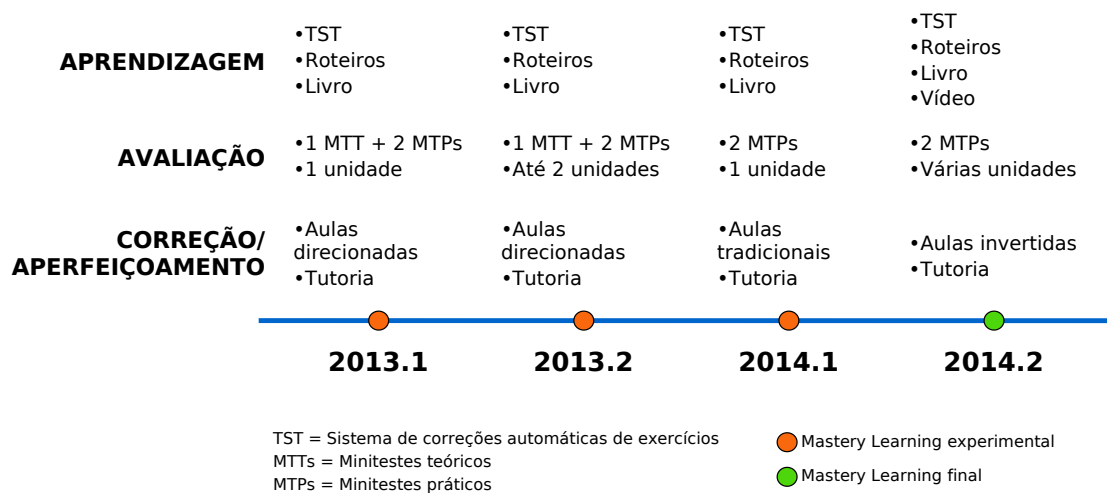


Figura 3.2: Mudanças no método de Programação I, ao longo dos períodos

3.3 Design atual

O *Mastery Learning* de hoje é resultado de uma série de melhorias baseadas na literatura, na experiência dos professores e nas descobertas deste trabalho. Como veremos nos próximos capítulos, a versão atual do método conseguiu proporcionar uma melhora considerável no desempenho da turma e possivelmente tornou sua rotina de estudos ainda mais rica. Nesta seção, entenderemos de uma forma mais estruturada como o método funciona atualmente e quais são seus componentes básicos. Mas, antes, veja o plano de curso da disciplina em 2014.2 no Anexo A.

O *Mastery Learning* implementado na UFCG atua sobre três eixos norteadores: aprendizagem, avaliação e correção/aperfeiçoamento. O primeiro eixo engloba todas as atividades oferecidas aos alunos para promover sua aprendizagem. O segundo contém as estratégias

de avaliação — formativa e somativa — utilizadas para medir o desempenho acadêmico e fornecer *feedback* para professores e alunos. O último eixo é o diferencial desta abordagem. No *Mastery Learning*, os alunos passam por atividades corretivas — caso não dominem o conteúdo — e de aperfeiçoamento — quando dominam. O objetivo deste terceiro eixo é permitir que todos tenham a oportunidade de superar suas dificuldades antes de avançarem na disciplina.

3.3.1 Aprendizagem

O *Mastery Learning* não é uma filosofia voltada ao ensino, mas à aprendizagem [6]. Por conta disso, a disciplina fornece uma gama de recursos educacionais, com o intuito de proporcionar aos alunos uma aprendizagem autônoma e personalizada. Na versão atual do método, os estudantes têm acesso a um banco de questões (integradas ao TST), vídeo-aulas, roteiros e livro, desde a primeira semana de aula. Desta forma, cada um pode aprender através do(s) recurso(s) que mais lhe agrada e não fica preso ao ritmo do professor.

Banco de questões e TST

Um dos diferenciais de Programação I é dispor de um banco de questões e do sistema de correções automáticas, chamado TST. Este ferramental vem sendo desenvolvido desde 2011 pelos professores da disciplina e já possui um número considerável de questões. O sistema é online e contém uma série de problemas para os alunos, organizados por unidades. Assim, os estudantes podem se focar na unidade que estão estudando e praticar a qualquer hora.

Para acertarem um problema, os alunos precisam passar nos testes de caixa preta, disponibilizados na questão, e também nos testes secretos, visíveis apenas pelos professores. Caso os sujeitos não passem em todos os testes, é dado um *feedback* sobre os possíveis erros. O esquema de *feedback* do TST consiste em receber as respostas dos alunos e analisar se há erros de sintaxe ou erros comuns, como divisão por zero ou loop infinito, por exemplo. Uma vantagem de utilizar um sistema de correções automáticas como o TST é proporcionar aos alunos a oportunidade de praticar e ter uma resposta imediata sobre o que deu errado.

Vídeo-aulas

As vídeo-aulas foram introduzidas na disciplina com o intuito de dar uma visão geral sobre cada conceito, explicar a resolução de determinados problemas por meio de exercícios resolvidos e direcionar os alunos para outros materiais instrucionais. Os vídeos são ferramentas importantes, pois os alunos têm a possibilidade de avançar e retroceder a execução várias vezes, até compreender o que o professor quis dizer. Isso não era possível nas aulas expositivas tradicionais, onde o professor ia transmitindo o conteúdo e dificilmente era interrompido por um sujeito que não havia entendido.

Roteiros e livro

Os roteiros são materiais resumidos sobre os conteúdos de cada unidade. Eles proporcionam uma leitura simples e são ideais para um primeiro contato com os novos conceitos, já que trazem vários exemplos e exercícios. Os alunos também têm acesso ao livro-texto da disciplina, que tem o intuito de aprofundar os conceitos apresentados nos roteiros e servir de referência para eventuais dúvidas.

Outros

Os alunos ainda são incentivados a estudar por outros recursos educacionais como tutores online, documentação e sites especializados. Estes materiais instrucionais secundários geralmente são citados durante as aulas ou vídeo-aulas.

3.3.2 Avaliação

Critério de domínio

Na dinâmica atual de avaliação, os alunos precisam atingir o critério de domínio em cada unidade para que possam continuar avançando na disciplina. Este critério consiste em resolver corretamente⁶, pelo menos, dois problemas relacionados aos conceitos da unidade que está sendo avaliada. Esse critério é uma forma de garantir aos professores que os alunos só podem avançar de unidade quando realmente aprenderem o conteúdo.

⁶Passar em todos os testes automáticos.

Minitestes semanais

No *Mastery Learning*, a avaliação é contínua e segue o ritmo dos alunos. A cada semana, os estudantes são submetidos a minitests que avaliam seu nível de domínio na unidade que eles estão estudando atualmente. Caso eles consigam resolver corretamente os dois problemas de uma unidade em menos de 1 hora e meia, são liberados mais dois problemas da próxima unidade. A dinâmica atual de minitests permite aos alunos que precisam reforçar os conceitos de uma unidade ter tempo de recuperar o ritmo na próxima semana. Esta estratégia beneficia tanto os que se atrasam, quanto aqueles que pretendem terminar a disciplina em menos tempo.

Feedback

Pela característica formativa das avaliações no *Mastery Learning*, após cada miniteste, os alunos recebem uma série de *feedbacks* sobre seu desempenho. Primeiro, ao terminarem a avaliação, eles já sabem se passaram ou não de unidade(s). Segundo, os professores analisam seu código durante a semana e lhes informam quais os principais erros e como melhorar a qualidade do código e a lógica utilizada. Terceiro, eles têm reuniões com tutores para tirar dúvidas sobre os conceitos que ainda possuem dificuldades.

Projeto e Prova

Os alunos que conseguem terminar a disciplina antes do final do semestre têm a oportunidade de aprimorar seus conhecimentos por meio de um projeto, que aborda tanto conceitos vistos em Programação I, quanto assuntos de outras disciplinas (e.g. orientação a objetos). O projeto faz parte da avaliação, mas só é liberado para os estudantes que conseguem terminar as unidades até determinado prazo. Aqueles que não conseguem se inscrever para o projeto complementam sua nota por meio de uma prova, que testa os objetivos de aprendizagem de todas as unidades.

3.3.3 Correção e Aperfeiçoamento

Nas disciplinas baseadas no *Mastery Learning*, a avaliação não é a última fase do processo de ensino-aprendizagem. Na realidade, é a partir da avaliação formativa que os alunos recebem

reforço, de acordo com suas necessidades. Os alunos com dificuldades recebem apoio em atividades de correção, já aqueles que conseguem dominar o conteúdo, o reforçam por meio das estratégias de aperfeiçoamento.

As estratégias de correção neste método são compostas por aulas invertidas, tutoria e recursos educacionais. Já o aperfeiçoamento acontece por meio de atividades extracurriculares (e. g. treinamento para olimpíadas de programação) ou ao concluir todas as unidades, através do projeto que foi citado anteriormente.

Aulas invertidas

As aulas invertidas têm um papel fundamental nesta disciplina. Elas não são iguais a aulas tradicionais, onde o professor ministra um mesmo conteúdo para todos. Nesta modalidade, os alunos já precisam ter estudado o conteúdo da unidade e dedicam o tempo da aula para resolver problemas em conjunto com o professor e os outros colegas.

Como a disciplina de Programação I na UFCG possui 3 professores, a turma é dividida e os alunos que estão em unidades próximas participam de aulas com problemas relacionados a suas unidades. Diferente das aulas tradicionais, na sala de aula invertida, os alunos são ativos no processo. Às vezes eles resolvem problemas, em outras o professor se foca em ensinar sobre determinado conceito no qual parte da turma está com dificuldade. Portanto, as aulas são dinâmicas e imprevisíveis.

Tutoria

Na tutoria, os alunos têm um suporte mais individualizado, já que o tutor fica responsável por um número bastante reduzido de sujeitos. Além disso, como o tutor acompanha os estudantes durante todo o semestre, é mais fácil para ele identificar a causa das dificuldades de cada um.

Em Programação I, o tutor e os alunos se reúnem semanalmente para resolver problemas juntos e discutir sobre as dificuldades que eles podem estar enfrentando. Além disso, o tutor fica responsável por ir às aulas de laboratório e ajudar seus alunos, caso seja necessário.

Recursos educacionais

Além de obter auxílio dos professores e tutores, os alunos têm a sua disposição todos os materiais instrucionais da disciplina. Como complemento, eles também podem procurar por materiais na Internet e contar com a ajuda dos seus colegas de classe.

3.4 Conclusões

Neste capítulo, acompanhamos a evolução do *Mastery Learning* em Programação I, desde a sua adoção na UFCG. Aprendemos que, durante os primeiros 3 períodos, o método foi sofrendo alterações a partir das necessidades dos professores, mas, em 2014.2, houve uma mudança maior, resultante das pesquisas deste trabalho. A versão final do *Mastery Learning* permitiu que os alunos conseguissem aprender em seu próprio ritmo e melhorou de forma substancial o desempenho deles, superando as expectativas dos professores.

Apesar de muito já ter sido feito, o método ainda precisa ser aperfeiçoado em alguns aspectos. O esquema de tutoria, por exemplo, ainda não está totalmente integrado à dinâmica da disciplina. Os tutores ainda não têm acesso aos resultados dos minitests e acabam sem saber quais as dificuldades que seus alunos possuem durante a semana, até falar com eles. Outro ponto que precisa melhorar é o *feedback* dos professores para os alunos a respeito da qualidade do código. É preciso corrigir as submissões de forma manual e dar *feedback* personalizado a cada estudante, o que torna o processo lento. Acreditamos que estes dois problemas poderiam ser amenizados se os tutores tivessem acesso aos códigos de seus alunos e pudessem fazer uma primeira verificação, antes da correção dos professores.

Nos próximos capítulos, acompanharemos a análise da versão experimental⁷ do *Mastery Learning* e entenderemos como o método influenciou o comportamento e desempenho dos alunos, em relação à abordagem anterior. Além disso, veremos como a implementação final do método melhorou o desempenho da turma — comparado às versões experimentais — e como a abordagem foi aceita pelos estudantes de Programação I. Por fim, entenderemos as vantagens e desvantagens de utilizá-lo e quais as lições aprendidas pelos professores a partir de toda esta experiência.

⁷O *Mastery Learning* foi considerado experimental entre 2013.1 e 2014.1.

Capítulo 4

Os efeitos do Mastery Learning no comportamento acadêmico

Neste capítulo, investigamos se os alunos do *Mastery Learning* adotaram uma rotina de estudos voltada para o domínio de unidades — se focando em menos conteúdos e reforçando os conceitos até aprenderem — e comparamos se a mudança de método influenciou a cobertura dos conteúdos da disciplina. A partir dessas motivações, nos guiaremos pelas seguintes questões de pesquisa:

- (Q_1) A dispersão de estudo das turmas de Programação I mudou após a adoção do *Mastery Learning*?
- (Q_2) A rotina de estudo dos alunos de Programação I mudou após a implantação do *Mastery Learning*?
- (Q_3) A mudança de método afetou a cobertura do conteúdo de Programação I na UFCG?

4.1 Metodologia

Neste trabalho, foi realizada uma pesquisa quantitativa do tipo *ex post facto* para explorar a rotina de estudo dos alunos a partir dos seus dados históricos de notas e submissões de exercícios.

4.1.1 Dataset

Para este trabalho, foram coletados dados referentes a 5 semestres dos calouros de Programação I, do curso de Ciência da Computação, da Universidade Federal de Campina Grande. Nos primeiros 2 semestres foi utilizado o método tradicional e nos 3 últimos, o *Mastery Learning*. Os dados são formados pelas notas coletadas dos históricos escolares dos alunos e dos exercícios submetidos pelos sujeitos ao TST¹. Foram desconsiderados os estudantes que não alcançaram os critérios mínimos de atividade ao longo do período ou que submeteram uma quantidade extremamente alta de questões. Veja os critérios de exclusão a seguir:

- Ter submetido menos de 15 questões, o que significa aproximadamente uma questão por semana.
- Ter submetido questões em apenas uma semana, o que caracteriza desistência da disciplina.
- Ter mais de 500 submissões, pois foi considerado um *outlier*.

No total, foram considerados 258 alunos: 96 das turmas com o método tradicional e 162 que passaram pelo *Mastery Learning*. Foi mantida a mesma proporção de aprovados (73%) e reprovados (27%) em ambos os métodos por meio da exclusão aleatória dos sujeitos. A média de alunos considerados por período no método tradicional foi de 48 e de 54 no *Mastery Learning*.

4.1.2 Métricas

Algumas métricas foram criadas para representar o comportamento dos alunos na disciplina. A *dispersão de estudos* mede o quanto cada aluno é focado² em seus estudos semanais. Já a *rotina de estudos* mede a quantidade de avanços, revisões e retrocessos dos alunos na disciplina. Por fim, a *cobertura do conteúdo* mede o quanto os alunos praticaram cada unidade.

¹O TST — sistema de correções automáticas da disciplina — coleta dados das submissões como o horário, a unidade, o código, etc.

²Consideramos que um aluno é focado quando estuda um intervalo pequeno de unidades por semana.

Dispersão de estudos

Essa métrica serve para informar se o aluno praticou um intervalo grande ou pequeno de unidades por semana. Para calcularmos a dispersão de estudos de um aluno em determinada semana, identificamos a unidade menor ($\min(U_s)$) que foi praticada e a subtraímos da unidade maior ($\max(U_s)$). Para sabermos a dispersão média de um aluno na disciplina, calculamos a média aritmética das dispersões por semana. Veja a fórmula da dispersão³ a seguir.

$$\text{dispersão} = \frac{\sum_{s=0}^n (\max(U_s) - \min(U_s))}{n}$$

Para entender melhor essa métrica, acompanhe na Figura 4.1 um exemplo de como é calculada a dispersão de um aluno. Na semana $s = 0$ o sujeito praticou o conjunto de unidades $U_0 = \{1, 2, 3\}$, logo a dispersão dele nesta semana foi $3 - 1 = 2$. Na semana seguinte ($s = 1$), ele submeteu questões das unidades $U_1 = \{3, 4\}$, logo teve uma dispersão de $4 - 3 = 1$. Por fim, na última semana ($s = 3$), o sujeito submeteu questões das unidades $U_3 = \{3, 5\}$ e apresentou uma dispersão de $5 - 3 = 2$. Portanto, a dispersão média deste aluno (ou simplesmente dispersão) foi de $(2 + 1 + 2)/3 \simeq 1,7$.

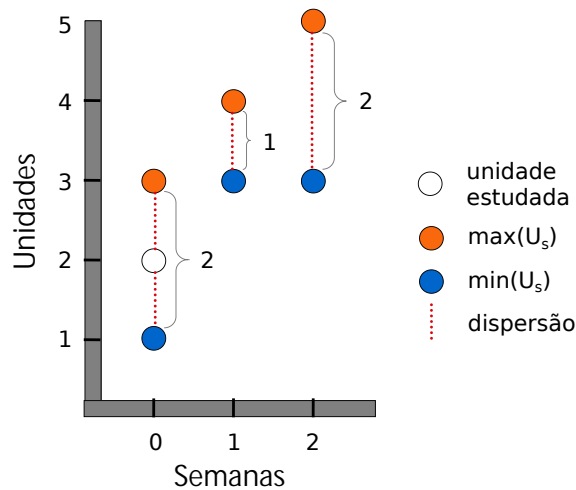


Figura 4.1: Exemplo de dispersão de estudo

Quanto mais próxima a zero for a dispersão de um aluno, mais focado ele terá sido, pois estudou um intervalo curto de unidades por semana. Por outro lado, um sujeito com uma

³Daqui para frente, quando falarmos de dispersão, estamos nos referindo à dispersão média.

dispersão alta é considerado disperso (ou sem foco), pois estudou um intervalo de unidades distantes a cada semana. Um estudante extremamente focado — que estudou apenas um conceito semanalmente — terá uma dispersão igual a 0. Por outro lado, um aluno totalmente sem foco — que pode ter praticado os conteúdos das 10 unidades por semana — terá uma dispersão de 9. Neste trabalho, as dispersões médias variaram entre 0,1 e 4,4. Portanto, encontramos alunos extremamente focados, mas não houve casos de sujeitos totalmente dispersos.

Rotina de estudos

Essa métrica ajuda a entender quanto tempo os alunos se dedicaram, ao longo da disciplina, a três estados de estudo: avanços, reforços e retrocessos. Um estudante avança quando passa a se focar em uma unidade superior à que estava praticando na semana anterior. Ele reforça, quando continua se focando na mesma unidade por duas semanas seguidas. E ele retrocede quando volta a estudar uma unidade inferior.

Um aluno pode praticar vários assuntos em uma semana. Para identificarmos o conteúdo no qual ele mais se focou, estabelecemos o conceito de unidade-foco ($u(s)$), que representa a unidade na qual o sujeito submeteu o maior número de questões ao longo de uma semana. Nós a calculamos por meio da moda⁴ das unidades praticadas na semana, representada pela fórmula $u(s) = \text{moda}(U_s)$. Veja na Figura 4.2 uma ilustração dos conceitos que foram apresentados.

Consideramos que um aluno avança toda vez que ele pratica na semana s uma unidade-foco que é superior àquela estudada em $s - 1$. Se observarmos o exemplo da Figura 4.2, percebemos que o sujeito avançou em seus estudos entre as semanas 0 e 1, pois se focou em exercícios da unidade 2 na primeira semana e passou a praticar a unidade 3 na próxima. Para calcularmos a frequência de avanços de um aluno na disciplina, utilizamos a fórmula a seguir.

$$\text{avanço} = \sum_{s=1}^n \begin{cases} 1, \text{ se } u(s) - u(s-1) > 0 \\ 0, \text{ caso contrário} \end{cases}$$

Dizemos que um indivíduo reforça toda vez que passa duas semanas seguidas se focando

⁴Se houver mais de uma moda, é considerada a maior unidade.

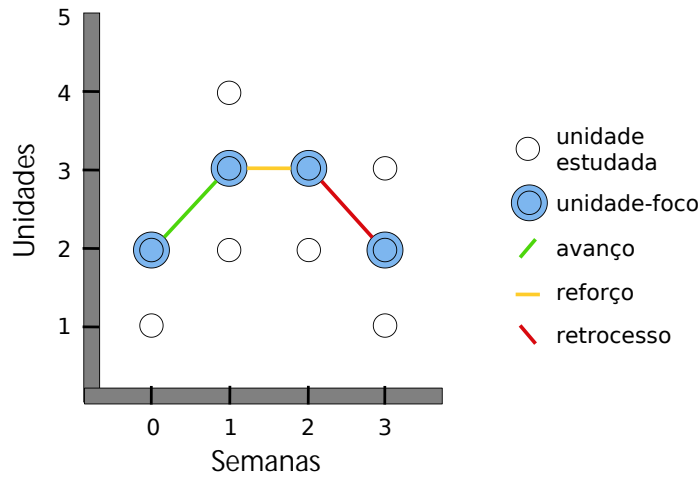


Figura 4.2: Exemplo de rotina de estudos

na mesma unidade. Voltando ao exemplo da Figura 4.2, percebemos que o aluno reforçou os conteúdos da unidade 3 entre as semanas 1 e 2. É importante destacar que, mesmo ele tendo estudado para outras unidades durante as duas semanas, o que levamos em consideração ao calcularmos a rotina é a unidade na qual ele submeteu a maior quantidade de exercícios em cada semana, isto é, a unidade-foco. Para calcularmos a frequência de reforços, utilizamos a fórmula a seguir.

$$\text{reforço} = \sum_{s=1}^n \begin{cases} 1, \text{ se } u(s) - u(s-1) = 0 \\ 0, \text{ caso contrário} \end{cases}$$

Por fim, consideramos que um aluno retrocede toda vez que pratica na semana s uma unidade inferior àquela estudada em $s-1$. Ainda na Figura 4.2, percebemos que a maioria das questões submetidas em $s=2$ pertencem à unidade 3. Já na semana 3, o aluno se focou em praticar exercícios da unidade 2, ou seja, retrocedeu. Para calcularmos a frequência de retrocessos de um aluno na disciplina, utilizamos a fórmula a seguir.

$$\text{retrocesso} = \sum_{s=1}^n \begin{cases} 1, \text{ se } u(s) - u(s-1) < 0 \\ 0, \text{ caso contrário} \end{cases}$$

Portanto, a rotina de estudos de um aluno na disciplina é representada pela frequência de avanços, reforços e retrocessos que ele realizou ao longo do semestre. Para analisarmos a ro-

tina entre métodos, transformamos as frequências de cada aluno em porcentagens, somando a quantidade de avanços, reforços e retrocessos e dividindo pelo total.

Cobertura do conteúdo

Esta métrica representa o quanto os alunos praticaram cada unidade. O percentual de cobertura do conteúdo de uma unidade é obtido pela quantidade de submissões da unidade, dividida pelo total de todas as submissões. Desta forma, a soma dos percentuais de cobertura de todas as unidades é igual a 100%.

Para entender melhor como funciona a cobertura, veja o exemplo de um aluno na Figura 4.3. Neste gráfico, o eixo x corresponde às unidades da disciplina e o eixo y representa o percentual de cobertura de cada unidade. Podemos dizer que a unidade que o aluno mais cobriu foi a 4, pois cerca de 22,5% das submissões de exercícios foram desta unidade. Esse gráfico também mostra que o sujeito cobriu pouco as unidades 7, 8 e 10, pois juntando a cobertura delas só chega a mais ou menos 7,5% das submissões.

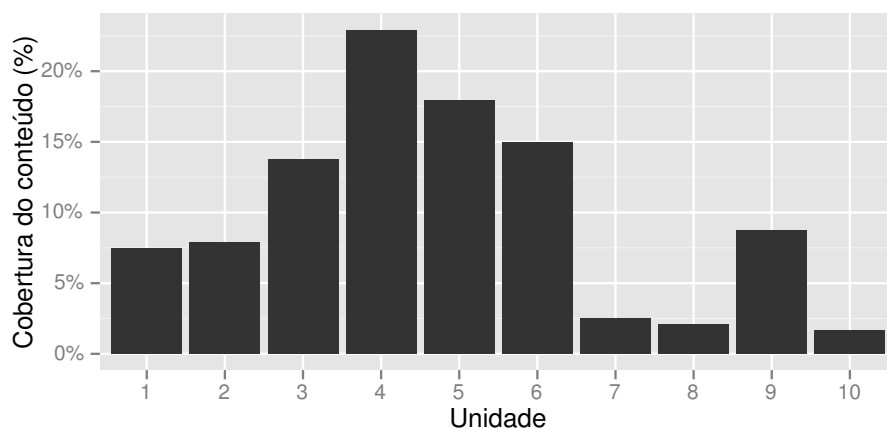


Figura 4.3: Exemplo da cobertura de conteúdos

Portanto, a cobertura do conteúdo serve para medir o quanto os alunos praticaram em cada unidade. A prática, neste contexto, significa a quantidade de submissões feitas pelos alunos durante um semestre. Neste trabalho, nós agrupamos a cobertura por método (i.e. juntamos as coberturas do conteúdo de todos os alunos em cada método) e comparamos se houve alguma mudança na prática de exercícios dos estudantes nestas duas abordagens.

4.1.3 Análise

Primeiramente, os dados foram explorados por meio de estatística descritiva. Foram analisados *boxplots*, gráficos de barras e de densidade e, em seguida, foram investigadas as medianas e o IQR de cada fator. Posteriormente, foram realizados testes de hipótese para verificar se havia diferença estatística entre os valores obtidos no método tradicional e no *Mastery Learning*. Foi aplicado o teste de *Mann-Whitney* para verificar se a dispersão e a rotina dos alunos no método tradicional foram maiores ou menores que no *Mastery Learning* ou se havia diferença entre os aprovados e reprovados em cada abordagem. Além disso, foi utilizado o teste Chi-quadrado para verificar se havia diferença entre a proporção de avanços, reforços e retrocessos em ambos os métodos.

4.2 Dispersão de estudos

A dispersão de um aluno representa a média dos intervalos de unidades que ele estudou a cada semana. Podemos considerar que, quanto maior for a dispersão, menos focado o estudante será (e vice-versa). Ao analisarmos este fator, descobrimos que os estudantes no *Mastery Learning* apresentaram um foco maior nos estudos (dispersão média = 0,9), quando comparados aos sujeitos no método tradicional (dispersão média = 1,8)⁵.

Nós também exploramos a dispersão dos alunos em cada método, separada pela situação final deles na disciplina (veja Figura 4.4). Confirmamos que todos os grupos no *Mastery Learning* apresentaram uma dispersão menor⁶ em relação às turmas do método tradicional. Além disso, descobrimos que os sujeitos reprovados por média no *Mastery Learning* foram os que se focaram, semanalmente, em menos unidades. Nos outros grupos, os alunos apresentaram uma dispersão em torno de 1, o que significa que eles praticaram entre uma e duas unidades por semana.

Para entender melhor a relação entre a evolução do *Mastery Learning* e a dispersão dos alunos, investigamos como este fator mudou ao longo dos períodos (veja na Figura 4.5).

⁵De acordo com o teste de Mann-Whitney ($p\text{-valor} = 2,23 \times 10^{-10}$), a dispersão dos alunos no método tradicional é maior em relação ao *Mastery Learning*.

⁶De acordo com o teste de Mann-Whitney, as hipóteses alternativas de que a dispersão de estudo dos alunos no método tradicional foi maior em relação aos sujeitos no *Mastery Learning* para todos os grupos (p-valores: aprovados por média = 0,0213; aprovados na final = $0,2088 \times 10^{-4}$; reprovados na final = 0,0065; reprovados por média = $0,5859 \times 10^{-6}$).

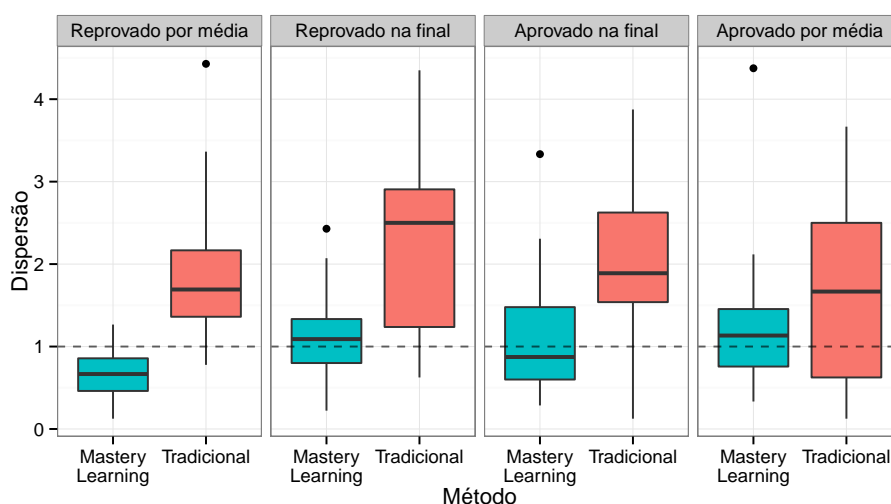


Figura 4.4: Dispersão dos diferentes perfis de alunos em cada método

Constatamos que quando mudamos de método, as turmas ficaram mais focadas, pois os alunos passaram a estudar menos unidades por semana. No entanto, nos períodos seguintes, notamos que a dispersão das turmas voltou a crescer. A maioria dos alunos no último período estudou entre de 2 e 3 unidades por semana.

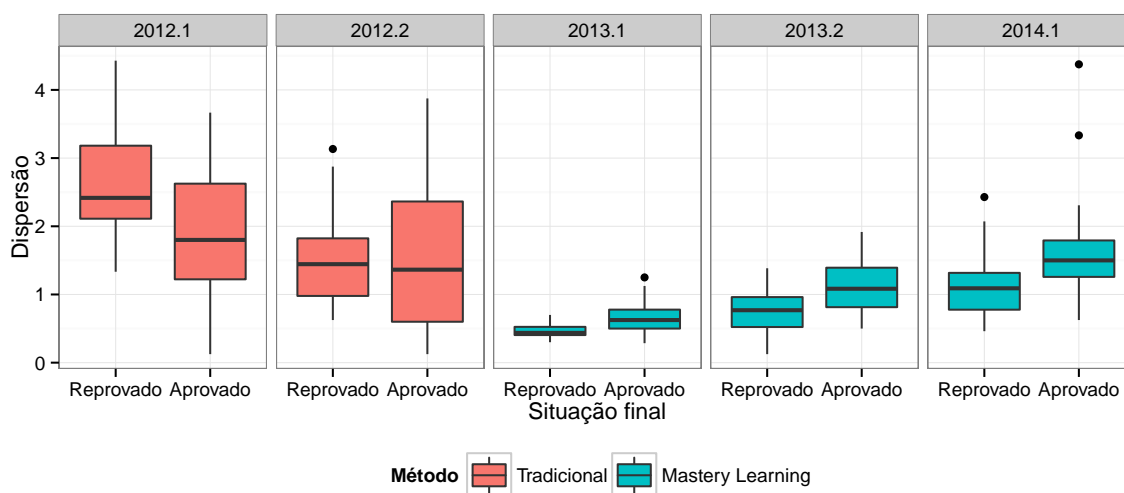


Figura 4.5: Dispersão dos alunos aprovados e reprovados em cada período

Outro fato interessante na Figura 4.5, é que a dispersão dos alunos aprovados e reprovados mudou ao migrarmos para a nova abordagem. No método tradicional, os alunos reprovados tendiam a ser mais dispersos e explorar unidades distantes; já os aprovados, se focavam em menos unidades, quando comparados ao outro grupo. Quando começamos a utilizar o

Mastery Learning, os alunos reprovados diminuíram drasticamente a quantidade de unidades estudadas por semana e se focaram em assuntos mais próximos. Isso possivelmente ocorreu por causa da rotina de estudos e avaliações adotada após o *Mastery Learning*, na qual os estudantes precisavam dominar uma unidade para poder estudar a próxima. Quando analisamos os últimos três períodos, percebemos que os alunos aprovados passaram a explorar mais unidades em relação aos reprovados no mesmo método. Isso pode refletir a possibilidade que eles tiveram de estudar em um ritmo independente, sem precisarem esperar o resto da turma.

4.3 Rotina de estudos

Nesta seção, analisamos como o *Mastery Learning* pode ter influenciado a rotina de estudo das turmas de Programação I. Nosso intuito foi explorar se os alunos de diferentes aptidões adotaram uma rotina que lhes permitisse dominar os conceitos efetivamente. Veja os resultados da rotina de estudo das turmas em ambos os métodos na Figura 4.6.

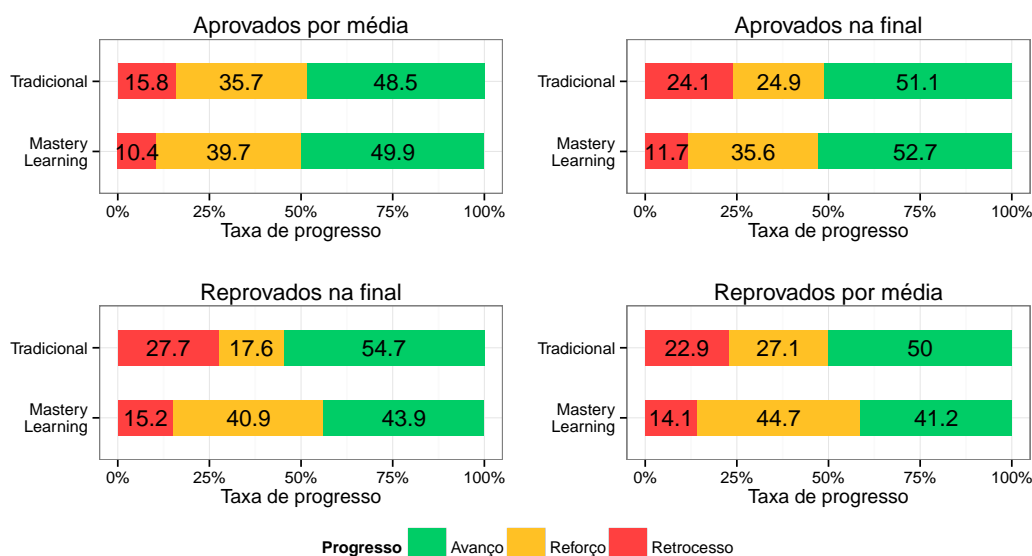


Figura 4.6: Rotina dos alunos em cada método

Como resultado principal deste trabalho, descobrimos que os alunos no *Mastery Learning* adotaram a característica do domínio de unidades e passaram mais tempo reforçando os conteúdos antes de avançarem para os próximos. Como consequência disso, notamos que houve um declínio nos retrocessos de todos os grupos, o que significa que os estudan-

tes passaram menos tempo revisando conceitos que já haviam sido estudados em semanas anteriores. Por fim, constatamos que os alunos aprovados apresentaram taxas de avanço semelhantes nas duas abordagens, mas os reprovados avançaram menos no *Mastery Learning*.

Quando analisamos a rotina de estudo dos alunos aprovados por média no método tradicional, percebemos que eles já passavam grande parte do tempo reforçando os conceitos antes avançarem para um novo assunto. Por conta disso, a rotina desse perfil foi a que menos sofreu alterações ao implantarmos o *Mastery Learning*. Com relação aos aprovados na final, constatamos que eles mantiveram a taxa de avanços, mas passaram mais tempo reforçando os conceitos e retrocederam menos.

De acordo com a Figura 4.6, os alunos reprovados foram o perfil que mais mudou sua rotina de estudos após a transição do método tradicional para o *Mastery Learning*. Antes dos professores mudarem de abordagem, eles mal se dedicavam a reforçar os conceitos antes de avançarem. Quando o *Mastery Learning* passou a ser utilizado, esses estudantes passaram a dedicar cerca de 40% de sua rotina para reforçar as mesmas unidades por mais de uma semana. Isso acabou reduzindo a incidência de retrocessos.

4.4 Cobertura do conteúdo

Nesta seção, investigamos como foi a cobertura do conteúdo entre os diferentes perfis de alunos no *Mastery Learning* e no método tradicional. Além disso, exploramos se as versões iniciais do *Mastery Learning* realmente interferiram na cobertura dos conteúdos — como discutimos no capítulo 3 — e se os professores solucionaram este problema ao aprimorarem os critérios de domínio. Veja na Figura 4.7 como foi a cobertura das unidades pelos alunos aprovados e reprovados em cada método.

No método tradicional, tanto os alunos aprovados, quanto os reprovados fizeram questões de todas as unidades. No *Mastery Learning*, percebemos que, quanto pior a situação final dos alunos na disciplina, menor era sua cobertura nas últimas unidades. Cerca de 95% das submissões feitas pelo reprovados por média eram da unidade 5 ou anteriores. Da mesma forma, entre os alunos reprovados na final, apenas 10% dos exercícios eram de unidades acima de 7.

Nós também exploramos como a cobertura do conteúdo mudou ao longo dos períodos

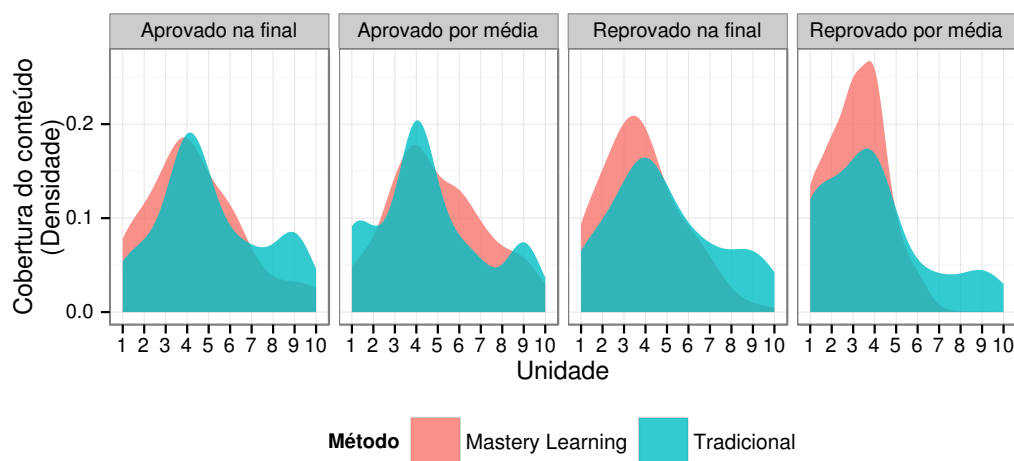


Figura 4.7: Cobertura do conteúdo em cada método, separada pela situação final dos alunos

nos quais o *Mastery Learning* foi aplicado. Veja na Figura 4.8 que ela aumentou, à medida que o método foi sendo aprimorado. Ao analisarmos os 3 períodos onde o *Mastery Learning* foi aplicado, percebemos que os estudantes reprovados por média submeteram poucos exercícios — ou não praticaram — em unidades superiores à 5. O período onde os alunos conseguiram cobrir mais unidades foi 2013.2 (com submissões até a unidade 9). Já 2013.1 foi período onde os alunos cobriram menos conteúdo (unidades até 6).

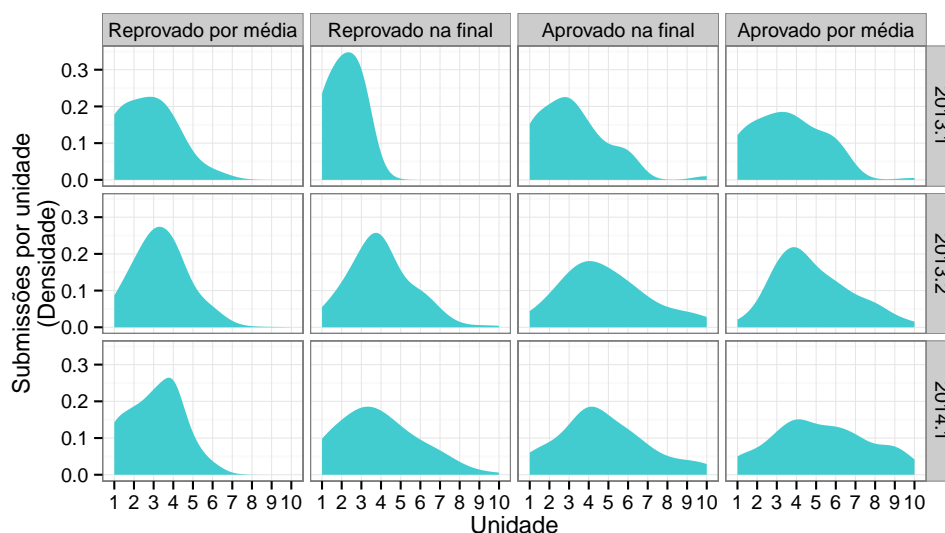


Figura 4.8: Cobertura das unidades pelos alunos aprovados e reprovados em cada período

Entre os alunos no *Mastery Learning* que foram reprovados na final, descobrimos que houve uma melhora significativa na cobertura do conteúdo em 2013.2 e 2014.1 em relação ao

período piloto. No período piloto do *Mastery Learning*, os alunos só responderam exercícios de unidades inferiores à 5; já nos semestres seguintes, os estudantes submeteram exercícios de todas as unidades, embora 3/4 dessas submissões sejam de unidades menores que 6.

Com relação aos alunos aprovados na final no *Mastery Learning*, percebemos que as turmas de 2013.2 e 2014.1 praticaram mais as unidades acima de 6 (32% das submissões) quando comparadas a 2013.1 (2% das submissões). Por fim, entre os alunos aprovados por média, descobrimos que houve uma melhora progressiva na cobertura dos conteúdos a cada semestre. Em 2013.1, apenas 1% das submissões eram de unidades acima de 6; em 2013.2, essa porcentagem subiu para 20% e em 2014.1 para 31%.

4.5 Discussões

Nesta seção, discutiremos em mais detalhes como o comportamento dos diferentes perfis de alunos foi afetado ao migrarmos do método tradicional para o *Mastery Learning*. Temos a convicção de que os alunos conseguiram ter uma rotina de estudos mais rica e focada no domínio do conteúdo após a nova abordagem.

4.5.1 Aprovados por média

Os alunos aprovados por média foram os sujeitos que apresentaram menos diferenças no comportamento ao migrarmos do método tradicional para o *Mastery Learning*. Em ambas as abordagens, percebemos que eles dedicaram boa parte de sua rotina para reforçar os conteúdos e não passaram muito tempo revisando unidades anteriores. O que supomos é que a rotina de estudo desses sujeitos já é intrinsecamente organizada e voltada ao domínio do conteúdo, não importando o método de ensino que seja utilizado.

Apesar do comportamento dos aprovados por média ser bastante semelhante nas duas abordagens, existe uma diferença que precisa ser destacada. No método tradicional, eles se focaram em resolver exercícios das unidades básicas. Já no *Mastery Learning*, os estudantes cobriram o conteúdo da disciplina de uma forma mais homogênea. Uma possível explicação para isso é que, na primeira abordagem, os professores dedicavam mais tempo aos conceitos que causavam problemas à maioria da turma. Como todos seguiam o ritmo do professor, era necessário que alguns ficassem esperando, até que os docentes avançassem para um novo

assunto. Já no *Mastery Learning*, os alunos que dominavam determinado conteúdo podiam passar para o próximo, sem precisar esperar o resto da turma. Assim, eles conseguiram praticar todas as unidades de uma forma mais homogênea.

★ Importante

O *Mastery Learning* permitiu que os alunos aprovados por média pudessem progredir em seu próprio ritmo, sem depender do resto da turma.

4.5.2 Aprovados na final

Como já falamos em outras ocasiões, a turma de programação possui alunos com os mais variados ritmos de aprendizagem. Por conta disso, alguns deles, mesmo se esforçando, precisam de mais tempo para dominar o conteúdo. Em um método que segue o ritmo do professor, esses alunos acabam levando dúvidas adiante e têm que conciliar entre revisar os conceitos anteriores e estudar para o assunto que o professor está ministrando. Foram exatamente essas características que identificamos nos alunos aprovados na final que passaram pelo método tradicional. Eles foram mais dispersos e dividiram sua rotina entre revisar os conceitos anteriores e estudar para o conteúdo atual. Estes alunos passaram na disciplina, mas provavelmente tiveram uma aprendizagem superficial e possivelmente apresentarão dúvidas no futuro, ao precisarem programar [27].

Quando analisamos o mesmo perfil de alunos no *Mastery Learning*, descobrimos que os sujeitos apresentaram uma rotina de estudos muito mais focada e eficiente. Primeiro, como não precisavam acompanhar o ritmo do professor, eles praticaram menos unidades por semana, pois só tinham que se focar em superar suas dificuldades na unidade que estavam estudando; segundo, começaram a dedicar grande parte do tempo para revisar os conteúdos antes de avançarem, o que provavelmente permitiu que eles dominassem todos os conceitos e não levassem dúvidas adiante.

Mas se os alunos que foram para a final no *Mastery Learning* apresentaram um comportamento parecido com aqueles aprovados por média, por que eles não passaram direto, como seus colegas? Diferente do método tradicional, onde os estudantes cobriram o conteúdo, mas levaram dúvidas adiante, os alunos da segunda abordagem precisaram dominar cada unidade para conseguirem progredir na disciplina. Por causa do prazo do semestre, eles não tive-

ram tempo de passar nas unidades mais avançadas (mas possivelmente as estudaram) e não atingiram a nota necessária para serem aprovados por média.

Existe uma diferença considerável entre as duas situações mostradas nesta seção: no primeiro método, os alunos passaram na disciplina, mas possivelmente tiveram uma aprendizagem superficial, pois não conseguiram acompanhar o ritmo do professor e levaram suas dúvidas adiante; na segunda abordagem, os alunos não conseguiram passar nas últimas unidades, mas dominaram todo o conteúdo básico da disciplina. Portanto, acreditamos que os sujeitos que passaram pelo *Mastery Learning* aprenderam efetivamente os conceitos de Programação I, embora tenham precisado ir para a final.

4.5.3 Reprovados na final

O grupo dos alunos reprovados na final foi o que apresentou a maior diferença de comportamento ao mudarmos de método. Na abordagem tradicional, eles estudaram de uma maneira bastante dispersa. Além disso, passaram a maior parte do tempo em um vai-e-vem: avançavam e retrocediam. Isso provavelmente aconteceu porque eles sentiram dificuldades, mas não souberam como lidar com esta situação. Eles tinham que conciliar suas dificuldades nos conceitos iniciais e estudar para o assunto que o professor estava abordando.

No *Mastery Learning*, os alunos se focaram mais no que precisavam aprender e passaram grande parte do tempo revisando os conceitos que sentiam dificuldade. Uma consequência disso foi que a proporção de retrocessos caiu quase pela metade em relação ao método anterior. A dinâmica imposta pelo *Mastery Learning* nitidamente influenciou para que estes alunos tivessem uma rotina de estudos mais organizada e eficiente. No entanto, eles não conseguiram ser aprovados por não possuírem ritmo e/ou perseverança suficientes para avançarem o mínimo de unidades antes do final do período.

★ Importante

O *Mastery Learning* melhorou a rotina de estudo dos alunos reprovados, tornando-os mais focados e organizados.

4.5.4 Reprovados por média

Os reprovados por média no *Mastery Learning* apresentaram um comportamento bastante parecido com os reprovados na final. A diferença entre eles foi que os primeiros ficaram tão presos em dominar os conteúdos que possuíam dificuldades, que não conseguiram cobrir nem dois terços do currículo da disciplina. Esse resultado levanta novamente a questão: vale a pena abrir mão da cobertura do conteúdo para permitir que os alunos passem mais tempo superando suas dificuldades e dominando os conceitos básicos da disciplina? Nós achamos que sim.

Apesar dos alunos reprovados no *Mastery Learning* não terem tido a oportunidade de cobrir uma parcela considerável do conteúdo de Programação I — por passarem muito tempo dominando os conceitos das primeiras unidades — acreditamos que isso não é tão crítico. Eles terão a chance de estudar novamente a disciplina no próximo período e possivelmente avançarão mais rápido nas unidades que dominaram no semestre anterior⁷. Se esses mesmos alunos tivessem sido reprovados no método tradicional, eles possivelmente enfrentariam as dificuldades que expomos no capítulo 1 e teriam uma chance maior de não aprender a programar. Portanto, acreditamos que o comportamento dos alunos reprovados no *Mastery Learning* vai permitir que eles voltem mais preparados para a disciplina em relação àqueles que passaram pelo método tradicional.

4.6 Conclusões

Na primeira questão de pesquisa, questionamos se a dispersão de estudo das turmas tinha mudado quando os professores migraram do método tradicional para o *Mastery Learning*. Descobrimos que, após a mudança de abordagem, os alunos foram mais focados em relação àqueles que cursaram a disciplina baseada no método tradicional. Antes do *Mastery Learning*, os sujeitos estudavam um intervalo maior de unidades por semana. Depois da adoção dessa nova abordagem, percebemos que os alunos praticaram um intervalo menor de unidades semanalmente. Esse resultado pode estar relacionado ao fato de que, no último método, os estudantes foram incentivados a se focarem na unidade atual, pois não podiam avançar antes de dominá-la.

⁷Como foi explicado por Gentile e Lalley [27].

Na segunda questão de pesquisa, nos interessamos em saber se a rotina de estudo dos sujeitos era diferente nos dois métodos. Descobrimos que os estudantes realmente apresentaram rotinas distintas. No método tradicional, os alunos — principalmente os aprovados na final ou reprovados — reforçavam por pouco tempo os conteúdos antes de avançar e dedicavam cerca de um quarto da sua rotina de estudos para revisar conteúdos que já haviam estudado anteriormente. Já os alunos no *Mastery Learning*, passaram mais tempo se focando nos mesmos conteúdos e revisaram menos os conceitos anteriores. Estes resultados são importantes, pois dão indícios que os alunos passaram mais tempo dominando cada conceito e, possivelmente, levaram menos dúvidas adiante.

Na terceira questão de pesquisa, investigamos se a adoção do *Mastery Learning* influenciou a cobertura do conteúdo da disciplina. Descobrimos que houve mudanças no comportamento dos alunos entre as duas abordagens. No método tradicional, os sujeitos de diferentes perfis praticaram conteúdos de todas as unidades, embora tenham dado uma atenção maior às primeiras. Por outro lado, no *Mastery Learning*, percebemos que apenas os alunos aprovados estudaram todas as unidades. Os reprovados se focaram em dominar os primeiros conteúdos, mas não tiveram tempo de cobrir todos os conceitos da disciplina. Portanto, os sujeitos não foram reprovados no *Mastery Learning* por falta de domínio, mas porque não conseguiram cobrir o conteúdo a tempo.

Deixamos claro que esta pesquisa não possui validade externa, pois os resultados aqui obtidos são limitados à utilização do *Mastery Learning* no contexto de Programação I na UFCG. Como trabalho futuro, esperamos que estes resultados possam ser replicados em outras instituições, para que seja garantida a generalização da eficácia do *Mastery Learning* nas disciplinas de programação introdutória.

Durante as discussões deste capítulo, levantamos algumas suposições que, por questão de tempo, não puderam ser avaliadas. Esperamos que, no futuro, seja investigada a relação entre os alunos estudarem em uma disciplina baseada no *Mastery Learning* e eles conseguirem reaprender mais rapidamente os conceitos iniciais de programação, caso sejam reprovados ou precisem utilizá-los em uma disciplina correlata.

Apesar dos problemas enfrentados nas primeiras implementações do *Mastery Learning* em Programação I, os professores estão satisfeitos e confiantes com os efeitos provocados pelo método no comportamento dos alunos. Por meio desta abordagem, os estudantes com

diferentes ritmos de aprendizagem tiveram a oportunidade de aprender. Os resultados alcançados neste capítulo são animadores e nos levam a crer que a nova abordagem proporcionou a todos um processo de ensino-aprendizagem mais rico e personalizado.

Capítulo 5

O Mastery Learning e o desempenho acadêmico

Quando falamos em avaliação de métodos educacionais, o primeiro critério que vem à nossa mente é o desempenho. Neste capítulo, exploramos os efeitos do *Mastery Learning* no desempenho das turmas de Programação I a partir de duas perspectivas: (1) *entre métodos*, investigando as taxas de aprovação antes e depois da adoção da abordagem; e (2) *no método*, analisando as taxas de aprovação e o desempenho das turmas em cada período do *Mastery Learning*.

O objetivo desta pesquisa foi identificar se houve mudanças nas taxas de aprovação após a adoção do novo método e verificar como as mudanças feitas no *Mastery Learning* a cada período podem ter afetado o desempenho das turmas. Ao final, teremos respondido as seguintes questões de pesquisa:

- (Q_1) A adoção do *Mastery Learning* influenciou as taxas de aprovação em Programação I?
- (Q_2) Os aprimoramentos nas versões experimentais do *Mastery Learning* influenciaram o desempenho das turmas de Programação I?
- (Q_3) As mudanças na versão final do *Mastery Learning* melhoraram o desempenho da turma, em relação aos resultados obtidos nas versões experimentais?

5.1 Metodologia

O estudo realizado neste capítulo foi do tipo *ex post facto* [17] e teve como objetivo investigar os efeitos do *Mastery Learning* no desempenho acadêmico e nas taxas de aprovação e reprovação das turmas introdutórias de programação.

5.1.1 Dataset

Foram analisados dados de 12 semestres das disciplinas introdutórias de Programação¹ do curso de Ciência da Computação da Universidade Federal de Campina Grande. Foi utilizado o método de ensino Tradicional² nos 8 primeiros semestres e o *Mastery Learning* nos últimos quatro. Foram considerados os dados de 935 alunos (entre 2009.1 e 2014.2) na análise entre métodos e dados de 438 sujeitos (entre 2013.1 e 2014.2), na análise no método.

Os alunos que cursaram a disciplina na turma de repetentes³ foram removidos do estudo, pois vivenciaram uma abordagem de ensino diferente de ambos os métodos analisados nesta pesquisa. Os alunos desistentes — que se matricularam, mas não cursaram a disciplina — também foram eliminados porque não eram relevantes ao escopo desta pesquisa.

5.1.2 Métricas

Uma das métricas analisadas neste trabalho foi a taxa de aprovados e reprovados em cada período letivo. São aprovados por média os alunos com notas maiores ou iguais a 7,0; os aprovados na final são aqueles que conseguiram obter uma nota final maior ou igual a 5,0; os estudantes reprovados na final obtiveram uma nota final abaixo de 5,0 e os reprovados por média não conseguiram alcançar o valor mínimo (4,0) para realizarem a prova final.

O desempenho acadêmico foi representado pela nota dos minitests práticos nas turmas que utilizaram o *Mastery Learning*. Estes minitests são avaliações realizadas no laboratório, nas quais os alunos precisam criar e implementar soluções para um conjunto de problemas que envolvem a utilização de programação. A nota final não foi escolhida como métrica de desempenho porque ela possui muitos critérios incorporados, o que compromete sua validade

¹Programação 1 e Laboratório de Programação 1.

²Aulas expositivas e no laboratório, além de provas e minitests.

³Os alunos reprovados na disciplina são separados dos alunos novatos no período seguinte e assistem aulas em uma turma especial, com um ritmo mais rápido.

[4].

5.1.3 Análise

Três análises foram realizadas neste trabalho: na primeira, foram investigadas as taxas de aprovação entre os dois métodos de ensino; posteriormente, foram exploradas as taxas de aprovação e reprovação (de uma forma mais detalhada) entre os períodos nos quais o *Mastery Learning* foi aplicado; e, por fim, foi analisado o desempenho das turmas no *Mastery Learning*.

A primeira análise consistiu na normalização das taxas de aprovação das turmas entre 2009.1 e 2014.2. Foi utilizada a normalização *standard score* [2] para explorar o desvio-padrão entre as médias de aprovação no método tradicional e no *Mastery Learning*. Primeiramente foi realizada uma análise descritiva por meio da distribuição dos valores normalizados e, posteriormente, foi utilizado o teste *t* para verificar se as médias de aprovação nos dois métodos realmente eram iguais. O teste *t* foi utilizado porque os dados foram considerados normais⁴.

Para a análise das taxas de aprovação e reprovação no *Mastery Learning*, foram considerados os dados de 2013.1 a 2014.2. Os dados foram explorados por meio de gráficos de barras e os resultados foram confirmados através do teste Chi-quadrado [45]. Este teste foi utilizado para verificar se a proporção de aprovados e reprovados mudou ao longo dos períodos nos quais o *Mastery Learning* foi aplicado.

Na análise do desempenho acadêmico, foram considerados os dados de 2013.1 a 2014.2. Não houve uma comparação entre o desempenho dos alunos nos dois métodos porque os critérios de avaliação mudaram de uma abordagem para a outra. No método tradicional, a nota dos minitestos práticos era apenas um complemento da nota geral e sua avaliação era mais superficial. Após o *Mastery Learning*, os minitestos práticos passaram a ter uma avaliação mais criteriosa e sua nota tornou-se o principal critério de desempenho. Os dados foram analisados com estatística descritiva e foi aplicado o teste de Mann-Whitney [3] em cada par de períodos para verificar se havia uma diferença significativa entre o desempenho das turmas analisadas. O teste de Mann-Whitney foi escolhido porque as notas não seguiam

⁴De acordo com o teste Shapiro-Wilk, a hipótese nula foi aceita ($p\text{-valor} = 0,8771$) de que a distribuição das taxas de aprovação é normal.

uma distribuição normal [45].

5.2 Efeitos na aprovação e reprovação

As taxas de aprovação/reprovação são métricas utilizadas na literatura para comparar a eficácia de abordagens educacionais. Investigamos, nesta seção, se as taxas de aprovação mudaram entre o método tradicional e o *Mastery Learning* e como essas métricas se comportaram ao longo dos semestres quando a segunda abordagem foi utilizada.

Método tradicional x *Mastery Learning*

Analisamos a distribuição das taxas de aprovação (veja Figura 5.1) entre 2009.1 e 2014.2 e chegamos à conclusão de que tanto no método tradicional, quanto no *Mastery Learning*, a média de aprovados não se alterou significativamente.

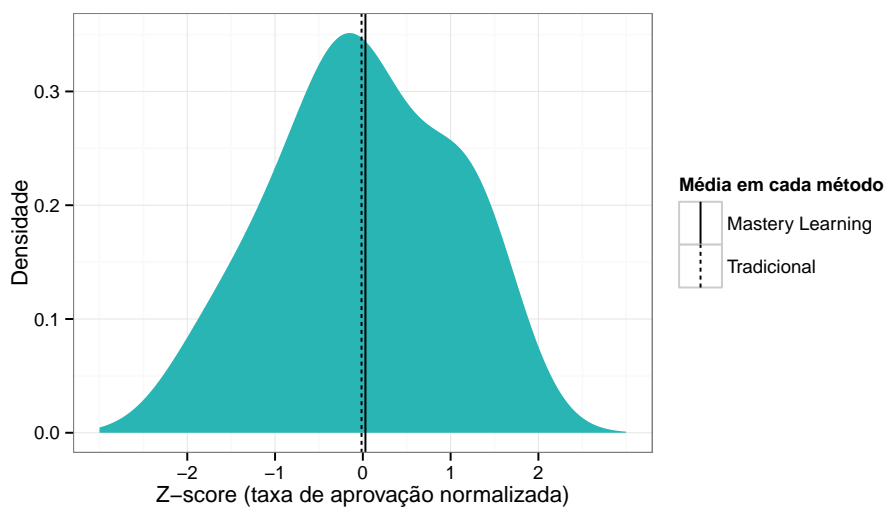


Figura 5.1: Distribuição das taxas de aprovação dos últimos 6 anos em P1

Descobrimos que a média de aprovados no método tradicional e no *Mastery Learning* foi cerca de 73%. Na Figura 5.1 constatamos que o desvio-padrão entre as médias de aprovação nas duas abordagens é próximo de zero, quando comparado à média da população. Para confirmar esta igualdade, aplicamos o teste *t* e aceitamos a hipótese nula ($p\text{-valor} = 0,9472$) de que as taxas de aprovação no método tradicional e no *Mastery Learning* não possuem uma diferença significativa.

Aprovação e reprovação no *Mastery Learning*

Com relação às taxas de aprovados e reprovados em cada período do *Mastery Learning*, conseguimos resultados animadores (veja Figura 5.2). Descobrimos que a proporção de aprovados aumentou significativamente no último período e a taxa de reprovados por média despencou para cerca de 5%. Além disso, constatamos que a taxa de aprovados por média vem aumentando⁵ gradativamente, desde o período piloto.

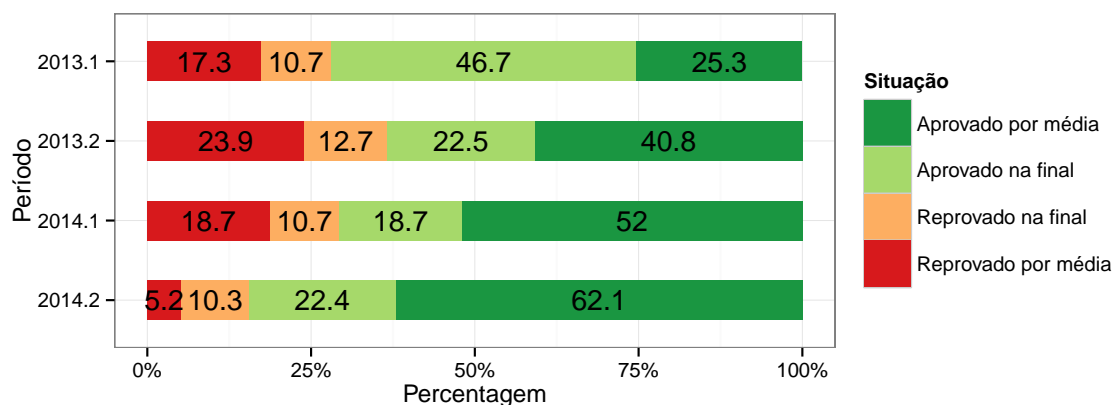


Figura 5.2: Taxas de aprovação e reprovação em cada período do *Mastery Learning*

As taxas de reprovação sempre foram apontadas na literatura como um dos principais problemas nos cursos introdutórios de programação até Watson e Li [54] reunirem evidências que levaram a crer que esses relatos poderiam ser apenas anedóticos. De acordo com seu estudo, os cursos possuem uma taxa média de aprovação de 67%, podendo chegar a aproximadamente 70,1% na população. Se compararmos esses valores com as taxas de aprovação apresentadas nesta seção, constatamos que a taxa de sucesso atuais de Programação I, na UFCG, superaram os resultados encontrados na literatura.

5.3 A evolução do método e o desempenho

Nós vimos no capítulo 3 que o *Mastery Learning* não nasceu pronto em Programação I. Ele evoluiu a partir das experiências dos professores e das pesquisas deste trabalho. Nesta seção, analisamos se estas mudanças influenciaram positivamente o desempenho das turmas e se a

⁵De acordo com o teste Chi-quadrado, a hipótese alternativa foi aceita ($p\text{-valor} = 0,6163 \times 10^{-4}$) de que a proporção de alunos aprovados por média e na final é diferente entre os períodos do *Mastery Learning*.

versão final do método teve resultados melhores, em relação às anteriores.

Quando analisamos as distribuições de notas nos períodos correspondentes ao uso do *Mastery Learning* (veja Figura 5.3), percebemos que 2013.1 foi a turma que apresentou o pior desempenho (apenas 20% dos alunos obtiveram notas superiores ou iguais a 7,0), principalmente entre os alunos aprovados⁶. Em contrapartida, as distribuições de notas nos períodos seguintes parecem seguir uma tendência positiva, na qual o desempenho acadêmico vem melhorando gradativamente. Em 2013.2, cerca de 35% dos alunos apresentaram notas maiores ou iguais a 7,0, sendo que a mediana das notas dos aprovados subiu para 7,0 e entre os reprovados aumentou de 3,3 para 3,7. Em 2014.1, não houve alteração nas medianas das notas dos aprovados e reprovados, mas cerca de metade dos alunos alcançou uma nota de pelo menos 7,0. O semestre de 2014.2 — onde foi utilizada a versão final do método — foi o que apresentou o melhor desempenho depois da implantação do *Mastery Learning* e cerca de 73% dos alunos conseguiram obter notas iguais ou maiores que 7,0.



Figura 5.3: Distribuição da nota dos minitests práticos a cada período

Ao realizarmos testes de *Mann-Whitney* com as notas de cada par de períodos (*e.g.* 2013.1 e 2013.2), constatamos que não houve uma diferença significativa entre as transi-

⁶A mediana das notas dos aprovados em 2013.1, 2013.2 e 2014.1 foi, respectivamente, igual a 5,6; 7,0 e 7,1.

ções de 2013.1 para 2013.2⁷ e deste semestre para 2014.1⁸. No entanto, ao compararmos o desempenho dos alunos entre 2013.1 e 2014.1⁹ ou 2014.2 e qualquer outro período¹⁰, obtemos resultados estatisticamente significativos.

★ Importante

A versão final do *Mastery Learning* provocou uma melhora considerável no desempenho de toda a turma, em relação às versões experimentais. Isso dá indícios que as estratégias adotadas em 2014.1 foram um sucesso.

5.4 Discussões

Quando os professores começaram a utilizar o *Mastery Learning* em 2013, pouco sabiam sobre esta filosofia e se basearam principalmente no trabalho *Good-bye teacher...*, de Keller, e em seu *Personalized System of Instruction*. A meta deles, ao adotarem esta nova abordagem, foi proporcionar a todos os estudantes uma aprendizagem efetiva e assegurar que eles não levassem dúvidas adiante.

Descobrimos neste capítulo que as versões iniciais do *Mastery Learning* proporcionaram resultados contrários aos que esperávamos, mas, à medida que o método foi sendo aprimorado, obtivemos resultados surpreendentes. Nós acreditamos que as mudanças adotadas a cada semestre afetaram diretamente o desempenho dos alunos na disciplina. Nesta seção, discutiremos os possíveis motivos que provocaram o baixo desempenho das primeiras turmas e quais as precauções que devemos tomar para evitarmos tais problemas.

5.4.1 Domínio x cobertura

Quando os professores começaram a utilizar o *Mastery Learning*, eles se focaram em garantir o domínio de unidades (i.e. todos os alunos só progredissem na disciplina depois que

⁷A hipótese nula foi aceita ($p\text{-valor} = 0,7229$) de que não há diferença entre o desempenho das turmas em 2013.1 e 2013.2

⁸A hipótese nula foi aceita ($p\text{-valor} = 0,4117$) de que não há diferença entre o desempenho das turmas em 2013.2 e 2014.1

⁹A alternativa foi aceita ($p\text{-valor} = 0,0116$) de que o desempenho dos alunos em 2014.1 foi maior que em 2013.1

¹⁰As hipóteses alternativas foram aceitas de que o desempenho dos alunos em 2014.2 foi maior em relação aos períodos de 2013.1 ($p\text{-valor} = 0,2020 \times 10^{-8}$), 2013.2 ($p\text{-valor} = 0,5352 \times 10^{-7}$) e 2014.1 ($p\text{-valor} = 0,4199 \times 10^{-6}$).

dominassem os conceitos exigidos em cada etapa). O domínio de unidades é a característica que define a base do *Mastery Learning* e é um dos fatores mais significantes em determinar o desempenho dos alunos nesta abordagem [26]. Esta característica pode provocar grandes melhorias no desempenho da turma, mas também pode prejudicar a cobertura do currículo, dependendo de como for implementada. De acordo com Slavin [51], as abordagens baseadas no *Mastery Learning* possuem um dilema: terem que priorizar a cobertura ou o domínio do conteúdo.

Nas primeiras versões do *Mastery Learning*, ficar preso em uma unidade era sinônimo de ter menos tempo para cobrir todo o conteúdo da disciplina. Houve vários casos de alunos que se atrasaram porque não passaram em algum miniteste e isso acabou gerando uma sucessão de atrasos, que provocaram seu baixo desempenho, pois eles não conseguiram passar em todas as unidades a tempo. Neste sentido, Slavin [51] alerta que passar muito tempo tentando dominar uma unidade e deixar de cobrir as outras pode ser menos produtivo que cobrir todos os objetivos.

Pensando em outra perspectiva, cobrir todo o currículo, mas não permitir o domínio dos conteúdos também prejudica a aprendizagem dos alunos. Em curto prazo, existe o efeito LEM¹¹: se o aluno não dominar os conceitos iniciais de programação, ele apresentará cada vez mais dificuldade nos próximos assuntos, já que um conteúdo é pré-requisito para o próximo. Em longo prazo, Gentile e Lalley [27] explicam que se os alunos não dominarem todos os conceitos, caso forem reprovados ou precisem usá-los em outra disciplina, eles apresentarão dificuldade em reaprendê-los e poderão ficar desmotivados. Pensando neste dilema de exigir dos alunos o domínio do conteúdo, mas também permiti-los cobrir todos os conceitos, adotamos, ao longo dos períodos, estratégias que conciliassem os dois extremos.

Lição aprendida

Ao decidir implementar o *Mastery Learning*, é necessário planejar a avaliação para que haja equilíbrio entre o domínio de unidades e a cobertura do conteúdo.

¹¹*Learning Edge Momentum* é um efeito proposto por Robins [47].

5.4.2 Aprendendo com os erros

Neste capítulo, descobrimos que as primeiras versões do *Mastery Learning* parecem ter provocado resultados negativos no desempenho dos alunos. Nós acreditamos que isto aconteceu porque os professores adotaram critérios de avaliação que atrasaram e penalizaram os estudantes e, por outro lado, não ofereceram oportunidades efetivas para que eles pudessem voltar ao ritmo antes que acabasse o período.

Como vimos no capítulo 3, no começo, os professores se focaram somente em garantir o domínio das unidades, mas não estabeleceram alternativas para que os alunos atrasados voltassem ao ritmo. Alguns estudantes passavam tanto tempo presos nas mesmas unidades — porque só podiam fazer um miniteste por semana — que não tinham tempo de cobrir todo o conteúdo a tempo. O problema deixou de ser os alunos aprenderem o conteúdo superficialmente e passou a ser eles conseguirem estudar as 10 unidades da disciplina dentro do prazo.

Após 3 períodos tentando encontrar um equilíbrio entre garantir o domínio do conteúdo e permitir a cobertura do currículo, em 2014.2 os professores finalmente aprenderam com os erros. A partir deste período, os alunos passaram a ter a possibilidade de fazer vários minitests no mesmo dia, desde que passassem nas unidades em sequência. Isso permitiu que eles voltassem ao ritmo, caso ficassem atrasados ou adiantassem a disciplina, se já soubessem o conteúdo. Essa mudança na avaliação fez com que todos pudessem aprender em seu próprio ritmo e parece ter beneficiado os diferentes perfis de alunos.

Nós acreditamos que a falta de estratégias para os alunos recuperarem o ritmo e a carga de critérios punitivos de avaliação contribuíram para os resultados negativos nos primeiros semestres em que o *Mastery Learning* foi implementado. Por conta disso, destacamos que é essencial para o planejamento desta abordagem que haja medidas como as que foram adotadas em 2014.2, para que os alunos possam dominar efetivamente os conceitos, mas que também tenham tempo de cobrir todo o currículo da disciplina.

Lição aprendida

Precisa-se adotar estratégias para que os alunos consigam voltar ao ritmo, caso fiquem muitas semanas presos em uma unidade.

5.5 Conclusões

Na primeira questão de pesquisa, procuramos saber se a adoção do *Mastery Learning* havia influenciado as taxas de aprovação da disciplina. Acabamos por descobrir que não houve uma mudança significativa entre as taxas de aprovação no método tradicional e no *Mastery Learning*. É importante destacar que a aprovação tem significados diferentes nestas duas abordagens. Na primeira, os alunos podem ser aprovados sem terem dominado os conceitos da disciplina. Já na segunda, apenas aqueles que conseguem dominar e cobrir cerca de dois terços do conteúdo são aprovados. Portanto, embora as taxas de aprovação tenham sido iguais, nós acreditamos que os alunos aprovados no *Mastery Learning* tiveram uma aprendizagem mais sólida.

Na segunda questão de pesquisa, indagamos se os aprimoramentos das versões experimentais do *Mastery Learning* influenciaram o desempenho dos alunos. Nós descobrimos que as estratégias de avaliação adotadas nas primeiras versões do método acabaram influenciando negativamente o desempenho de parte da turma. Todavia, notamos que estes resultados foram melhorando a cada semestre, à medida que os professores aprimoraram os critérios de domínio. Esses resultados serviram para que os professores de Programação I tivessem uma segurança maior de que as mudanças realizadas no método realmente trouxeram benefícios ao desempenho da turma.

Por fim, investigamos se havia diferença entre o desempenho das turmas nas versões experimentais do *Mastery Learning* e em sua versão final. Percebemos que o desempenho dos alunos na versão final foi superior em relação às notas das turmas de todas as versões experimentais. Embora não seja possível garantir que o desempenho dos alunos continue alto nos próximos períodos, acreditamos que a abordagem atual continuará provocando resultados positivos, pois permitiu que os estudantes pudessem dominar e cobrir o conteúdo em seu próprio ritmo.

A partir da experiência obtida nestes dois anos de *Mastery Learning* e com os resultados deste trabalho, acreditamos na seguinte hipótese: para que os alunos de diferentes perfis consigam aprender efetivamente, é necessário evitar as avaliações punitivas e estabelecer estratégias que incentivem tanto o domínio, quanto a cobertura do conteúdo. Para isso, os professores precisam desenvolver avaliações formativas, que permitam aos alunos entender

e superar suas dificuldades, mas que não os proíbam de cobrir currículo básico da disciplina.

Como trabalho futuro, planejamos replicar este estudo em Programação I e em outros cursos, para reforçarmos os resultados obtidos neste trabalho. Esperamos que esta pesquisa possa servir de base para que outros pesquisadores também avaliem a eficácia do *Mastery Learning* em outras instituições e melhorem seu processo de ensino-aprendizagem, assim como aconteceu na UFCG.

Capítulo 6

O *Mastery Learning* sob a perspectiva dos estudantes

Ao longo deste trabalho, descobrimos que o *Mastery Learning* proporcionou efeitos positivos tanto na rotina de estudo quanto na aprendizagem dos alunos de Programação I. Por causa destes resultados e da experiência em sala de aula, o método tem sido bem aceito entre os professores da disciplina. Mas será que os alunos também aprovaram as mudanças provocadas por esta nova abordagem? Neste capítulo, exploramos a aceitação dos alunos em relação ao *Mastery Learning* e suas percepções em relação às vantagens e desvantagens de sua utilização. Ao final deste capítulo, teremos respondido as seguintes questões de pesquisa:

- (Q_1) Os estudantes aprovaram a implementação do *Mastery Learning* em Programação I?
- (Q_2) Os alunos preferem que outras disciplinas usem o *Mastery Learning* ao invés do ensino tradicional?
- (Q_3) Quais as percepções dos alunos em relação às vantagens e desvantagens do método?

6.1 Metodologia

Os resultados deste capítulo foram obtidos por meio de uma pesquisa mista. Num primeiro momento, foi realizado um estudo quantitativo — por meio de um *survey* — para explorar a aceitação e percepções dos alunos em relação ao *Mastery Learning*. Posteriormente, foi

conduzida uma pesquisa qualitativa — através de entrevistas — com o intuito de investigar mais profundamente as respostas do *survey*.

6.1.1 Participantes

Os participantes do *survey* foram alunos novatos da disciplina de Programação I (P1) do curso de Ciência da Computação da UFCG (Tabela 6.1). Foram entrevistados alunos das turmas de 2013.2 e 2014.1 e desconsiderados aqueles que desistiram da disciplina, já que estes podem não ter vivenciado o método por tempo suficiente. A pesquisa qualitativa foi realizada com 9 alunos de 2014.1, que foram selecionados por conveniência.

	2013.2	2014.1
Aprovados	39 (42.4%)	46 (54.1%)
Reprovados	26 (28.3%)	22 (25.9%)
Desistentes	21 (22.8%)	10 (11.8%)
Aprovados na proficiência	6 (6.5%)	7 (8.2%)
Total	92 (100%)	85 (100%)

Tabela 6.1: Caracterização dos alunos matriculados em Programação I

O grupo de alunos aprovados na proficiência é formado por aqueles que passaram em provas que foram realizadas durante o semestre, que tinham o objetivo de liberar os alunos que demonstrassem domínio sobre os conceitos da disciplina. Essa foi uma iniciativa dos professores para permitir que parte dos sujeitos progredisse mais rápido e pudesse aprimorar sua aprendizagem por meio de projetos extracurriculares.

Foram selecionados para o *survey* 54 alunos em 2013.2 e 57 em 2014.1, o que correspondeu a cerca de 76% da turma em cada período. A amostragem foi não-probabilística, utilizando cotas¹ [18] a partir do desempenho acadêmico dos estudantes. Para isso, as turmas foram categorizadas em dois grupos: (A) alunos que dominaram o conteúdo e (B) sujeitos que não conseguiram dominar. O grupo A foi formado por indivíduos que apresentaram média parcial das unidades maior ou igual a 6,0 e o grupo B possuía alunos com média abaixo de 6,0. O número 6,0 foi utilizado como ponto divisório porque representava a nota mínima necessária para passar de unidade. O teste de *Mann-Whitney* confirmou² que as amostras de ambos os períodos eram representativas.

¹Na amostragem por cotas procura-se manter a mesma proporção de indivíduos na amostra e na população, de acordo com uma característica estabelecida.

²A hipótese nula de que as distribuições da amostra e da população são iguais foi aceita ($p\text{-valor} > 0,05$).

6.1.2 Coleta de dados

Foram utilizados questionários online para coletar os dados do *survey*. Eles possuíam questões de múltipla escolha, escalas de Likert e questões discursivas. O questionário de 2014.1 foi um aprimoramento daquele aplicado no período anterior, por conta disso, contém mais questões.

Os questionários foram administrados antes das notas finais serem liberadas para evitar viés na resposta dos alunos que apresentaram baixo desempenho. Os estudantes receberam o questionário em seus e-mails institucionais e tiveram uma semana para respondê-lo. Após esse período, entramos em contato com aqueles que ainda não haviam respondido por meio das redes sociais. Em ambos os períodos, a coleta durou cerca de duas semanas.

Na pesquisa qualitativa foram realizadas entrevistas semi-estruturadas: o entrevistador possuía um roteiro, mas ficava livre para abordar outros assuntos, de acordo com seu interesse. As entrevistas aconteceram durante as 3 últimas semanas de aula e duraram cerca de 13 minutos.

6.1.3 Análise

Na primeira pesquisa, foram analisados inicialmente os dados quantitativos dos questionários por meio de estatística descritiva. Os resultados das escalas de Likert foram agrupados em 3 categorias: insatisfeitos (escalas de 1 a 3), neutros (escala 4) e satisfeitos (escalas de 5 a 7). Posteriormente, foi verificado por meio do teste exato de Fisher [1] se havia relação entre as respostas dos alunos e seu desempenho na disciplina. Os dados das questões abertas foram examinados por meio de análise temática [8]. Na segunda pesquisa, as entrevistas também passaram por um processo de análise temática, onde foram criados temas referentes à opinião dos alunos sobre cada elemento do *Mastery Learning* como o domínio de unidades, ritmo individual, tutoria, avaliação, etc.

6.2 Aceitação

Nesta seção, exploramos a satisfação dos alunos em relação à implementação do *Mastery Learning* em Programação I e a aprovação deles a respeito de características específicas do

método.

6.2.1 Satisfação pelo *Mastery Learning*

Ao realizarmos este estudo, nosso principal objetivo foi saber se as mudanças provocadas pelo método foram aprovadas pela turma. Como forma de identificarmos a satisfação dos estudantes, pedimos que os entrevistados respondessem, em uma escala de 1 a 7, seu nível de satisfação em relação ao *Mastery Learning*. Veja os resultados na Tabela 6.2.

	Insatisfeitos	Neutros	Satisfeitos	Moda
2013.2	2 (3.7%)	7 (13%)	45 (83.3%)	7
2014.1	1 (1.8%)	6 (10.5%)	50 (87.7%)	6

Tabela 6.2: Satisfação pelo *Mastery Learning* entre períodos

A maioria dos entrevistados (85,6%) afirmou que está bastante satisfeita com o método e apenas 3/111 (2,7%) alunos demonstraram insatisfação. Descobrimos também que essa satisfação se manteve de 2013.2 para 2014.1 e não está relacionada com o desempenho dos alunos na disciplina³.

★ Importante

A maioria dos alunos entrevistados afirmou estar satisfeita com o *Mastery Learning*.

6.2.2 Aprovação das características

O *Mastery Learning* possui diversas características que o diferenciam do método tradicional. Nessa seção, investigamos a aprovação dos alunos com relação a alguns dos principais componentes do método⁴.

Domínio de unidades

O principal objetivo do *Mastery Learning* é garantir que todos os alunos possam aprender efetivamente o conteúdo de programação. Para isso, uma das medidas adotadas na disciplina — domínio de unidades — estabelece que os alunos precisam dominar a unidade anterior antes de passarem para a próxima. Nós perguntamos aos estudantes se eles concordavam

³Teste exato de Fisher: p-valor > 0,05.

⁴Veja no Apêndice A as questões abordadas nas tabelas desta seção.

com a rotina imposta pelo domínio de unidades. Tanto os alunos de 2013.2, quanto os de 2014.1 (Tabela 6.3) concordaram que o domínio de unidades os ajudou a aprender e não levar dúvidas adiante. No total, apenas 17/111 (15,3%) dos alunos discordaram dessa característica.

	Discordam	Neutros	Concordam	Moda
2013.2	9 (16.7%)	9 (16.7%)	36 (66.7%)	7
2014.1	8 (14%)	6 (10.5%)	43 (75.4%)	7

Tabela 6.3: Aprovação do domínio de unidades, separada por períodos

Em 2014.1, além de questionarmos sobre o domínio de unidades em si, investigamos também os elementos que complementam essa característica (divisão dos conceitos em unidades e avaliação contínua) e a opinião dos alunos sobre a possibilidade de avançar as unidades sem precisar dominá-las. Veja os resultados na Tabela 6.4 e suas respectivas perguntas no apêndice A.

	Discordam	Neutros	Concordam	Moda
du.divisao	1 (1.8%)	4 (7%)	52 (91.2%)	7
du.positivo	8 (14.1%)	6 (10.5%)	43 (75.4%)	7
du.minitests	4 (7.1%)	3 (5.3%)	50 (87.6%)	7
du.negativo	32 (56.1%)	10 (17.5%)	15 (26.3%)	1

Tabela 6.4: Aprovação do domínio de unidades em 2014.1

A afirmação mais bem aceita entre os alunos (91,2%) foi que a divisão dos conteúdos em unidades facilitou a aprendizagem deles. Além disso, grande parte dos estudantes (87,6%) declarou que prefere minitests semanais a provas ocasionais. Um dos entrevistados explicou:

É bom porque você consegue se acompanhar e ser acompanhado mais continuamente. Se você está com problema em uma unidade de ordenação [por exemplo], você já aprende logo, antes de esperar uns 2 meses para poder ter uma prova e perceber [suas dificuldades].

Ao serem questionados se prefeririam passar de unidade, mesmo não tendo dominado a anterior, mais da metade dos alunos foi contra e apenas 15 foram favoráveis. Um dos entrevistados a favor, disse:

Se eu estivesse indo bem, eu acho que passaria nessa unidade, mas continuaria estudando ela. Passaria, porque às vezes que a gente perde uma questão por dificuldades em matemática não sabemos resolver na hora. Mas eu sei a parte de programação, sei como desenvolver.

Ritmo individual

No *Mastery Learning*, todos os alunos têm a oportunidade de aprender em seu próprio ritmo — ritmo individual. Esta estratégia permite que alunos com experiências e ritmos de aprendizagem diferentes consigam atingir os objetivos propostos no curso. Nós perguntamos aos entrevistados qual a opinião deles sobre a possibilidade dos alunos progredirem na disciplina em ritmos independentes. Grande parte da turma em 2013.2 (81,5%) e 2014.1 (75,4%) concordou que essa estratégia é uma vantagem para a disciplina (Tabela 6.5). Por outro lado, apenas 10/111 (9%) alunos foram contrários a essa característica.

	Discordam	Neutros	Concordam	Moda
2013.2	4 (7.4%)	6 (11.1%)	44 (81.5%)	7
2014.1	6 (10.5%)	8 (14%)	43 (75.4%)	7

Tabela 6.5: Aprovação do ritmo individual, separada por períodos

Durante 2014.1, aprofundamos nossas perguntas sobre o ritmo individual. Exploramos o ponto de vista dos alunos sobre o fato da turma ser dividida, à medida que eles vão progredindo, e a opinião deles sobre a possibilidade de avançarem mais rápido na disciplina. Veja os detalhes na Tabela 6.6.

	Discordam	Neutros	Concordam	Moda
ri.positivo	6 (10.5%)	8 (14%)	43 (75.4%)	7
ri.separacao	3 (5.3%)	8 (14%)	46 (80.7%)	7
ri.avanco	12 (21.1%)	14 (24.6%)	31 (54.4%)	4
ri.negativo	41 (71.9%)	6 (10.5%)	10 (17.5%)	1

Tabela 6.6: Aprovação do ritmo individual em 2014.1

Quando questionada sobre o fato dos alunos serem separados de acordo com seu ritmo de aprendizagem, a maioria dos sujeitos (80,7%) ficou satisfeita. Um dos entrevistados disse: “Eu acho de certa forma necessário, porque não adianta tentar empurrar o aluno que não saiba do assunto, para um assunto mais complexo, sendo que ele não vai acompanhar”. Ao perguntarmos se os alunos concordavam que todos deveriam seguir o ritmo da disciplina, ao

invés de terem ritmos independentes, grande parte (71,9%) discordou. Isso nos leva a crer que eles realmente aprovam uma disciplina que permita cada um progredir em seu ritmo.

Houve um impasse na questão que discutia a possibilidade dos alunos poderem avançar rapidamente na disciplina. Embora metade dos entrevistados tivessem concordado que os professores ofereceram oportunidades para eles avançarem mais rápido, um quarto dos alunos ficou neutro e 21,1% discordaram da afirmação. Um dos alunos afirmou o seguinte: “Por mais que você queira avançar, não tem como. Você está preso. A única coisa que pode fazer é o teste de proficiência para se livrar da disciplina, mas não avançar”. Isso confirma o que afirmamos nos capítulos anteriores sobre não existir estratégias eficientes antes de 2014.2 que ajudassem os alunos a voltarem ao ritmo.

Materiais instrucionais

Os materiais instrucionais são elementos fundamentais cursos baseados no *Mastery Learning*. Como os alunos precisam aprender por conta própria, eles devem ter à sua disposição uma gama de recursos educacionais. Na disciplina, além do livro, são disponibilizados materiais referentes a cada unidade (roteiros). Nós perguntamos aos alunos entre 2013.2 e 2014.1 se eles concordavam que os roteiros eram úteis para seus estudos. Pelos resultados da Tabela 6.7, descobrimos que os alunos de 2013.2 ficaram estatisticamente mais satisfeitos em relação aos entrevistados do último período⁵.

	Discordam	Neutros	Concordam	Moda
2013.2	4 (7.4%)	7 (13%)	43 (79.6%)	7
2014.1	16 (28.1%)	11 (19.3%)	30 (52.6%)	6

Tabela 6.7: Aprovação dos roteiros, separada por períodos

Ao aplicarmos o questionário em 2014.1, também exploramos a opinião dos alunos sobre a eficácia dos livros e outros materiais instrucionais. Veja os resultados na Tabela 6.8. No geral, cerca de metade dos entrevistados em 2014.1 concordou que os livros e roteiros foram úteis e suficientes para estudar. Mas segundo dois terços dos estudantes, outros materiais foram mais importantes, quando comparados àqueles disponibilizados pelos professores.

Tentamos investigar quais materiais os estudantes utilizavam além dos livros e roteiros,

⁵Aplicamos o teste exato de Fisher e rejeitamos a hipótese nula (p -valor $< 0,05$) que as opiniões entre períodos eram iguais.

	Discordam	Neutros	Concordam	Moda
mat.negativo	7 (12.4%)	12 (21.1%)	38 (66.6%)	7
mat.livro	16 (28.1%)	9 (15.8%)	32 (56.1%)	6
mat.roteiros	16 (28.1%)	11 (19.3%)	30 (52.6%)	6
mat.positivo	21 (36.8%)	6 (10.5%)	30 (52.7%)	5

Tabela 6.8: Aprovação dos materiais instrucionais em 2014.1

mas percebemos que não houve um consenso entre eles. Alguns preferiam assistir a vídeo-aulas, outros gostavam de estudar por sites ou blogs, existia também aqueles que preferiam apenas verificar detalhes na documentação.

Feedback

O *feedback* é um dos elementos essenciais para o sucesso do *Mastery Learning*. Como os alunos passam por uma avaliação contínua, eles precisam saber constantemente seu desempenho e quais estão sendo suas principais dificuldades, além de receberem orientações sobre como superá-las. Para isso, contamos com dois tipos de *feedback*: o diagnóstico, que consiste nas informações sobre o desempenho do aluno e seus resultados nos exercícios; e o corretivo, que são os comentários — disponibilizados pelos professores após cada avaliação — que contêm orientações sobre como os alunos podem melhorar seu código. Ao questionarmos se os estudantes concordavam que os testes automáticos os ajudavam a aprender, quase todos⁶ foram favoráveis a essa afirmação, tanto em 2013.2, quanto em 2014.1 (Tabela 6.9).

	Discordam	Neutros	Concordam	Moda
2013.2	0 (0%)	1 (1.9%)	53 (98.1%)	7
2014.1	0 (0%)	1 (1.8%)	56 (98.2%)	7

Tabela 6.9: Aprovação do *feedback* dos testes automáticos, separada por períodos

Exploramos também a visão dos alunos quanto à eficiência dos comentários nos minitestos e da ajuda dos professores e tutores após o *feedback*. Veja os resultados na Tabela 6.10.

Grande parte dos alunos (70, 2%) concordou que os comentários feitos pelos professores a respeito dos erros cometidos nos minitestos são úteis para o aprendizado. Um dos entrevistados comentou: “[Os comentários] ajudam a entender porque o meu programa falhou

⁶Dois alunos ficaram neutros.

	Discordam	Neutros	Concordam	Moda
fb.comentarios	6 (10.6%)	11 (19.3%)	40 (70.2%)	7
fb.testes	0 (0%)	1 (1.8%)	56 (98.2%)	7
fb.negativo	45 (78.9%)	6 (10.5%)	6 (10.5%)	1

Tabela 6.10: Aprovação do *feedback* em 2014.1

ou qual o erro na organização do código. [Isso] ajuda para que eu preste mais atenção e não cometa esse erro outra vez.” Com relação à ajuda dos professores e tutores depois do *feedback*, a maioria dos entrevistados (79%) deixou claro que existe sim uma assistência dos instrutores, caso eles precisem.

Tutoria

A tutoria é vista como uma oportunidade para os alunos poderem praticar e terem ajuda de alguém mais experiente. Nesta modalidade, o tutor fica responsável por acompanhar e auxiliar um grupo de alunos durante todo o semestre. Investigamos durante 2013.2 e 2014.1 o ponto de vista dos alunos sobre a importância da tutoria (veja Tabela 6.11) e descobrimos que 87,4% dos entrevistados concordaram que o acompanhamento semanal dos tutores foi importante. No outro extremo, apenas 8/111 (7,2%) discordaram da afirmação. Um dos entrevistados comentou o seguinte:

Eu achei interessante essa iniciativa de ter alguém mais experiente com um contato mais próximo comigo. Já que o professor não pode dar uma atenção especial, pois a turma é grande. Mas o tutor está mais próximo de você, ele tem menos alunos e vai tirar melhor suas dúvidas e você vai poder interagir melhor com ele.

	Discordam	Neutros	Concordam	Moda
2013.2	2 (3.7%)	3 (5.6%)	49 (90.7%)	7
2014.1	6 (10.5%)	3 (5.3%)	48 (84.2%)	7

Tabela 6.11: Aprovação da tutoria, separada por períodos

Nós também investigamos — em 2014.1 — a satisfação dos alunos pelos tutores e a opinião deles em relação a uma afirmação negativa sobre a tutoria. Veja os resultados na Tabela 6.12.

Descobrimos que mais de 3/4 dos entrevistados ficaram satisfeitos com a ajuda dos tutores. Por outro lado, apenas 4/57 (7%) estudantes acharam que os tutores não ajudaram.

	Discordam	Neutros	Concordam	Moda
tut.positivo	6 (10.6%)	3 (5.3%)	48 (84.1%)	7
tut.tutor	4 (7%)	9 (15.8%)	44 (77.2%)	7
tut.negativo	44 (77.1%)	9 (15.8%)	4 (7.1%)	1

Tabela 6.12: Aprovação da tutoria em 2014.1

Veja, a seguir, o que um dos entrevistados falou sobre seu tutor/monitor.

Não é sempre que temos uma pessoa que nos ajuda a resolver questões nas quais temos dúvidas. Por exemplo, o monitor traz atividades e se eu não resolver, ele explica o que eu estou errando. Nem sempre o professor tem a disponibilidade de me ajudar.

Quando questionamos aos alunos se a tutoria não tinha contribuído em nada para seu aprendizado, 77,1% deles discordaram e 4 foram a favor dessa afirmação.

6.3 Preferência

Nesta seção, investigaremos se os alunos preferem que a disciplina de Programação I continue usando o *Mastery Learning* ou volte a usar a abordagem tradicional. Exploraremos também a opinião deles a respeito da expansão do método para turmas de repetentes e para Programação II.

6.3.1 Continuação nos próximos períodos

Ao serem questionados sobre qual método deveria ser utilizado em Programação 1 no próximo período, quase todos os entrevistados (97%) responderam que a melhor opção seria continuar com o *Mastery Learning*. Num total de 111 alunos, apenas 3 sugeriram que os professores deveriam voltar ao método tradicional de ensino.

Comparando as respostas de 2013.2 e 2014.1 (Figura 6.1), notamos que os estudantes do último período ficaram mais satisfeitos com a forma que o método foi implementado⁷. Em 2013.2, o número de entrevistados que sugeriram mudanças no método foi mais que o dobro em relação aos que estavam totalmente satisfeitos com a implementação. Já em 2014.1, essa

⁷Há uma diferença significativa (p-valor < 0,05) entre a preferência dos alunos de 2013.2 e 2014.1.

diferença praticamente se equilibrou. Importante ressaltar que essa preferência dos alunos — em ambos os períodos — não está relacionada ao seu desempenho na disciplina⁸.

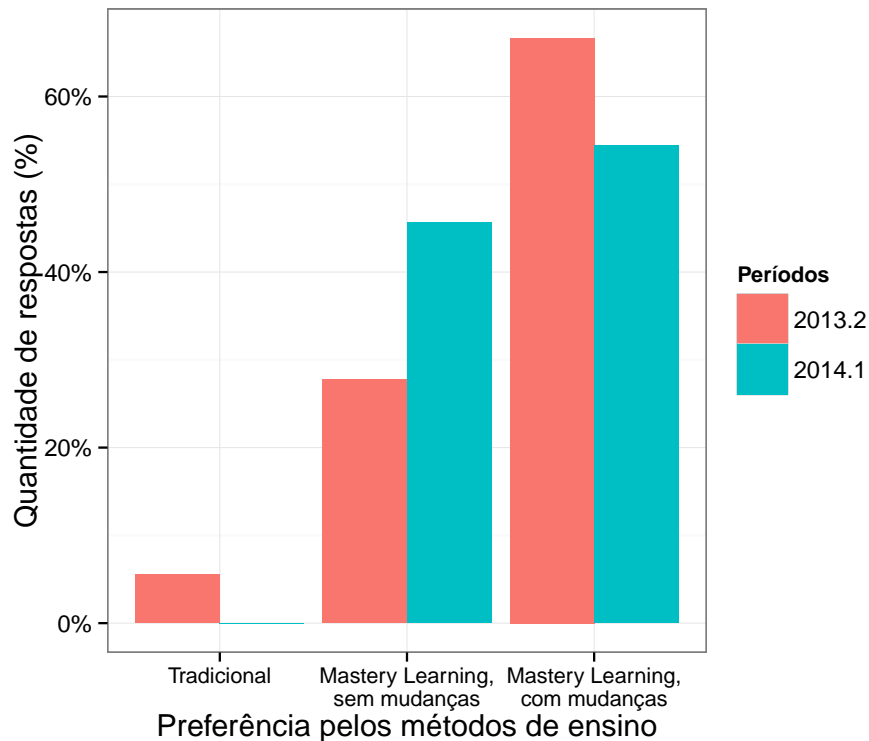


Figura 6.1: Preferência pela continuação do *Mastery Learning* em P1

A principal crítica dos entrevistados em 2013.2 que não ficaram totalmente satisfeitos com o *Mastery Learning* foi a respeito da dificuldade de progredir na disciplina por causa das restrições na avaliação. Neste período, os estudantes precisavam passar em um miniteste teórico e dois práticos para conseguirem avançar de unidade. Se o aluno não conseguisse passar em uma dessas avaliações, ele ficaria para trás e teria como acompanhar o resto da turma. Um dos entrevistados declarou: “O fato de só poder fazer o miniteste prático depois de ter passado no teórico, isso pode atrasar muito às vezes”.

Lição aprendida

Os professores prejudicaram o progresso dos alunos, pois criaram um critério de domínio muito rígido em 2013.2 e, principalmente, não ofereceram oportunidades para que os atrasados pudessem voltar ao ritmo.

⁸Não há uma diferença significativa ($p\text{-valor} > 0,05$) entre as respostas e o desempenho dos alunos na disciplina.

Em 2014.1, os estudantes que sugeriram mudanças no *Mastery Learning* criticaram 3 aspectos em sua implementação: (a) a falta de personalização do ensino; (b) o atraso no *feedback*; e (c) o tempo disponível para avaliação.

- (a) No período de 2014.1, os professores deixaram de personalizar o ensino para os alunos que estavam em unidades diferentes e voltaram a ensinar os mesmos conteúdos para toda a classe. Vários entrevistados reclamaram que assistiram aulas sobre conteúdos avançados enquanto ainda estavam presos em unidades anteriores. Um deles disse: “[o método] dificulta para alunos que aprendem em um ritmo muito lento, pois [eles] têm que entender conteúdos novos sem terem entendido os anteriores”. Outro entrevistado alertou: “Quem deixa de acompanhar a disciplina, mesmo que por pouco [tempo], acaba se perdendo no meio do caminho e torna-se difícil retornar ao ritmo da turma, tendendo muito facilmente à reprovação.”
- (b) Outra crítica bastante recorrente foi a respeito do atraso na correção e *feedback* sobre os minitests. Um entrevistado explicou bem a situação: “[A] demora na correção de alguns minitests de certa forma anula o benefício de haver avaliações semanais, tendo em vista que se não tenho meu miniteste corrigido antes de fazer o próximo, não estarei completamente ciente do meu atual desempenho.”
- (c) Alguns alunos reclamaram que o tempo para realização dos minitests era curto para unidades mais avançadas, onde eles tinham que resolver questões mais complexas. Um dos entrevistados sugeriu que “o tempo dos minitests deveria ser menor nas primeiras unidades e progredir com o avançar delas!”.

Lição aprendida

A volta das aulas tradicionais em 2014.1 acabou atrapalhando os alunos que ficavam para trás. Eles precisavam assistir aulas dos próximos conteúdos, enquanto ainda estavam se focando e sendo avaliados em unidades anteriores.

6.3.2 Continuação após reprovação

Nós exploramos a opinião dos estudantes sobre continuar utilizando o *Mastery Learning* caso fossem reprovados. A maioria deles (87%) afirmou que, se não fossem aprovados,

preferiria continuar em uma turma que seguisse a dinâmica do *Mastery Learning*. Apenas 14/111 alunos (13%) quiseram cursar uma disciplina que se baseasse no método tradicional. Veja na Figura 6.2 a preferência dos alunos em cada período.

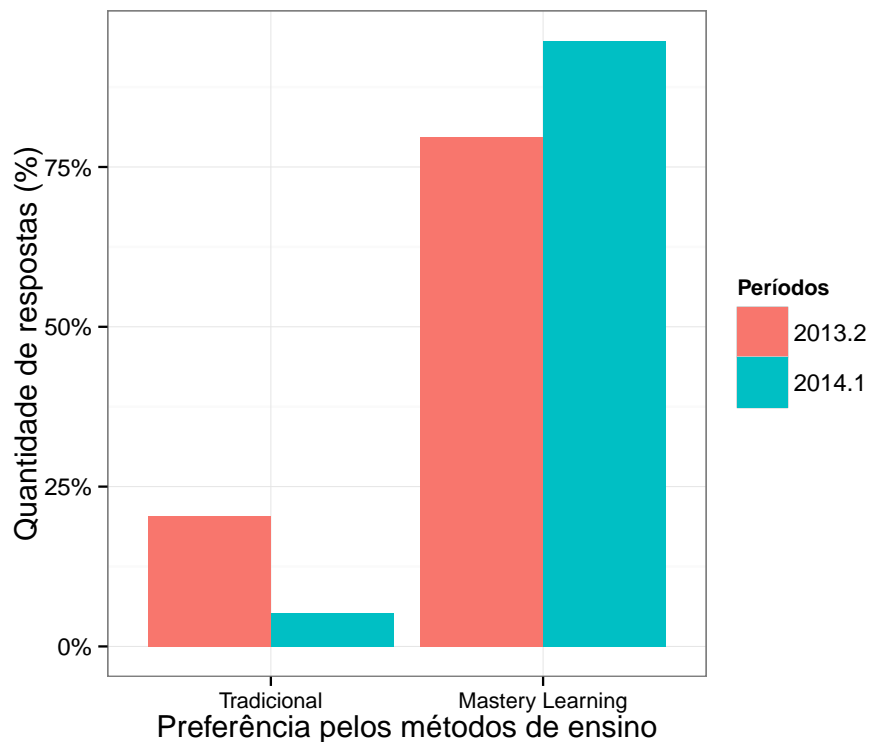


Figura 6.2: Preferência pela continuação do *Mastery Learning* em turmas de alunos que reprovaram

Ao analisarmos as respostas por período, constatamos que entre 2013.2 e 2014.1, houve um aumento na quantidade de alunos satisfeitos com o *Mastery Learning* ¹. Em 2013.2, o número de entrevistados que apoiaram o método foi 4 vezes maior que os interessados no método tradicional. Já em 2014.1, essa diferença aumentou para 18 vezes. Além disso, em 2013.2, os sujeitos que escolheram o método tradicional geralmente apresentaram baixo desempenho na disciplina; já em 2014.1, não houve uma relação entre o desempenho e a preferência dos entrevistados².

6.3.3 Expansão para outra disciplina

Investigamos a opinião dos estudantes em relação à expansão do *Mastery Learning* para Programação II (P2). A maioria (86%) se mostrou positiva para essa mudança e apenas

15/111 alunos (14%) foram contra.

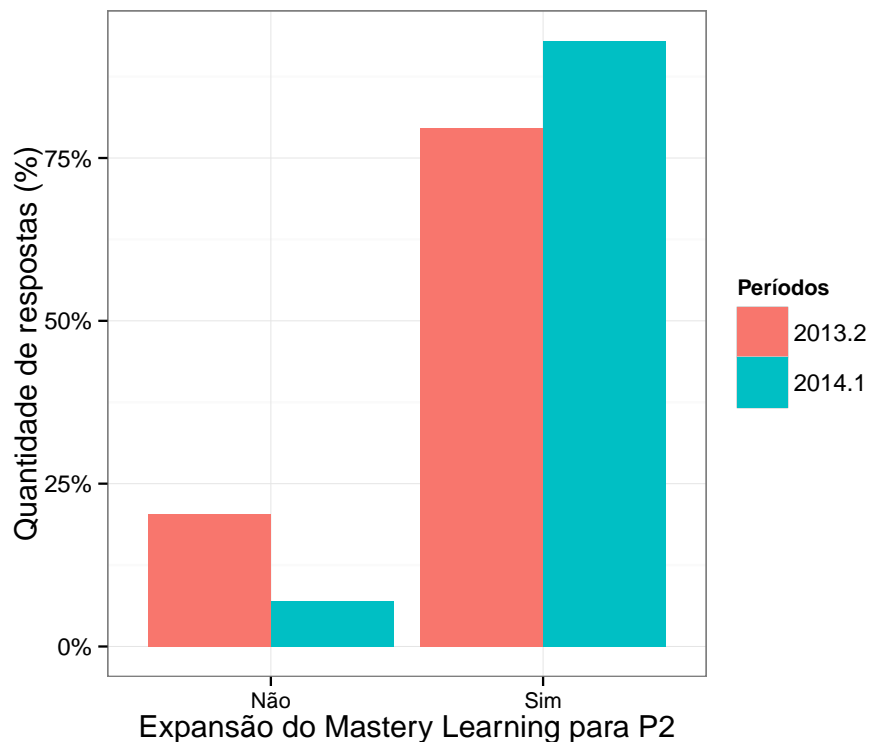


Figura 6.3: Preferência pela expansão do *Mastery Learning* para P2

Seguindo a tendência das seções 6.3.1 e 6.3.2, os resultados da Figura 6.3 mostram que os entrevistados em 2014.1 foram mais favoráveis à adoção do *Mastery Learning* em relação àqueles de 2013.2¹, embora a diferença tenha sido mais equilibrada. Em 2013.2, havia 4 vezes mais alunos que apoiavam a mudança em relação aos que eram contra. Em 2014.1, essa diferença passou a ser de 13 vezes. Os entrevistados em 2013.2 que apresentaram baixo desempenho tenderam a preferir o ensino tradicional, enquanto aqueles com desempenho maior geralmente optaram pelo *Mastery Learning*. Em 2014.1, não houve relação entre o desempenho dos alunos e sua preferência pelos métodos².

Ao serem questionados sobre a preferência pela expansão do *Mastery Learning* para P2, os estudantes que foram contra não explicaram sua opinião, já aqueles que foram favoráveis à mudança justificaram que o método possui uma avaliação contínua, que os ajuda a acompanhar melhor sua evolução na disciplina, permitindo que eles se policiem a continuar estudando. Um dos alunos disse o seguinte:

O método de avaliação adotado em P1 é excelente para o aprendizado,

tendo em vista que em qualquer dificuldade, o aluno pode se manter no assunto até dominá-lo. Além disso, serve como um medidor de desempenho, propiciando aos professores focarem a atenção nos estudantes que estão em unidades menores, ou seja, os que estão com dificuldade no assunto; tudo isso sem atrasar o conteúdo ou limitar os alunos que já têm algum conhecimento ou que adiantam-se na matéria.

6.4 Prós e contras

Nesta seção, descobriremos as percepções dos alunos em relação ao *Mastery Learning*. A partir dos temas gerados na análise temática, conseguimos identificar as vantagens e desvantagens do método, do ponto de vista dos entrevistados.

6.4.1 Vantagens

Divisão dos conceitos

Em muitas universidades o conteúdo programático da disciplina é dividido em 2 ou 3 unidades e os professores abordam inúmeros conceitos antes de realizarem a primeira avaliação. Com isso, eles perdem a oportunidade de identificar precocemente os alunos que estão com dificuldades e, conseqüentemente, ajudá-los a superar suas dificuldades.

No *Mastery Learning*, cada conceito é separado em uma unidade, cujos objetivos de aprendizagem são bem definidos e explícitos para os estudantes. Na visão de um aluno, “a divisão do conteúdo gera maior aprendizado”; outro complementa que “[...] é mais fácil de aprender se o assunto for transmitido aos poucos”. Um terceiro entrevistado sugere que a divisão dos conceitos facilita a identificação de possíveis dificuldades: “[...] o fato do conteúdo ser avaliado semanalmente me ajuda a encontrar carências em determinados assuntos e assim posso me focar nelas, diferentemente do que ocorre nas outras disciplinas, que descarregam vários conteúdos e cobram tudo de uma vez”⁹.

⁹Precisamos adaptar a citação para facilitar a leitura. Veja a versão original: “[...] o fato de ser analisado semanalmente mostra a carência em determinados assuntos e assim podemos focar, diferentemente no que ocorre nas outras disciplinas, no qual descarregam um montante de conteúdo e cobram tudo de uma vez”

Avaliação contínua

No *Mastery Learning*, os alunos fazem minitestos semanais, referentes aos conteúdos estudados na unidade atual. Essa estratégia visa diminuir a quantidade de conteúdo por avaliação e incentivar que os alunos estudem continuamente para a disciplina.

Para alguns entrevistados, “[os] minitestos semanais fazem com que o aluno estude diariamente” e “esteja sempre envolvido com programação”. De acordo com outros alunos, “a avaliação contínua é bem melhor para aprender que apenas 3 avaliações em todo período” porque “você estuda mais para um único conteúdo, o que faz com que esse conteúdo seja melhor assimilado”.

Domínio de unidades

A principal característica do *Mastery Learning* é garantir que os alunos só avancem para a próxima unidade quando dominarem os conceitos da unidade anterior. Corroborando com essa filosofia, um entrevistado afirmou que “o ponto mais positivo [do método] é que o aluno só pode avançar quando dominar o conteúdo anteriormente ministrado”. De acordo com outro sujeito, “[o método] exige que você tenha realmente aprendido o que é ensinado em cada unidade”.

Ritmo independente

No *Mastery Learning*, os alunos aprendem em ritmos diferentes e recebem auxílio apropriado para conseguirem avançar na disciplina. De acordo com alguns entrevistados, “[a disciplina] não força os alunos a seguirem um ritmo específico” e “cada aluno tem a oportunidade de aprender no seu tempo”. Na percepção de um aluno, “[o método] facilita para alunos que aprendem em um ritmo mais rápido ou que já tiveram contato com programação ou com a linguagem”. Isso é possível graças aos testes de proficiência que acontecem ao longo do período e servem para adiantar os estudantes que já possuíam experiência anterior em programação. No outro extremo, os alunos que estão com dificuldades são assistidos pelos tutores e professores.

Feedback

O *feedback* rápido, diagnóstico e corretivo é uma das características essenciais do *Mastery Learning*. Como cada aluno tem a possibilidade de progredir em seu ritmo, ele precisa de um mecanismo eficiente que informe qual sua situação na disciplina. Além disso, os alunos com dificuldades precisam entender o que estão errando e, principalmente, como superar essas dificuldades.

Para tentar auxiliar os estudantes, a disciplina oferece *feedback* em dois elementos: nos exercícios, por meio de um sistema automático de correção de exercícios; e na avaliação, por meio da divulgação semanal de notas e de comentários nos códigos dos alunos.

Segundo o ponto de vista de alguns entrevistados, a disciplina “tem um sistema eficiente de *feedback*”. Além disso, alunos destacaram que o “*feedback* [é] rápido para tudo o que fazemos” e “o *feedback* dos minitestes resalta onde estão os erros e dificuldades”.

6.4.2 Desvantagens

Ritmo intenso

Vimos na seção anterior que a avaliação contínua foi apontada por certos alunos como sendo uma vantagem. No entanto, outros reclamaram que ter minitestes semanais é desgastante porque exige um esforço contínuo durante todo o semestre.

De acordo com alguns entrevistados, o método de avaliação é desgastante: “a carga de provas e correções desgastam alunos e professores”, “embora o miniteste force o aluno a testar seus conhecimentos, uma vez por semana, é muito exaustivo, às vezes” e “[o] método de avaliação [é] um pouco exaustivo, já que há a necessidade de praticar sempre”. Outro aluno reclamou, mas concordou que essa cobrança do método traz benefícios: “Provas semanais continuam a ser provas. Para quem não está completamente acostumado pode relaxar ou pressionar em excesso. Embora sejam melhores para focar nas dificuldades, pois garantem que se tenha certo domínio do assunto”.

Retenção nas unidades

Para os alunos progredirem de unidade em 2013.2, eles precisavam passar em um miniteste teórico — que não valia nota, mas era requisito para liberar os minitestes práticos — e alcan-

çar a nota 6,0 em pelo menos dois minitests práticos. Como já discutimos anteriormente, esta estratégia causou várias reações negativas por parte dos entrevistados. Alguns afirmaram que o miniteste teórico os atrasou: “o fato de só poder fazer o miniteste prático depois de ter passado no teórico, isso pode atrasar muito as vezes” e “você ficar retido numa unidade por causa da prova teórica [...] muitas vezes você sabe o conteúdo, porém por falta de atenção ou por situações nunca vistas antes você termina ficando preso”. Outro estudante alegou que a disciplina não ofereceu estratégias para que os alunos recuperassem o ritmo, caso ficassem atrasados: “quando alguns alunos estão com dúvidas e não conseguem passar de unidade, é criada uma espécie de bola de neve e [eles] acabam ficando sem nota na maioria das unidades, enquanto os outros já acabaram todas”.

Aulas descontextualizadas

Em 2014.1, houve uma redução no quadro de professores e eles resolveram dar aulas tradicionais, onde ensinaram a todos os mesmos conteúdos. Isso foi de encontro à característica do *Mastery Learning* de oferecer auxílio personalizado aos diferentes tipos de alunos e, infelizmente, trouxe resultados negativos para a turma. Os indivíduos que ficaram para trás precisaram estudar conceitos de unidades avançadas, ficando desestimulados. De acordo com um entrevistado, foi problemático “não dar o conteúdo de acordo com o ritmo determinado do aluno, ou seja passar pra ele um conteúdo à frente, sendo que ele está preso no conteúdo anterior”. Outro estudante, justificou que isso “dificulta para alunos que aprendem em um ritmo muito lento, pois [eles] têm que entender conteúdos novos sem ter entendido os anteriores, e enquanto aprendem conteúdos novos [precisam] se preparar para os minitests semanais”. Um terceiro entrevistado alertou: “quem deixa de acompanhar a disciplina, mesmo que por pouco, acaba se perdendo no meio do caminho e torna-se difícil retornar ao ritmo da turma, tendendo muito facilmente à reprovação”.

6.5 Conclusões

Na primeira questão de pesquisa deste capítulo, questionamos se os estudantes aprovaram a implementação do *Mastery Learning* em Programação I. A partir dos resultados, descobrimos que grande parte dos alunos ficou satisfeita com o método. Dentre as características

observadas, o *feedback* dos testes automáticos, a divisão dos conteúdos em unidades e a tutoria foram as que apresentaram o maior nível de aceitação. Características como o domínio de unidades e o ritmo independente também foram bem aceitas entre os alunos, embora alguns deles tenham apontado falhas em sua implementação.

Nós também investigamos — a partir da segunda questão de pesquisa — se os alunos preferem que o *Mastery Learning* continue sendo utilizado em Programação I e seja expandido para Programação II. Nós descobrimos que a maioria dos entrevistados concordou que o método deveria continuar na disciplina e ser expandido para outras. Além disso, grande parte dos alunos afirmou que preferiria utilizar novamente o *Mastery Learning* caso fosse reprovado.

Na última questão de pesquisa, investigamos as percepções dos alunos em relação às vantagens e desvantagens do *Mastery Learning*. Percebemos que as vantagens foram relacionadas aos elementos do método, como avaliações contínuas, domínio das unidades e ritmo individual. Com relação às desvantagens, descobrimos que os sujeitos não se focaram em características da abordagem, mas em aspectos da implementação que lhes causaram impactos negativos. Esses resultados foram importantes, pois permitiram que os professores aprimorassem a versão final do *Mastery Learning*, evitando os aspectos negativos e reforçando as características positivas à turma.

Nós tomamos algumas precauções para garantirmos a validade dessa pesquisa. Para combatermos ameaças de seleção à validade interna, realizamos uma amostragem por cotas e conseguimos amostras que fossem representativas. Com relação à validade de conteúdo, expandimos o questionário de 2014.1 para que ele abrangesse todas as características do *Mastery Learning*. Nós também tentamos garantir a validade de construto ao investigarmos a aceitação dos alunos a partir de várias perspectivas, além de usarmos a triangulação das respostas dos questionários e das entrevistas. Infelizmente, não conseguimos assegurar a validade externa, já que investigamos apenas a opinião dos alunos de uma instituição. Esperamos que, futuramente, este trabalho seja replicado em outras instituições para que haja uma generalização dos resultados.

Apesar de termos utilizado algumas estratégias para evitarmos ameaças à validade desta pesquisa, destacamos que não há como garantir que as respostas dadas pelos alunos no *survey* e nas entrevistas retratem, de fato, suas reais opiniões sobre o *Mastery Learning*. Temos que

considerar que as respostas dos alunos possam ter algum viés positivo ou negativo em relação ao método. Portanto, são necessários estudos mais específicos (como por perfis de aprovados e reprovados, separadamente) e abrangentes (durante mais períodos) para reforçarmos os resultados desta pesquisa.

O *Mastery Learning* provocou mudanças consideráveis na rotina dos estudantes e os transformou em membros ativos no processo de ensino-aprendizagem. Antes de realizarmos esta pesquisa, os professores não tinham ideia de como os alunos haviam reagido a essa mudança de paradigma. Hoje, os docentes possuem indícios de que a maior parte dos estudantes é favorável ao método e que eles gostariam que outras disciplinas também adotassem essa abordagem.

Capítulo 7

Discussões

7.1 E aí, o *Mastery Learning* deu certo?

O *Mastery Learning* foi proposto por Bloom [11] como uma solução intermediária — entre o método tradicional e o ensino de um-para-um — para promover uma aprendizagem mais justa entre os diferentes tipos de alunos. Essa abordagem se adapta ao sistema educacional de hoje e, ao mesmo tempo, garante que todos os estudantes consigam dominar o conteúdo em seu próprio ritmo. Mas será que a implementação do método na UFCG realmente trouxe benefícios para a turma de Programação I? Nós achamos que sim.

Com a adoção do *Mastery Learning*, os alunos passaram a ter uma rotina de estudos mais focada e personalizada. Aqueles que antes não sabiam o que estudar passaram a se focar em uma unidade de cada vez e puderam aprender cada conceito antes de avançar. Já os mais adiantados, não precisaram esperar o resto da turma e conseguiram terminar o conteúdo antes do final do semestre, passando a explorar tarefas mais complexas. Embora a disciplina tenha sido a mesma para todos, ficou nítido que cada aluno aprendeu a partir de estratégias completamente diferentes.

Nós acreditamos que a rotina de estudo proporcionada pelo *Mastery Learning* permitiu que os alunos levassem menos dúvidas adiante. Uma justificativa para isso é o fato deles passarem mais tempo reforçando um conteúdo antes de avançar e revisarem menos os conceitos que já haviam estudado antes. Se levarmos em consideração a teoria do Modelo de Aprendizagem Escolar [16] (que define a aprendizagem como sendo uma função entre o tempo gasto estudando e o necessário para aprender), podemos afirmar que os alunos no *Mastery*

Learning tiveram uma aprendizagem mais sólida, pois passam mais tempo superando suas dificuldades.

Hipótese

A rotina de estudos no *Mastery Learning* proporcionou que os alunos levassem menos dúvidas adiante.

Apesar do *Mastery Learning* ser mais centrado nos alunos com dificuldades, é importante destacar que ele também exerceu um papel fundamental na aprendizagem dos indivíduos mais adiantados. Eles tiveram a oportunidade de dominar cada unidade e cobrir o conteúdo de uma forma mais homogênea, pois não precisaram passar muito tempo estudando os mesmos conceitos (diferente dos alunos no método tradicional, que ficavam praticando determinadas unidades mais tempo que o necessário, porque precisavam esperar pelo resto da turma). Além disso, estes sujeitos tiveram a chance de enriquecer seu conhecimento por meio das medidas de aperfeiçoamento, como o desenvolvimento de um jogo ou atividades extracurriculares, como o treinamento para maratonas de programação. Desta forma, todos conseguiram explorar ao máximo seu potencial na disciplina.

O domínio de unidades, em conjunto com as medidas de correção e aperfeiçoamento, parecem ter permitido que todos os estudantes alcançassem uma aprendizagem mais sólida em Programação I. Os aprovados demonstraram domínio sobre todos (ou quase todos) os conceitos da disciplina e os reprovados conseguiram reforçar suas dificuldades nas unidades básicas. Se considerarmos o processo de reaprendizagem citado por Gentile e Lalley [27] (no qual os sujeitos que dominam um conteúdo inicialmente têm mais facilidade de reaprendê-lo futuramente), temos a impressão de que os aprovados no *Mastery Learning* podem apresentar menos dificuldades aprender conteúdos de disciplinas correlatas — como Programação II — e os reprovados conseguem reaprender com mais facilidade os conteúdos básicos de Programação I.

Hipótese

Os alunos reprovados e aprovados no *Mastery Learning* têm mais facilidade em reaprender os conceitos de Programação I e estudar disciplinas correlatas, respectivamente.

Portanto, a resposta para o título desta seção é sim, o *Mastery Learning* deu certo em Programação I. Ao contrário do método tradicional, que promove uma rotina de estudos pas-

siva e genérica (*one fits all*), a abordagem analisada neste trabalho trouxe consigo elementos que favoreceram uma aprendizagem mais rica, ativa e personalizada. Depois do *Mastery Learning*, acreditamos que os alunos começaram a sair da disciplina muito mais preparados para reaprender os conceitos no futuro e possivelmente apresentaram menos dificuldades em disciplinas correlatas.

7.2 Por que todos não usam o *Mastery Learning*?

Os benefícios do *Mastery Learning* já foram amplamente relatados na literatura ao longo dos últimos 40 anos. No entanto, grande parte dos professores continua preferindo utilizar o método tradicional ao invés desta outra abordagem. Nós acreditamos que eles deixam de adotar o *Mastery Learning* principalmente por causa do custo inicial de implementação e pela resistência a mudanças. Para alguns, é mais cômodo continuar ministrando as mesmas aulas todo semestre e fazer avaliações ocasionais a ter que se dedicar em reformular todos os aspectos da disciplina.

O custo inicial para começar a utilizar o *Mastery Learning* realmente não é baixo. É necessário repensar vários os aspectos da disciplina (estratégias de ensino, avaliação, correção, aperfeiçoamento, etc.), de forma que todos os alunos tenham a oportunidade de dominar o conteúdo. Para isso, o professor precisa planejar com antecedência a divisão do conteúdo em unidades menores (com objetivos de aprendizagem bem definidos), criar ou procurar materiais instrucionais, elaborar questões para as avaliações semanais, entre outras tarefas. Adotar o *Mastery Learning* demanda tempo e esforço. Por isso, a transição de métodos geralmente não acontece em apenas um período.

Em Programação I, os professores levaram 2 anos para adaptar todos os elementos do *Mastery Learning* à dinâmica da disciplina. Foi necessário elaborar os roteiros de estudo e vídeo-aulas, bem como criar constantemente novas questões para as listas de exercícios e avaliações semanais. No entanto, hoje, após a consolidação do método e do ferramental, os professores têm mais tempo para explorar as particularidades de cada aluno.

Outro fator que dificulta a adoção do *Mastery Learning* é a resistência de alguns professores ou dos administradores de instituições de ensino. Alguns professores têm receio de tentar abordagens que sejam muito diferentes das que foram utilizadas quando eles eram alunos.

Para os professores da UFCG, por exemplo, foi difícil deixar de ministrar aulas expositivas, pois eles acreditavam que esta era a forma mais eficaz dos alunos aprenderem. Felizmente, ao “inverterem” a sala de aula, eles perceberam que esta mudança só trouxe benefícios à rotina de estudos proporcionada pelos outros elementos do *Mastery Learning*.

Apesar das dificuldades que existem ao se tentar adotar o *Mastery Learning*, é preciso ter em mente que seus benefícios ao processo de ensino-aprendizagem não deixam dúvidas: vale a pena mudar. Na próxima seção, serão apresentadas algumas lições aprendidas pelos professores da UFCG, durante o processo de amadurecimento do método na disciplina. Esperamos que elas sirvam como ponto de partida para outros professores que estejam interessados nesta abordagem.

7.3 Lições aprendidas

Para adotar o *Mastery Learning* é necessário dedicar tempo e esforço para planejar como o método atuará no ensino e na avaliação, além de definir quais serão as estratégias de correção e aperfeiçoamento. O professor precisa sair de sua zona de conforto e redefinir suas concepções sobre o processo de ensino-aprendizagem. Esta seção possui um resumo das lições aprendidas pelos professores durante os dois anos de experiência com o *Mastery Learning* em Programação I.

Um dos primeiros desafios que o docente precisa lidar ao implementar o *Mastery Learning* é possibilitar que todos aprendam em seu próprio ritmo. Infelizmente, o modelo de aulas expositivas é incompatível com esta proposta, pois não há como um único professor ministrar aulas de várias unidades na mesma semana. Em Programação I, como a disciplina possuía 3 professores, eles tentaram manter as aulas expositivas por um tempo. No entanto, quando chegavam em determinado ponto da disciplina — no qual havia alunos estudando em mais de 3 unidades — os professores precisavam continuar no ritmo dos mais avançados e deixavam os sujeitos que estavam em unidades mais antigas para trás. Portanto, precisamos entender que, se quisermos utilizar o *Mastery Learning*, é necessário abrir de mão das aulas expositivas.

Os professores de Programação I perceberam que não fazia sentido oferecer uma abordagem personalizada, mas ministrar aulas que só beneficiavam os que estavam mais adiantados.

A saída para este problema foi incorporar à disciplina os elementos da sala de aula invertida, incentivando os alunos estudarem novos conceitos em casa e frequentarem as aulas apenas para discutirem com professores e colegas suas dúvidas e estratégias de resolução de problemas. Desta forma, os alunos poderiam estudar em seu próprio ritmo e o professor utilizaria o tempo das aulas para criar interações entre professor-alunos e alunos-alunos, a fim de criar um ambiente mais ativo, onde todos ensinariam e aprenderiam ao mesmo tempo.

Quando o professor adota um modelo de sala de aula invertida, é preciso oferecer aos alunos uma gama de recursos educacionais, para que eles tenham a possibilidade de explorar os conceitos de várias maneiras e, assim, chegar na sala de aula com uma ideia formada sobre o assunto. Para solucionar esse problema, aconselhamos que, nos primeiros semestres, sejam utilizados materiais de outras universidades, enquanto não são elaborados os materiais da disciplina. À medida que o método for se estabilizando e o trabalho for diminuindo, o professor terá mais tempo para produzir seu próprio material.

Outro ponto que devemos levar em consideração ao utilizarmos o *Mastery Learning* é a avaliação. Ela precisa garantir o domínio de unidades, mas, ao mesmo tempo, permitir que os alunos voltem ao ritmo, caso fiquem atrasados. Como vimos ao longo deste trabalho, esta etapa é crítica e exige que o professor tome cuidado para equilibrar o critério de domínio de uma forma que nem atrase demais os alunos, nem deixe a avaliação muito leve, o que não garantiria o domínio do conteúdo. Nós acreditamos que a solução encontrada em 2014.2 parece solucionar este problema, mas demanda esforço para ser implementada.

Para que haja avaliações semanais, é preciso que o professor elabore com antecedência um banco de questões, pois são necessários minitestes para todas as unidades que estão sendo estudadas na semana. Em programação I, para facilitar a elaboração e correção, os professores desenvolveram o TST — sistema de correções automáticas — que funciona como um juiz online, recebendo questões padronizadas (com entradas e saídas esperadas) e avaliando as respostas por meio de testes de caixa preta. Embora esta solução seja eficiente, o professor precisa desenvolver um sistema semelhante, ou se adaptar às restrições de algum que já esteja disponível, como o URI¹ ou o SPOX².

Por fim, o último ponto que achamos relevante é o *feedback*. Como o *Mastery Learning*

¹<https://www.urionlinejudge.com.br/academic>

²<http://spox.spoj.pl/>

é um método no qual os alunos são constantemente testados quanto à sua aprendizagem e precisam receber medidas para corrigir suas dificuldades ou incentivar seu bom desempenho, é necessário que eles estejam sempre a par de como estão se saindo em cada unidade. Da mesma forma, os tutores também precisam acompanhar de perto o desempenho de seus estudantes, para que possam auxiliá-los ao longo da disciplina. Neste sentido, é necessário que os professores estabeleçam um *feedback* imediato a todos os envolvidos, de modo que eles consigam entender o mais rápido possível quais as dificuldades que precisam superar. Nós aconselhamos que os professores disponibilizem aos tutores os resultados das avaliações de seus alunos, para que eles identifiquem as necessidades de cada um e auxiliem-nos ao longo da semana.

Começar a utilizar o *Mastery Learning* não é fácil e requer um esforço considerável nos primeiros semestres. No entanto, após o método estar consolidado, o professor não tem muito trabalho para mantê-lo. Para facilitar este processo de adaptação, recomendamos que o método seja implementado aos poucos e evolua, à medida que o trabalho inicial for diminuindo. Acreditamos que as lições aprendidas nesta seção possam tornar ainda mais fácil o processo de implementação do *Mastery Learning* em outras instituições. Esperamos que, no futuro, outros professores também possam colher os frutos dessa mudança, assim como aconteceu na disciplina de Programação I na UFCG.

7.4 O *Mastery Learning* vale a pena?

Como já falamos durante todo este trabalho, temos grande confiança de que o *Mastery Learning* melhorou significativamente a rotina de estudo e o desempenho dos diferentes perfis de alunos. O método lhes incentivou a estudar continuamente — num ritmo independente — e proporcionou uma aprendizagem mais ativa e efetiva. Mas este não é o único motivo que nos anima a dizer que o *Mastery Learning* vale a pena. Além de todas as vantagens ligadas à aprendizagem, descobrimos que esta abordagem foi bastante aceita tanto pelos alunos, quanto pelos professores.

Nós descobrimos que os estudantes ficaram bastante satisfeitos por terem passado pelo *Mastery Learning* e acham que essa abordagem precisa continuar e ser expandida para outras disciplinas. Como justificativa, eles alegaram que o método lhes proporcionou a ter uma

rotina de estudos mais ativa e organizada. Os professores também demonstraram extrema satisfação em relação ao método por causa das experiências positivas que ele vem proporcionando à disciplina. Eles alegaram que, além de melhorar as notas, a abordagem possibilita que eles tenham uma rotina de estudos mais produtiva.

Em suma, acreditamos que o *Mastery Learning* é uma abordagem que tem tudo pra dar certo e possui vantagens que superam de longe o trabalho inicial que é preciso para sua implementação. Por conta disso, achamos que vale a pena começar a utilizá-lo no contexto de programação ou em outras áreas, tanto pela sua eficácia, quanto pela satisfação dos professores e alunos.

Capítulo 8

Considerações finais

Os professores tinham a noção de que o método tradicional não beneficiava completamente todos os tipos de alunos. A partir desta inquietação, começaram a buscar alternativas que tornassem processo de ensino-aprendizagem mais personalizado. O *Mastery Learning* proporcionou que os alunos se tornassem muito mais ativos e conseguissem aprender o conteúdo de uma forma mais profunda. Para avaliar os efeitos do método, buscamos entender o que mudou em relação ao comportamento dos alunos no curso e qual a influência disso em seu desempenho.

Com relação às influências do *Mastery Learning* no comportamento dos estudantes, descobrimos que houve mudanças significativas na rotina de estudo dos diferentes perfis de alunos. Aqueles com maior aptidão conseguiram se focar em mais unidades e avançaram mais rápido na disciplina. Já os que aprendiam mais devagar, puderam passar mais tempo reforçando os conceitos que possuíam dificuldades. Além disso, notamos que houve uma diminuição na incidência de retrocessos entre os estudantes, o que pode significar que eles realmente levaram menos dúvidas adiante.

Com relação ao desempenho, descobrimos que as taxas de aprovação e reprovação inicialmente não foram afetadas quando os professores migraram do método tradicional para o *Mastery Learning*. Mas, na última versão do método, a proporção de aprovados conseguiu superar a média de sucesso descrita na literatura [54]. Também percebemos que, à medida que as características da abordagem foram sendo aprimoradas, a taxa de estudantes aprovados por média foi crescendo. Isso pode significar que a quantidade de alunos que conseguiu cobrir e dominar o conteúdo aumentou, à medida que o método foi aprimorado.

Nós também investigamos se os alunos apoiaram a mudança de abordagem. Constatamos que eles ficaram bastante satisfeitos com a utilização do *Mastery Learning* em Programação I. Além disso, descobrimos que a maioria deles prefere cursar disciplinas de programação baseadas no *Mastery Learning*. Vale destacar que as críticas feitas por eles geralmente não eram relacionadas às características da abordagem em si, mas a aspectos da implementação que lhes causaram impactos negativos, como os minitests teóricos em 2013 e as aulas tradicionais em 2014.1. Por fim, conseguimos identificar as vantagens e desvantagens do *Mastery Learning*, sob o ponto de vista dos alunos. Esses resultados contribuíram para a consolidação do método na UFCG, pois permitiram que os professores aprimorassem seus aspectos negativos e reforçassem a continuação dos elementos que trouxeram benefícios à disciplina.

Após uma série de avaliações, percebemos que o *Mastery Learning* está tomando o rumo certo. Embora os professores tenham enfrentado dificuldades nos períodos iniciais, estamos convictos de que as últimas versões do método trouxeram benefícios inquestionáveis tanto para o desempenho, quanto para a rotina de estudo dos alunos. Sejam eles sujeitos que aprendem o conteúdo em pouco tempo, ou estudantes que possuem muitas dificuldades.

Como contribuições deste trabalho estão a avaliação dos efeitos do *Mastery Learning* nas turmas de Programação I da UFCG, as lições aprendidas durante o processo de sua implementação e as métricas criadas para avaliar a dispersão e a rotina de estudo dos alunos. A primeira contribuição ajudou os professores a encontrar evidências sobre a eficácia do *Mastery Learning* a partir de várias perspectivas. A segunda pode ajudar os professores que estejam interessados em implementar o *Mastery Learning* a não passarem pelos mesmos problemas enfrentados na UFCG. Já a última contribuição pode permitir que os pesquisadores avaliem os métodos de ensino a partir da dispersão e da rotina de estudo. Para isso, basta que eles mapeiem os exercícios à sequência de conceitos vistos em sala de aula (e.g. exercícios sobre for, if, while, etc.; ou por unidades, como foi feito na UFCG) para conseguirem coletar quais unidades (ou conceitos) os alunos estudaram em cada semana.

É importante destacar que o escopo dessa pesquisa não garante a generalização dos resultados, mas dá indícios muito fortes que o método atingiu as expectativas dos professores. Como trabalhos futuros, podemos investigar os efeitos do *Mastery Learning* em outras instituições para verificar se os resultados coincidem aos encontrados na UFCG. Nós também planejamos investigar a influência do *Mastery Learning* na aprendizagem dos alunos que pas-

saram pela disciplina. Acreditamos que os reprovados na nova abordagem conseguem reaprender mais facilmente o conteúdo de Programação I e os aprovados assimilam melhor os conceitos de disciplinas correlatas, como Programação II. Outro trabalho futuro interessante seria investigar a relação entre a influência do *Mastery Learning* na rotina de estudo dos alunos e o Pensamento Computacional, já que os sujeitos passaram a estudar de uma forma mais estruturada após a adoção do método.

Por fim, destacamos que os resultados encontrados neste trabalho são animadores, porque nos levam a crer que o método influenciou a rotina de estudo e a aprendizagem dos diferentes tipos de alunos. Temos convicção de que os estudantes que passam pela disciplina saem com uma mentalidade diferente de quando entraram. Com o *Mastery Learning*, os professores estão conseguindo muito mais que boas notas. Estão desenvolvendo melhores alunos.

Bibliografia

- [1] Fisher's exact test of independence. Disponível em: <<http://udel.edu/~mcdonald/statfishers.html>>. Acesso em: 25 nov 2014.
- [2] Teste de mann-whitney. Disponível em: <http://en.wikipedia.org/wiki/Standard_score>. Acesso em: 22 mar 2015.
- [3] Teste de mann-whitney. Disponível em: <http://www.inf.ufsc.br/~verav/Testes_de_Hipoteses/Testes_nao_parametricos_Mann-Whitney.pdf>. Acesso em: 17 dez 2014.
- [4] James D. Allen. Grades as valid measures of academic achievement of classroom learning. *The Clearing House*, 78(5), 2005.
- [5] Danilo T. Alves, Sandro A.V. Souza, Sílvio C.F. Pereira Filho, e Wallace de Sousa Elias. Análise de metodologia baseada no sistema de ensino individualizado de Keller aplicada em um curso introdutório de eletromagnetismo. *Revista Brasileira de Ensino de Física*, 33(1):1–12, 2011.
- [6] Lorin W. Anderson e Beau F. Jones. Designing instructional strategies which facilitate learning for mastery. *Educational psychologist*, 16(3):121–138, 1981.
- [7] Stephen A. Anderson. Synthesis of Research on Mastery Learning. Technical report, 1994.
- [8] Jodi Aronson. A pragmatic view of thematic analysis. Disponível em: <<http://www.nova.edu/ssss/QR/BackIssues/QR2-1/aronson.html?viewType=Print&viewClass=Print>>, 1994. Acesso em: 25 nov 2014.

-
- [9] Jonathan Bergmann e Aaron Sams. *Flip your classroom: reach every student in every class every day*. ISTE, Washington, USA, 2012.
- [10] James H. Block e Robert B. Burns. Mastery Learning. *Review of Research in Education*, 4(1):3–49, 1976.
- [11] Benjamin S. Bloom. Learning for Mastery. *Evaluation comment*, 1(2):1–12, 1968.
- [12] Benjamin S. Bloom. Mastery learning. In James H. Block, editor, *Mastery learning: Theory and practice*. Holt, Rinehart and Winston, New York, 1971.
- [13] Benjamin Samuel Bloom, Committee of College, e University Examiners. *Taxonomy of educational objectives*, volume 1. David McKay New York, 1956.
- [14] BS Bloom. The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher*, 13(6):4–16, 1984.
- [15] Matt Boutell e Curt Clifton. SPLICE: Self-Paced Learning in an Inverted Classroom Environment. In *ACM Technical Symposium on Computer Science Education*, 2011.
- [16] John Carroll. A model of school learning. *The Teachers College Record*, 64(8):723–723, 1963.
- [17] Louis Cohen, Lawrence Manion, e Keith Morrison. Ex post facto research. In *Research Methods in Education*, chapter 12. Routledge, London, 5 edition, 2007.
- [18] Louis Cohen, Lawrence Manion, e Keith Morrison. Sampling. In *Research Methods in Education*, chapter 4. Routledge, London, 5 edition, 2007.
- [19] Louis Cohen, Lawrence Manion, e Keith Morrison. Surveys, longitudinal, cross-sectional and trend studies. In *Research Methods in Education*, chapter 9. Routledge, London, 5 edition, 2007.
- [20] Kathleen Cotton e W. G. Savard. Mastery learning. Technical report, Northwest Regional Educational Lab., Portland, 1982.
- [21] Saeed Dehnadi e Richard Bornat. The camel has two humps. Technical report, Middlesex University, 2006.

- [22] Paul B. Duby. Attributions and Attributional Change: Effects of a Mastery Learning Instructional Approach. Technical report, University of Illinois, Los Angeles, 1981.
- [23] HL Eyre. Keller's Personalized System of Instruction: Was it a Fleeting Fancy or is there a Revival on the Horizon. *The Behavior Analyst Today*, 8(3):317–324, 2007.
- [24] John Farmer, Gerald D Lachter, Joel J Blaustein, e Brett K Cole. The role of proctoring in personalized instruction. *Journal of Applied Behavior Analysis*, 5(4):401–404, 1972.
- [25] KA Foss, SK Foss, Scott Paynton, e Laura Hahn. Increasing college retention with a personalized system of instruction: a case study. *Journal of Case Studies in Education*, 5:1–20, 2013.
- [26] EJ Fox. The Personalized System of Instruction : A Flexible and Effective Approach to Mastery Learning. In Daniel J. Moran e Richard W. Malott, editors, *Evidence-based Educational Methods*, chapter 12, pages 201–221. Elsevier, San Diego, 2004.
- [27] J. Ronald Gentile e James P. Lalley. *Standards and mastery learning: Aligning teaching and assessment so all children can learn*. Corwin Press, 2003.
- [28] LK Grant e RE Spencer. The Personalized System of Instruction : Review and applications to distance education. *The International Review of Research in Open and Distance Learning*, 4(2), 2003.
- [29] Thomas R. Guskey. Defining the Critical Elements of a Mastery Learning Program. Technical Report 4, University of Kentucky, San Francisco, 1986.
- [30] Thomas R. Guskey. Benjamin S. Bloom's Contributions to Curriculum, Instruction, and School Learning. In *American Educational Research Association*, pages 1–23, Seattle, Washington, USA, 2001.
- [31] Thomas R. Guskey. Mastery learning. Technical report, 2001.
- [32] Thomas R. Guskey. Formative Classroom Assessment and Benjamin S. Bloom: Theory, Research, and Implications. In *American Educational Research Association*, number April, pages 1–11, 2005.

- [33] Thomas R. Guskey. Closing Achievement Gaps: Revisiting Benjamin S. Bloom's "Learning for Mastery". *Journal of Advanced Academics*, 19(1):8–31, 2007.
- [34] Thomas R. Guskey. The rest of the story. *Education Leadership*, 65(4):28–35, 2007.
- [35] Thomas R Guskey e Sally L Gates. A synthesis of research on group-based mastery learning programs. 1985.
- [36] Thomas R Guskey e Therese D Pigott. Research on group-based mastery learning programs: A meta-analysis. *The Journal of Educational Research*, pages 197–216, 1988.
- [37] IR Hambleton, WH Foster, e JTE Richardson. Improving student learning using the personalised system of instruction. *Higher Education*, 35(2):187–203, 1998.
- [38] Glenn M. Hymel e Walter E. Dyck. The Internationalization of Bloom's Learning for Mastery: A 25-Year Retrospective-Pro prospective View. In *American Educational Research Association*, pages 1–19, Atlanta, 1993.
- [39] Fred S. Keller. Good-Bye Teacher... *Journal of Applied Behavior Analysis*, 1(1):79–89, 1968.
- [40] Fred S. Keller. Ten years of personalized instruction. *Teaching of Psychology*, 1(1):4–9, 1974.
- [41] B. A. Kilby. Must we give lectures? *Biochemical Education*, 3(4), 1975.
- [42] CLC Kulik, JA Kulik, e RL Bangert-Drowns. Effectiveness of mastery learning programs: A meta-analysis. *Review of educational Research*, 60(2):265–299, 1990.
- [43] JA Kulik, CLC Kulik, e PA Cohen. A meta-analysis of outcome studies of Keller's personalized system of instruction. *American Psychologist*, 34(4):307–318, 1979.
- [44] Andrew Mackie, Brian Usher, e Andrew Martchenko. A new implementation of Keller plan teaching for an undergraduate electronic engineering course. In *Australasian Association for Engineering Education*, Melbourne, 2012.

- [45] John H. McDonald. *Handbook of Biological Statistics*. Sparky House Publishing, Baltimore, Maryland, 3 edition, 2014.
- [46] Hallgeir Nilsen e Even Aby Larsen. Using the personalized system of instruction in an introductory programming course. In *NOKOBIT*, number november, pages 27–38, Nordland, 2011.
- [47] Anthony Robins. Learning edge momentum: a new account of outcomes in CS1. *Computer Science Education*, 20(1):37–71, March 2010.
- [48] Paulo Santana Rocha, Benedito Ferreira, Dionne Monteiro, Danielle da Silva Costa Nunes, e Hugo Cezar do Nascimento Góes. Ensino e Aprendizagem de Programação: Análise da Aplicação de Proposta Metodológica Baseada no Sistema Personalizado de Ensino. *RENOTE*, 8(3):1–11, 2010.
- [49] Wong Bin Sheng e Kang Lifeng. Mastery learning in the context of university education education. *Journal of the NUS Teaching Academy*, 2(4):206–222, 2012.
- [50] J. Gilmour Sherman. Reflections on PSI: good news and bad. *Journal of Applied Behavior Analysis*, 1(1):59–64, 1992.
- [51] Robert E. Slavin. Mastery Learning Reconsidered. Technical Report 7, John Hopkins University, 1987.
- [52] Robert E. Slavin. Mastery learning re-reconsidered. *Review of Educational Research*, pages 300–302, 1990.
- [53] BC Tatum e JC Lenel. A Comparison of Self-Paced and Lecture/Discussion Methods in an Accelerate Learning Format. *Journal of Research in Innovative Teaching*, 5(1):139–156, 2012.
- [54] Christopher Watson e Frederick W.B. Li. Failure rates in introductory programming revisited. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education*, ITiCSE '14, pages 39–44, New York, NY, USA, 2014. ACM.

-
- [55] John B Willett, June JM Yamashita, e Ronald D Anderson. A meta-analysis of instructional systems applied in science teaching. *Journal of research in science teaching*, 20(5):405–417, 1983.

Apêndice A

Questionário sobre a aprovação do Mastery Learning pelos estudantes

Todas as questões pediam que os entrevistados escolhessem, em uma escala de 1 a 7, se eles concordavam (5-7), discordavam (1-3) ou ficariam neutros (4) em relação às afirmativas apresentadas em cada questão.

A.1 Domínio de unidades

du.minitests: Prefiro minitests semanais a provas ocasionais.

du.divisao: A divisão do conteúdo da disciplina em pequenas unidades facilita minha aprendizagem.

du.positivo: Eu acho correto só poder avançar para a próxima unidade quando dominar a atual. Isso faz com que eu aprenda melhor cada conteúdo e não leve dúvidas adiante.

du.negativo: Eu preferiria passar para as próximas unidades, mesmo não tendo dominado a unidade atual.

A.2 Ritmo independente

ri.positivo: A característica de cada aluno progredir em seu próprio ritmo é uma vantagem desta disciplina.

ri.separacao: Sou a favor da separação de alunos por unidades, de acordo com seu ritmo de aprendizagem.

ri.avanco: A disciplina permite que eu avance de unidades rapidamente.

ri.negativo: Todos os alunos deveriam seguir o ritmo da disciplina, ao invés de progredirem mais rápido ou devagar.

A.3 Materiais instrucionais

mat.positivo: Os materiais instrucionais disponibilizados pelos professores são suficientes para eu aprender e passar nas unidades.

mat.livro: O livro recomendado pelos professores é útil em meus estudos.

mat.roteiros: Os roteiros são úteis em meus estudos.

mat.negativo: Existem outros materiais que são mais úteis que o livro e os roteiros disponibilizados pelos professores.

A.4 Feedback

fb.testes: A possibilidade de enviar exercícios e ter o resultado dos testes instantaneamente é ótima para meu aprendizado.

fb.comentarios: Os comentários que os professores fazem a respeito dos erros cometidos nos minitestos são úteis em meu aprendizado.

fb.negativo: Eu recebo feedback, mas nem os professores, nem monitores me ajudam a superar minhas dificuldades.

A.5 Tutoria

tut.positivo: Eu acho importante a disciplina ter tutores que nos acompanham semanalmente.

tut.tutor: O tutor me ajuda a superar minhas dificuldades.

tut.negativo: As reuniões de tutoria/monitoria não contribuem em nada para meu aprendizado.

Anexo A

Plano de Curso de Programação I - 2014.2

Universidade Federal de Campina Grande - UFCG

Departamento de Sistemas e Computação - DSC

Curso de Ciência da Computação - CCC

Plano de Curso Programação 1 2014.2

(Versão 1.0¹)

Dalton Serey, Jorge C. A. de Figueiredo, Wilkerson Andrade e João Arthur B. Monteiro
{dalton, abrant, wilkerson, joao.arthur}@computacao.ufcg.edu.br
Departamento de Sistemas e Computação - DSC

Este documento apresenta o plano de curso das disciplinas introdutórias de programação do Curso de Ciência da Computação da UFCG (CCC) para o segundo semestre de 2014 (2014.2). O curso consiste em duas componentes curriculares (ou disciplinas, como são mais conhecidas pelos alunos): Programação I e Lab. de Programação I. Na prática, as duas disciplinas são tratadas como uma única disciplina, com 8h de aulas semanais, sendo 4h em laboratório e 4h em sala de aula convencional. Assim, todas as atividades realizadas no curso são válidas em ambas para efeitos de avaliação e aprovação.

Introdução

Seja bem-vindo ao curso de Programação I do Curso de Ciência da Computação da UFCG. O único objetivo deste curso é apoiar você no aprendizado de programação de computadores em nível introdutório. No curso adotamos a linguagem de programação Python e o sistema operacional Linux para realizarmos as atividades. A ideia, contudo, não é fazermos um curso técnico sobre estas tecnologias, mas utilizá-las para focar nos conceitos fundamentais e gerais da prática de programação que se aplicam a uma grande variedade de linguagens de programação e/ou sistemas operacionais. Por isso, tanto quanto possível, não utilizaremos facilidades específicas de Python, exceto a título de curiosidade, informação e cultura.

Aprender a programar é bem mais do que acumular conhecimento. Trata-se, principalmente, de desenvolver habilidades e competências — o foco é saber fazer. Por tal motivo, nosso curso é organizado e ministrado usando uma metodologia bastante diferente daquelas às quais você está acostumado. Nosso processo é baseado em três princípios que você deve compreender desde cedo, para melhorar seu desempenho: 1) flipped classroom; 2) avaliação contínua; e 3) self-paced, mastery learning. Vejamos o que cada um desses conceitos quer dizer.

Flipped classroom. Nós invertemos a organização clássica de "assistir a aulas" em sala de aula e "estudar/exercitar" em casa. Nossas principais aulas expositivas foram gravadas em

¹ Baixe a versão mais recente em: <http://bit.ly/prog1planodecurso20141>

pequenos vídeos no Youtube, para que você possa assisti-las de casa e possa repeti-las quantas vezes quiser. Logo, em sala de aula, nós não apresentamos conteúdo. Nosso foco em sala de aula é discutir os tópicos estudados por vocês, estudantes, ajudando-os a eliminar dúvidas e complementar a compreensão.

Avaliação contínua. A avaliação é dita contínua porque ao invés de fazermos três ou quatro provas ao longo do semestre, fazemos avaliações semanais, desde a *primeiríssima* semana do curso. Em compensação, cada avaliação é bem menor, tanto em termos do número de questões, quanto em termos do tempo disponível para resolver as questões. Além disso, nossa avaliação é feita em horário normal de aulas, sem a necessidade de calendário. Assim, lembre-se **toda sexta-feira é dia de avaliação**².

Self-paced e Mastery Learning. O estudo de programação depende acima de tudo de dedicação e motivação, que variam de aluno para aluno e também ao longo do semestre, pelos mais variados motivos. Consequentemente, cada estudante tem sua própria velocidade em cada momento do curso para absorver o conteúdo e avançar no domínio (*mastery*) das habilidades exigidas. Por isso, permitimos que cada estudante determine e avance em sua própria velocidade. O importante é que até o final do semestre o estudante tenha conseguido demonstrar que tem domínio do conteúdo mínimo exigido e que tenha cumprido com as obrigações formais da disciplina (presença, entrega de projetos, etc).

Self-paced vs Avaliação Contínua. Pode parecer contraditório dizer que fazemos avaliação contínua e que permitimos ao estudante avançar em sua própria velocidade. Não há, contudo, qualquer inconsistência. A razão é que a cada semana, disponibilizamos questões para avaliar você e os demais estudantes, independente do estágio do curso em que se encontrem. Se você não conseguir resolver os problemas referentes a uma determinada unidade em uma semana, você será submetido a problemas da mesma unidade na semana seguinte. E isso se repetirá tantas vezes, quanto for necessário até que você consiga demonstrar que dominou os conceitos e as habilidades referentes àquela unidade. Além disso, como a avaliação funcional dos problemas é feita automaticamente, você sabe imediatamente se está ou não aprovado na unidade. Além disso, se você for aprovado em uma unidade, serão liberadas imediatamente as questões da unidade seguinte. Observe que este método permite que você recupere tempo atrasado em semanas anteriores ou até mesmo que você se adiante, mediante estudo independente.

Proficiência. O curso é focado no conceito de proficiência. O que fazemos é avaliar continuamente se consideramos você é minimamente proficiente na linguagem, no ferramental adotado e, indiretamente, nos conceitos de programação. Desde a primeira semana de avaliação, disponibilizaremos questões de todas as unidades. Isso significa que estudantes que já sejam proficientes em programação, podem demonstrá-lo desde as primeiras semanas. Tão logo um estudante seja considerado proficiente (desde que em tempo hábil), ele passará à fase de projeto da disciplina.

Avaliação das questões. A avaliação das questões de programação é feita por uma

² Excepcionalmente, em algumas semanas, pode ser necessário fazer avaliação em outros dias e horários da semana. Contudo, avisaremos com antecedência nesses casos.

combinação de trabalho automatizado e de inspeção manual. O julgamento de **corretude** (ou funcional) é feito de forma 100% automatizada. Isso permite que você saiba imediatamente, ao submeter a questão, se está ou não aprovado nela e que já tenha liberadas novas questões (possivelmente, até de outras unidades). Os programas submetidos, contudo, serão ainda avaliados de forma manual pelos professores. Contudo, por depender de avaliação humana que é mais demorada, essa avaliação será feita com menor periodicidade e não será levada em consideração para efeitos de progresso nas unidades do curso. A idéia é proporcionar a você uma opinião do professor sobre a qualidade do código que você produz. Veja a seção avaliação mais adiante.

Observe que os conhecimentos, habilidades e competências que abordaremos são fundamentais para o restante do curso de Ciência da Computação. Logo, parte significativa e que levamos a sério de nosso trabalho é avaliar e julgar o quanto você tem esses conhecimentos, habilidades e competências, para decidirmos se você está apto a seguir adiante no curso.

Metodologia

Atividades. Cursar esta disciplina significa que você deve participar ativamente de 5 tipos de atividades: 1) assistir às vídeo-aulas que cobrem o conteúdo do curso (atividade que deve ser feita em casa e/ou nos laboratórios da UFCG, de preferência fora dos horários dedicados às aulas); 2) ler e estudar o material bibliográfico recomendado; 3) responder aos exercícios selecionados no sistema online que usamos no curso (os exercícios são de tipos variados, tais como *quizzes*, compreensão de programas, compreensão de problemas, exercícios de programação, mini-projetos, etc); 4) participar das discussões e das aulas demonstrativas e/ou interagir com os professores ou monitores a fim de resolver dúvidas e esclarecer aspectos em que tenha dificuldades; e 5) participar das avaliações semanais para avaliar seu progresso no curso.

Como mencionado antes, o curso é baseado em um método de ensino e aprendizagem conhecido como *mastery learning*. Nesse método, o conteúdo (conhecimento + habilidades) do curso é organizado em unidades que o estudante deve abordar sequencialmente, progredindo entre unidades **somente** quando demonstrar ter dominado o conteúdo e as habilidades da unidade anterior. Para avaliar se o estudante domina o conhecimento e as habilidades requeridas em cada unidade, usamos minitestos semanais. É importante enfatizar: o estudante não poderá avançar para a unidade N, enquanto não for aprovado na unidade N-1. E para ser aprovado em uma unidade, deve ser capaz de resolver pelo menos dois problemas diferentes da unidade.

Local das atividades. Nos diferentes horários reservados ao longo da semana, contudo, parte das atividades (ou aulas) são realizadas em sala de aula convencional e parte em laboratório (especificamente, nos LCC1 e LCC2).

As atividades teóricas consistem em discussões e debates e, em menor número, aulas demonstrativas. Nelas, focamos nos aspectos teóricos de programação e de linguagens de programação. As atividades práticas consistem tipicamente em exercícios a serem feitos individualmente e/ou em pequenos grupos de estudantes, sob supervisão dos professores e/ou monitores. Nessas atividades, além de desenvolver o conhecimento e a habilidade de programar em si, esperamos que o estudante desenvolva proficiência com os aspectos práticos de programação — ferramentas de desenvolvimento, de testes, sistema operacional, estratégias de codificação, etc. As atividades práticas incluem roteiros experimentais, exercícios de programação e pequenos projetos. Parte das aulas práticas da semana também serão usadas para a aplicação de minitests que são a base para a **avaliação continuada** dos estudantes (mais sobre isto adiante). É importante ressaltar que as habilidades e os conteúdos abordados nos dois tipos de atividade são estritamente sincronizados e formam um único corpo de conhecimento. Por isso, no restante deste documento, tratamos das duas componentes curriculares como sendo um único curso.

Objetivo do curso. O objetivo do curso é capacitar o estudante a programar computadores em nível considerado introdutório de acordo com o projeto pedagógico do CCC/UFCG. No curso, adotamos a linguagem de programação Python³. A disciplina foca nos conceitos básicos e mais gerais de programação, tais como fundamentos de algoritmos, técnicas e estruturas de dados fundamentais e resolução de pequenos problemas de programação. Ao final do curso, o estudante deve ser capaz de resolver problemas cujas soluções requeiram a escrita de pequenos programas, tipicamente, com dimensões não excedendo uma centena de linhas de código, e que utilizem algoritmos e estruturas de dados fundamentais. Também se espera que o estudante domine as várias tecnologias relacionadas à programação de computadores, tais como operação básica do sistema operacional e ferramentas de suporte à programação⁴. Na disciplina **não focamos** aspectos avançados e/ou específicos da linguagem, nem mesmo nos conceitos de orientação a objetos (tema que é abordado no segundo semestre, no CCC).

Conteúdo Programático

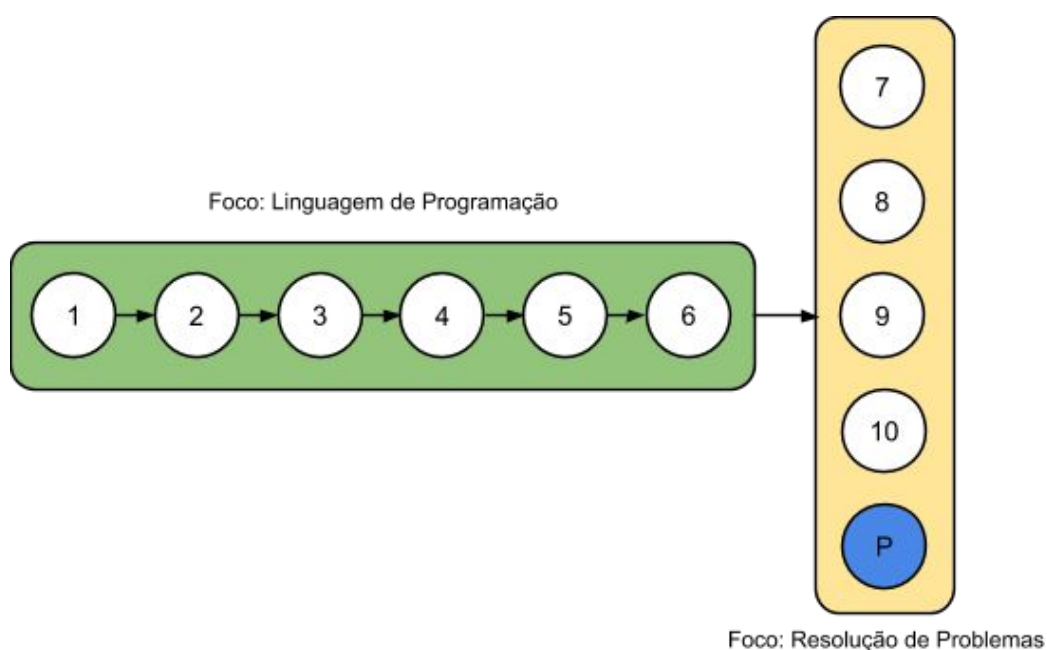
O conteúdo do curso foi organizado em 10 unidades, agrupadas em duas partes. A parte 1, que engloba as seis primeiras unidades, reúne os conceitos fundamentais de programação e de linguagens de programação. Nele, esperamos que o estudante desenvolva as habilidades para escrever pequenos programas que lidam com padrões simples de entrada e saída, tipos básicos da linguagem (numéricos, strings, booleans e listas simples), alternativas, condicionais, laços e funções. A parte 2 do curso, formada pelas unidades de 7 a 10 e pelo projeto, aprofunda o uso dos conceitos de linguagem vistos na parte 1, através do estudo de algoritmos e estruturas de dados um pouco mais avançados (tuplas, listas, conjuntos, matrizes, mapas e conjuntos). Na parte 2 também esperamos que o estudante desenvolva mais conscientemente suas habilidades e técnicas de compreensão e resolução

³ Mais especificamente, usaremos Python 2.7.x (<http://www.python.org/download/releases/2.7.5/>).

⁴ No curso adotamos especificamente o sistema operacional Linux. Embora em casa o estudante possa utilizar outros sistemas operacionais, é importante, para sua formação em Ciência da Computação, que domine a operação de sistemas da família *nix e o respectivo ferramental de programação.

de problemas. Finalmente, a parte 2 inclui um pequeno projeto de desenvolvimento, em que o estudante terá a oportunidade de desenvolver um programa de forma mais livre e em equipe.

As dependências entre as unidades, para efeito de progresso no curso, são mostradas na figura a seguir. Cada unidade é representada como um nó do grafo. As caixas indicam as partes 1 e 2 do curso. Setas representam as dependências entre as unidades/partes. Observe que entre as unidades de 7 a 10, bem como para o projeto, não há dependências obrigatórias. O conteúdo e as habilidades abordadas em cada unidade são apresentados a seguir.



Ambientação -

Unidade 1 - Conceitos elementares de programação. Ao final da unidade, o estudante deve entender como funcionam programas simples com fluxo linear (sem laços e/ou condições) que utilizem comandos de entrada e saída de dados, manipulação básica de dados com expressões de tipos numéricos e strings. O estudante também deve compreender o conceito de variável. Espera-se ao final da unidade 1 que seja capaz de **modificar** ou **complementar** programas dessa natureza, atendendo a especificações dadas, tais como mudança/adição/correção das expressões, bem como dos dados coletados e/ou impressos na saída. Conceitos de linguagem de programação abordados: conceito de tipo de dado, tipos numéricos (floats e inteiros), tipo string, variáveis, expressões e avaliação de expressões, invocação de funções matemáticas, formatação básica de strings, comandos de entrada e de saída.

Unidade 2 - Escrevendo programas simples. Ao final da unidade 2, o estudante deve ser capaz de escrever programas simples com fluxo linear, a partir de especificações e/ou

problemas dados. O estudante deve compreender os conceitos de qualidade funcional e não-funcional de programas. O estudante será apresentado ao ferramental de teste fornecido pela disciplina que será usado ao longo do curso e deve conhecer os critérios de qualidade de código que serão usados para a avaliação dos programas. Conceitos de linguagem abordados: os programas/problemas devem explorar e reforçar os mesmos conceitos de linguagem vistos na unidade 1.

Unidade 3 - Condições, alternativas e funções. Ao final da unidade, o estudante deve ser capaz de escrever programas que, além das características vistas até a unidade anterior, envolvam caminhos condicionais e/ou alternativas (simples e múltiplas), atendendo a especificações e/ou problemas dados. Conceitos de linguagem abordados: tipo boolean, expressões lógicas, `exit()`, comandos condicionais (`if`, `else`, `elif`), introdução a funções, uso e definições de funções simples (sem efeito colateral).

Unidade 4 - Laços definidos. Ao final da unidade, o estudante deve ser capaz de escrever programas que, além das características vistas até a unidade anterior, requeiram iterações simples sobre sequências de dados, tais como strings e listas. Em termos de algoritmos será apresentado os padrões fundamentais utilizados em laços simples: acumulador, mínimo/máximo. Conceitos de linguagem abordados: strings como sequências, listas, indexação de sequências, operador `in`, laços `for`, função `range`, iteradores (laços sobre valores) versus laços sobre índices.

Unidade 5 - Laços indefinidos. Ao final da unidade, o estudante deve ser capaz de escrever programas que requerem laços indefinidos e/ou definidos. Na unidade focamos nos diferentes tipos de laços indefinidos que podem ser necessários em diferentes situações. Em particular, diferenciamos os laços pelo ponto de saída do laço: `while`, `do..while` (e `repeat..until`), e `loop leave`. E veremos como cada tipo de laço pode ser implementado em linguagens de programação, em particular em Python. Conceitos de linguagem de programação: comando `while`, `break` e `continue`. Busca linear com `while` e com `for`.

Unidade 6 - Funções. Ao final da unidade, o estudante deve ser capaz de projetar, escrever e usar pequenas funções em seus programas e/ou escrever funções simples para uma dada especificação (assinatura e semântica). O foco será a compreensão da semântica de funções, bem como abstração funcional como técnica de design. A sintaxe e a semântica devem ser expressas como uma combinação de texto e de casos de teste (`asserts`). Conceitos de linguagem: invocação de funções, definição de funções, parâmetros formais, parâmetros reais, retorno, pilha de execução, semântica de execução de funções, escopo, funções com efeito colateral vs. funções puras, aliasing, stack frames.

Unidade 7 - Algoritmos com laços. Ao final da unidade, o estudante deve ser capaz de escrever programas que requerem laços mais complexos que os abordados nas unidades anteriores. Os problemas abordados nesta unidade envolvem formas básicas de manipulação de listas. Em particular, serão abordados os padrões inserção ordenada, merge e algoritmos básicos de ordenação (`bubble sort`, `insertion sort`, `selection sort`).

Unidade 8 - Estruturas de dados: sequências. Ao final desta unidade, o estudante deve ser capaz de escrever programas e/ou funções que requerem o uso e a manipulação (construção, extensão, redução, decomposição, etc) de estruturas de dados lineares, tais como listas, tuplas e strings. Conceitos de linguagem de programação: slices, métodos e construtores de listas e tuplas (+, append, insert, pop, remove, etc).

Unidade 9 - Estruturas de dados: matrizes. Ao final desta unidade, o estudante deve ser capaz de escrever programas e/ou funções que requerem o uso e a manipulação de dados em matrizes, representadas na forma de lista de listas.

Unidade 10 - Estruturas de dados: mapas. Ao final desta unidade, o estudante deve ser capaz de escrever programas e/ou funções que requerem o uso e a manipulação de dados em mapas/dicionários. Conceito de linguagem: sintaxe e semântica de mapas.

Avaliação

A avaliação será baseada no progresso do aluno nas unidades, o desempenho obtido nas avaliações automáticas (funcional) e qualitativa dos minitests e o projeto final. Os pesos de cada componentes são dados pela tabela abaixo.

Minitests	60% (*)
Avaliação Qualitativa	20%
Projeto ou Prova	20%
TOTAL	110% (o ponto a mais será descartado, naturalmente)

(*) Estudamos modificar estes percentuais, mas decidimos não alterar as regras inicialmente definidas. Assim, os percentuais serão os mesmos que foram anunciados desde o início do semestre.

Progresso nas unidades. Para ser aprovado em uma unidade e progredir para a seguinte, exigimos conhecimento em nível “satisfatório” (equivalente a uma nota 7,0 na unidade). Para ser habilitado ao projeto, contudo, o estudante deve ser aprovado na parte 1 do curso, até a data limite indicada no calendário do curso.

Condições de aprovação no curso. Formalmente, como exige a regulamentação da graduação na UFCG, o estudante deve obter média parcial acima de 7,0 (sete) para ser aprovado por média ou média final acima de 5,0 (cinco) — a média final é calculada com base na média parcial e uma nota obtida no exame final da disciplina. Para ter acesso ao exame final, o estudante deve obter média parcial igual ou superior a 4,0 (quatro).

Minitests e progresso nas unidades. Para cada unidade, em geral, o aluno deve ser aprovado em dois minitests. Para ser aprovado em uma unidade do curso e progredir para a seguinte, o estudante deve obter nota maior ou igual a 7,0 (sete) na unidade. A nota da unidade é calculada como a média das duas maiores notas obtidas pelo aluno nos

minitests práticos da unidade.

Exemplo de progresso. Suponha que um estudante fez dois minitests da unidade 1 e obteve as notas 7,0 e 4,0, nas duas semanas, respectivamente. Sua nota na unidade será, portanto, igual $(7,0 + 4,0) / 2,0 = 5,5$. E, portanto, não se encontrará apto a progredir para a unidade seguinte. Deverá, portanto, permanecer estudando o material da unidade 1 e, na semana seguinte, se submeter a um terceiro miniteste da mesma unidade. Suponha agora que no terceiro miniteste, o estudante obteve a nota 8,0 (oito). Pela regra apresentada, a nota evoluirá para $(8,0 + 7,0) / 2,0 - 0,5 = 7,0$. Com essa nota, o estudante estará aprovado na unidade 1 e autorizado a progredir para a unidade 2.

Bibliografia e Material Didático

Para estudar Python, há uma enorme diversidade de material bibliográfico. Em particular, muitos bons livros são disponibilizados publicamente na Internet em PDF, usando *copyleft*. Para o curso, contudo, desenvolvemos nosso próprio material, também em PDF e disponível na Internet. Além de nosso material e dos livros, há uma enorme disponibilidade de material adicional na Internet, tais como sites, blogs, vídeo aulas, etc. Aqui listamos apenas os mais relevantes deles.

Livros

- Think Python, de Allen Downey (<http://www.greenteapress.com/thinkpython/>). Há uma [versão em português do livro](#). Observe, contudo, que é uma tradução de uma versão já antiga.
- Python Programming: An Introduction to Computer Science, de John Zelle (<http://mcsp.wartburg.edu/zelle/python/>). Alguns capítulos deste livro são excelentes na explicação dos conceitos de programação.
- Dive into Python (<http://www.diveintopython.net/>). Este é um livro para quem já programa e quer aprender Python. Talvez não seja o caso para vários dos nossos alunos. Está aqui apenas, para registro.