

Departamento de Ciências e Tecnologias de Informação

**Aprendizagem na programação: um modelo de
continuidade de aprendizagem de programação.**

Fábio Miguel Matos Ferreira

Dissertação submetida como requisito parcial para obtenção do grau de
Mestre em Software de Código Aberto

Orientador:

Doutor Carlos J. Costa, Professor Auxiliar do DCTI ISCTE-IUL

Co-orientadora

Mestre Manuela Aparício, Docente Convidado do DCTI

Outubro, 2015

Agradecimentos

Aos meus orientadores, Professor Doutor Carlos J. Costa e Professora Mestre Manuela Aparício, pela prontidão e disponibilidade que demonstraram durante todo o tempo. Um muito obrigado pela ajuda e pelo apoio, que foi constante e que se tornou fundamental para a conclusão deste trabalho.

Aos meus colegas e amigos pela ajuda e paciência que tiveram ao longo do trabalho. E a todos os outros que escuso de mencionar.

E em especial à minha namorada Vânia Varela que sempre esteve presente para me apoiar e dar força nos melhores e piores momentos. Pelo encorajamento e por me ajudar a manter sempre focado na conclusão deste trabalho.

Resumo

Existe atualmente uma enorme dificuldade em aprender programação. Neste contexto foi realizada a presente dissertação que com base em revisão de literatura sobre os fatores dificultadores da aprendizagem de programação, nos paradigmas da programação, propõe um modelo teórico. Apresenta aqui também um estudo de estado de arte sobre o ensino da programação em Portugal fazendo um levantamento das universidades que lecionam cursos de informática. E aqui proposto e validado empiricamente um modelo que explica o nível de satisfação dos alunos, o nível de utilidade percebida e a intenção de continuar a estudar. Este modelo valida os fatores determinantes que levam a intenção de continuidade de aprender programação.

Palavras-chave

Educação, ensino da programação, modelo teórico, intenção, utilidade, satisfação

Abstract

There is currently a great difficulty in learning programming. In this context this master thesis presents a literature review on the programming difficulties, programming paradigms. Here it also presented the state of the art in computer science education in Portuguese universities. Based on this literature review it is proposed here a theoretical model. This theoretical model was proposed and validated empirically with university students of computer science. This model explains the determinants of satisfaction level of students. It also explains the usefulness level perceived by students. The proposed and validated model presents the drivers of students' continuance intention in programming learning.

Keywords

Computer science education, computer programming education, theoretical model, intention, usefulness, satisfaction

Índice

1	Introdução.....	4
1.1	Definição do problema e objetivos.....	4
1.2	Objetivos de investigação.....	4
1.3	Abordagem metodologia.....	5
1.4	Estrutura da dissertação.....	6
2	Revisão da Literatura	7
2.1	Definição conceito informática (Computer Science).....	7
2.2	Dificuldades na aprendizagem da programação.....	7
2.2.1	Falta de problemas práticos.....	8
2.2.1.1	Problemas orientados para o ambiente industrial.....	8
2.2.2	Falta de ginástica mental.....	9
2.2.3	Tempo vs. Programação.....	9
2.2.4	Motivação Social.....	10
2.2.5	Idade vs. Género vs. Programação.....	10
2.2.6	Dificuldade em dominar conceitos.....	12
2.2.7	Falta de teste – Simulação.....	12
2.3	Paradigmas de programação.....	13
2.4	Plano curricular dos cursos de Informática.....	19
2.4.1	Planos curriculares nacionais.....	19
2.4.2	Computer Science Curricula 2013 (CS2013).....	21
2.5	Fatores de Continuidade e confirmação na aprendizagem.....	22
2.5.1	Aplicação do modelo de estudo.....	22
2.6	Síntese da revisão da literatura.....	23
3	Proposta do modelo de investigação.....	24
3.1	Definição dos constructos adaptados à aprendizagem da programação.....	24
3.1.1	Proposta das hipóteses.....	25
3.2	Modelo de investigação.....	28
4	Análise de dados e resultados.....	33
4.1	Análise da amostra.....	33
4.1.1	Caracterização da amostra.....	34
4.2	Resultados da Avaliação do modelo de medida.....	37

4.3	Avaliação do Modelo Estrutural	40
5	Discussão de Resultados	43
6	Conclusões	46
6.1	Conclusões	46
7	Referências Bibliográficas	48
	Anexos.....	52

Índice de Tabelas

Tabela 1: Abordagem Metodológica.....	5
Tabela 2: Comparação das respostas dos estudantes por idade e género (Carter & Jenkins, 1999, p. 2).....	11
Tabela 3: Linguagens vs Paradigmas	18
Tabela 4: Linguagens de faculdades portuguesas	20
Tabela 5: Definição dos constructos	24
Tabela 6: Perguntas realizadas aos alunos	30
Tabela 7: Definição da dimensão mínima da amostra Cohen (1992).	33
Tabela 8: Amostra - Género e idades	34
Tabela 9: Amostra - Região de proveniência	34
Tabela 10: Amostra - Região do curso.....	35
Tabela 11: Amostra - Ano que frequenta	35
Tabela 12: Amostra - Escolaridade da Mãe	36
Tabela 13: Amostra - Escolaridade do Pai	36
Tabela 14: Amostra - Escolaridade alunos.....	36
Tabela 15: Amostra - Linguagens de programação.....	37
Tabela 16: Caracterização da amostra.....	37
Tabela 17: Análise de Cross-Loadings.....	39
Tabela 18: Análise do Critério de Fornell e Larcker.....	40
Tabela 19: Avaliação do modelo estrutural.....	42

Índice de Figuras

Figura 1: Linguagens C – Hello World.....	18
Figura 2: Linguagens Java – Hello World	18
Figura 3: Linguagens JavaScript – Hello World.....	19
Figura 4: Modelo de Investigação	29
Figura 5: Resultados do Modelo de Investigação	41

Abreviaturas

ACM	Association for Computing Machinery
EETC	Licenciatura em Engenharia Eletrónica e Telecomunicações e de Computadores
EI	Licenciatura em Engenharia Informática
EIC	Licenciatura em Engenharia Informática e de Computadores
ETI	Licenciatura em Engenharia de Telecomunicações e Informática
FCUL	Faculdade de Ciências da Universidade de Lisboa
IEEE	Institute of Electrical and Electronics Engineers
IGE	Licenciatura em Informática e Gestão de Empresas
ISCTE-IUL	ISCTE - Instituto Universitário de Lisboa
ISEL	Instituto Superior de Engenharia de Lisboa
IST	Instituto Superior de Tecnologias
POO	Programação Orientada a Objetos
UBI	Universidade da Beira Interior
IW	Informática Web
UM	Universidade do Minho
MIETI	Mestrado Integrado em Engenharia de Telecomunicações e Informática
MIEGSI	Mestrado Integrado em Engenharia e Gestão de Sistemas de Informação
UA	Universidade de Aveiro
UE	Universidade de Évora
UP	Universidade do Porto
UC	Universidade de Coimbra
UL	Universidade de Lisboa
UAlg	Universidade Algarve

UAçores	Universidade Açores
CS2013	Computer Science Curricula 2013
IRM	Informática - Redes e Multimédia
ECT	Escola de Ciências e Tecnologia
FE	Faculdade de Engenharia
ECT	Escola de Ciências e Tecnologia
FCT	Faculdade de Ciências e Tecnologia

1 Introdução

Muitos são os fatores influenciadores na hora de escolher o curso superior direcionado para a programação, como a opinião familiar, o suporte acadêmico oferecido pela universidade, a reputação da universidade também a expectativa criada pelo novo futuro aluno (Bhattacharjee, 2001; Kopanidis, 2008).

Os estudos existentes na área da dificuldade na aprendizagem na programação são a falta de problemas práticos, a falta de ginástica mental, o tempo vs. programação, a motivação social e a idade vs género vs programação que se aplica na dificuldade de dominar conceitos, entre outros (Costa, Aparício, & Cordeiro, 2012; Gomes & Mendes, 2007; Murphy et al., 2006).

O que pretendo estudar nesta investigação são os fatores que influenciam a escolha de um curso superior orientado para a aprendizagem da programação.

1.1 Definição do problema e objetivos

No âmbito deste trabalho, é proposto, um estudo de investigação cuja Questão de Investigação é a que se segue:

Que fatores influenciam a escolha de um curso superior orientado para a aprendizagem da programação?

Com este problema pretende-se abordar o Ensino da Programação no ensino superior.

1.2 Objetivos de investigação

A partir da definição do problema, numa primeira fase, os objetivos de programação que pretendo com este trabalho são os seguintes:

- Identificação dos problemas de aprendizagem da programação
- Identificação das universidades que ensinam programação e a sua estrutura curricular;
 - Conhecer as suas linguagens lecionadas, nas disciplinas de programação no ensino superior

- Comparação entre a CS-2013 e a estrutura curricular das Universidades
- Estudos realizados no ensino da programação (Género, Cultural);
 - Possível influência da dificuldade ou facilidade da aprendizagem dos conceitos de programação relacionados com o género e a cultura dos alunos que estão a realizar as unidades curriculares.
- Identificação dos fatores de escolha dos cursos de programação
- Desenvolvimento de um modelo teórico de continuidade na aprendizagem da programação
- Validação do modelo

1.3 Abordagem metodologia

A metodologia seguida para o desenvolvimento da dissertação, foi iniciada pela revisão da literatura sobre Ensino da Programação no ensino superior. Posteriormente elaborei a relação entre a programação que é lecionada e o ranking em que se encontram as universidades. Para poder complementar toda a informação que obtive anteriormente fiz uma pesquisa dos Estudos realizados no ensino da programação, tendo como foco o Género dos alunos e as suas culturas.

Tabela 1: Abordagem Metodológica

Abordagem Metodológica	Revisão de Literatura	Dificuldades na aprendizagem da programação.
		Tipos de linguagem da programação.
		Planos Curriculares
		Fatores de permanência nos cursos de programação.
	Modelo	Proposta de um Modelo de continuidade da aprendizagem de programação.
Trabalho Empírico	Validade do Modelo Proposto <ul style="list-style-type: none"> • Modelo de Medida • Modelo Estrutural 	

1.4 Estrutura da dissertação

Muitos são os fatores influenciadores na hora de escolher o curso superior direcionado para a programação, como a opinião familiar, o suporte acadêmico oferecido pela universidade, a reputação da universidade e também a expectativa criada pelo novo futuro aluno (Bhattacharjee, 2001; Kopanidis, 2008).

Questão de Investigação que será tratada nesta tese é - Que fatores influenciam a escolha de um curso superior orientado para a aprendizagem da programação?

O presente trabalho teve como principal ordem a identificação dos problemas de aprendizagem da programação, a identificação das universidades que ensinam programação e a sua estrutura curricular, a comparação entre a CS2013 e a estrutura curricular das Universidades, os estudos realizados no ensino da programação (Género, Cultural) e a identificação dos fatores de escolha dos cursos de programação (ACM/IEEE, 2013).

A abordagem metodológica está dividida em revisão da literatura, modelo e trabalho empírico. A revisão da literatura subdivide-se em dificuldades na aprendizagem de programação (Costa et al., 2012), tipos de linguagem da programação, planos curriculares (ACM/IEEE, 2013) e fatores de permanência nos cursos de programação (Davis, 1993; Fishbein & Ajzen, 1991; Rogers, 1983). É proposto um modelo de continuidade da aprendizagem de programação. O trabalho empírico divide-se em modelo de medida e modelo estrutural.

2 Revisão da Literatura

Para poder ter uma visão mais global, do ensino da programação em Portugal, baseei a minha pesquisa em seis aspetos. Primariamente foi analisado o plano curricular dos cursos de engenharia e os tipos de linguagens ensinadas. Seguidamente foram analisadas as problemáticas da falta de problemas práticos, falta de ginástica mental. Para concluir foram feitas algumas comparações tempo vs. programação e idade vs. género vs. programação.

2.1 Definição conceito informática (Computer Science)

No ano de 1980 o conceito informático tinha como principais atividades o processamento numérico, processamento de símbolos e o processamento de dados.

Com o evoluir dos tempos e das necessidades o foco situou-se em incluir a coordenação e a comunicação, ou seja, um processamento de informações e caracterizações teóricas das propriedades e limitações.

A necessidade de fiabilizar a informática e os computadores levou à necessidade de dar importância a questões como transparência, usabilidade, confiabilidade do hardware e software e a sua segurança.

A informática, conseguiu juntar numa só ferramenta (computador) a capacidade de desenvolver e criar novas evoluções em vários ramos do trabalho, sociedade e ciências. Exclui-se aqui a informática na ótica do utilizador (Denning, 2003).

2.2 Dificuldades na aprendizagem da programação

As dificuldades na aprendizagem da programação são a falta de problemas práticos, a falta de ginástica mental, o tempo dedicado á programação e também a disparidade presente nas idades em comparação entre géneros na programação.

2.2.1 Falta de problemas práticos

No ensino da programação existem falhas. Dessas falhas uma delas é a falta de problemas práticos a serem desenvolvidos ou resolvidos pelos novos aprendizes.

A necessidade de apresentar problemas sólidos e práticos prende-se com o fato de enfrentar novos desafios. É necessário que o aluno enfrente novos desafios práticos para que retire deles a aprendizagem que virá a ser útil no futuro. Os novos alunos devem, durante a aprendizagem, reunir um sem-número de novas ferramentas para o futuro, pois mais tarde, estas ser-lhe-ão úteis. Para que possam reunir maior número de ferramentas têm de ter um ensino mais vocacionado para a aplicação de conhecimentos e não apenas para a armazenagem dos mesmos. Durante a aprendizagem, os exemplos apresentados são muitas vezes superficiais. São pouco direcionados para casos práticos (Costa, Aparício, & Cordeiro, 2012).

2.2.1.1 Problemas orientados para o ambiente industrial

Durante o ensino escolar de programação os alunos são orientados para problemas de ambiente industrial, o que torna a sua visão para a resolução de outras situações redutora.

O fato de no ensino, a forma de resolução dos problemas ter sempre a mesma base ou o mesmo método acaba por cingir os alunos a apenas uma forma de resolução não lhes dando bases para que se iniciem noutros problemas, ficando assim limitados ao conhecimento tecnológico adquirido academicamente. Para além de não desenvolverem bases sólidas de resolução dos problemas, os alunos que completaram o curso irão enfrentar já na vida laboral dificuldades significativas de integração devido a falta de confiança nas bases que fomentaram a nível académico (Gomes, Henriques, & Mendes, 2008).

2.2.2 Falta de ginástica mental

Uma das falhas presentes nos novos alunos é a falta de ginástica mental. É necessário inculcar aos professores a necessidade de terem de ter tato para com os alunos e as suas diferentes formas de ver as diversas soluções na programação. É necessário não querer traçar um trilho obrigatório na programação para que possam expandir os vários pontos de vista sobre o desenvolvimento de código. Se por um lado é necessário não traçar um trilho, por outro lado é necessário, em alguns casos, colocar os alunos no início do trilho, para que entendam que é preciso encontrar novas formas de solucionar os problemas. Tornando assim alunos inaptos em alunos ativos a encontrar soluções novas, rápidas e eficazes. É ginasticar o cérebro e a forma de trabalhar dos novos alunos para seja que é possível evoluir na programação. Para que tudo seja possível os alunos, em parte, necessitam de aptidão para programar. É necessária capacidade de observação dos problemas que vão surgindo e proatividade para os solucionar, o que é difícil encontrar nos novos alunos (Costa et al., 2012; Gomes & Mendes, 2007).

2.2.3 Tempo vs. Programação

Há fatores importantes para o ensino da programação, um deles é – O tempo. O tempo por vezes pode ser um inimigo para quem ensina e para quem está a aprender. No caso da programação não é exceção. Para quem ensina o tempo parece imenso – em alguns casos – enquanto, para quem está a aprender esse tempo é curto – sempre e sem exceção (Costa et al., 2012; Gomes & Mendes, 2007).

Os planos curriculares com tempo pré-estabelecido, dificultam por vezes a aprendizagem dos alunos sobre a matéria. Não é necessário acelerar a absorção de conhecimento, pois cada aluno tem o seu ritmo. Esse tipo de aprendizagem, com tempo pré-estabelecido, só premeia os alunos mais predispostos para a programação, dificultando a progressão dos restantes. Existe o fator intrínseco a cada um mas é importante realçar que não é possível evoluir e obter bons resultados na programação caso o aluno não dedique tempo ao estudo e à aplicação em casa das matérias dadas em aula. A aprendizagem da programação requer tempo e dedicação o que

vários alunos não dedicam, dificultando os bons resultados (Costa et al., 2012; Gomes & Mendes, 2007).

2.2.4 Motivação Social

Socialmente a profissão de programador não é muito aclamada. É considerada uma profissão a anos-luz de uma qualquer outra profissão mais recorrente. O que lhe confere essa diferença é o fato de ter conceitos não tangíveis a grande parte da população (Gomes et al., 2008).

Os programadores têm de ter a capacidade de ver o abstrato o que não se torna fácil. O seu trabalho assenta em diversos conceitos que tem de ser perfeitamente capaz de dominar na totalidade de forma a desenvolver o seu trabalho com a maior rapidez e perfeccionismo possível.

A verdade é que por terem uma conotação não muito aclamada, retira aos novos programadores a vontade de fazer parte desse grupo de profissionais que todos os dias tem de aprofundar mais conhecimentos de forma a estar sempre de acordo com as evoluções tecnológicas. Faltando assim o apoio social de que necessitam para se sentirem motivados a aprofundar conhecimentos na programação (Gomes et al., 2008).

2.2.5 Idade vs. Género vs. Programação

Na tabela que se segue, pode visualizar-se o resultado e um estudo realizado nas universidades de Kent e Leeds. Como *sample* utilizaram sete questões de programação. Os resultados das questões foram organizados tendo em conta a idade e o género dos alunos. Como resultado pode observar-se que antes dos 21 anos a capacidade para a programação feita por homens é acrescida enquanto para as mulheres continua a ser mais diminuta.

Tabela 2: Comparação das respostas dos estudantes por idade e género (Carter & Jenkins, 1999, p. 2)

	Male>21		Male<21		Female>21		Female<21		All	
	Kent	Leeds	Kent	Leeds	Kent	Leeds	Kent	Leeds	Kent	Leeds
Sample	8	7	50	29	8	1	6	19	72	56
Q1	2.8	2.4	2.8	2.5	1.5	1.0	1.3	1.8	2.5	2.2
Q2	3.5	2.9	2.9	2.5	2.5	3.0	2.0	2.3	2.8	2.5
Q3	3.0	2.0	2.9	2.6	3.8	2.0	2.7	2.4	3.0	2.4
Q4	3.5	4.7	4.1	4.0	4.0	2.0	4.7	4.4	4.1	4.2
Q5	4.0	3.0	3.7	3.3	3.0	3.0	3.3	2.8	3.6	3.1
Q6	2.0	2.5	2.4	2.9	2.5	2.0	2.3	1.8	2.4	2.5
Q7	1.5	2.9	2.3	2.9	1.3	1.0	1.3	2.8	2.0	3.3

As culturas também influenciam as diversas formas de programar pela parte dos dois géneros.

Na consolidação de conhecimentos os alunos do sexo masculino têm maior predisposição.

"females generally considered the programming concepts to be no more difficult than did the men" (Murphy et al., 2006, p. 17)

As mulheres iniciam a sua formação na universidade com menos conhecimentos base, ou seja adquirem menos conhecimentos e menos experiência de programação nos anos antes da formação universitária. Este é um fator que leva as mulheres a terem menos confiança nas suas capacidades. (Murphy et al., 2006)

"There is also considerable research suggesting that women students have less confidence in their computing abilities than their male peers." (Murphy et al., 2006, p. 17).

Na pesquisa realizada, pude encontrar os resultados do estudo realizado aos alunos dos dois géneros e os resultados apresentados foram:

- As médias das idades estavam bastante perto, o que leva a uma análise mais fiel:
 - 24.5 anos para as alunas do género feminino;
 - 25 anos para os alunos do género masculino.
- As médias de início na programação mostram disparidades:
 - 18.5 anos para as alunas do género feminino;
 - 16.5 anos para os alunos do género masculino.

O desempenho académico apresenta-se também perto, apesar das mulheres manterem os resultados atrás dos homens (Murphy et al., 2006, p. 20):

- Para as alunas do género feminino:

(Na universidade) GPA - 3.28;

(Antes da universidade) CS GPA -3.25.

- Para os alunos do género masculino:

(Na universidade) GPA -3.34;

(Antes da universidade) CS GPA - 3.32.

2.2.6 Dificuldade em dominar conceitos

A base da programação são conceitos e paradigmas. Para se dominar o código tem de se dominar primariamente os conceitos de onde eles advêm.

Para um programador aprendiz esta é a primeira barreira que terá de transpor. Terá de estudar os conceitos, os paradigmas e os primórdios da programação para depois evoluir e escrever software e aplicações mais coesas evitando erros de semântica que originam sempre muitos erros e obtendo assim aplicações fluídas e seguras. Com tantos conceitos a adquirir e com a pouca predisposição para os assimilar o aluno de programação encontra aí uma das suas dificuldades. É de todo impossível criar software código sem que o entendamos primeiro os seus princípios (Gomes et al., 2008).

2.2.7 Falta de teste – Simulação

Após desenvolver um software existe a necessidade de realizar testes unitários. É necessário saber se o software apresenta erros ou falhas que tenham de ser alteradas e corrigidas e para isso é necessário realizar testes unitário para detetar as possíveis falhas ou erros antes de o dar como terminado ao cliente. A questão essencial da necessidade de realizar testes unitários de um novo software prende-se com o fato de podermos realizar as modificações e as melhorias dentro do prazo estipulado para o desenvolvimento do software. A verdade é que para os

novos programadores este é um passo que é ignorado ou esquecido. Não dando a devida atenção e importância ao software que desenvolvem (Gomes et al., 2008).

2.3 Paradigmas de programação

As linguagens mais comuns inseridas nos planos curriculares segundo a ACM e a IEEE são o *c* e o *java*. Como linguagem de scripting a utilizada é o *JavaScript*: (ACM/IEEE, 2013).

2.3.1 PROGRAMAÇÃO APLICACIONAL

A programação aplicacional é usada para resolver problemas específicos, para produzir relatórios específicos e para atualizar ficheiros específicos. Os termos deste tipo de programação são usados distintamente da programação de sistemas. Isto quer dizer que as ferramentas utilizadas para o funcionamento desta linguagem são ferramentas que os programadores usam (Cody, 2003).

Linguagens utilizadas:

- Aplicações de programação – Fortran, Ada
- Aplicações científicas - C
- Processamento de dados aplicacionais – Cobol

2.3.2 PROGRAMAÇÃO CONCORRENTE

A programação concorrente, consiste em diversas ações de um ou mais programas que executam ao mesmo tempo uma determinada ação. Um programa concorrente é composto por múltiplos processos computacionais. Em cada processo computacional cada um dos processos é uma computação sequencial, isto é uma forma linear ordenada por uma série de passos (Hemmendinger, 2003).

2.3.3 PROGRAMAÇÃO FUNCIONAL

A programação funcional, também chamada de programação de aplicação, é um modo que usa a aplicação de função como a única estrutura de controlo. Em vez de declarações condicionais, usa expressões condicionais para produzir resultados alternativos. Em vez de uma declaração de atribuição, usa a ligação de parâmetro para criar argumentos que dão origem a um nome e a um valor. Ao invés de sequenciamento explícito ou fluxo de controlo, usa um padrão de invocações para originar um resultado. A importância imediata da resolução de problemas em geral é que, assim como a função pode demorar vários argumentos, um resultado pode ter vários componentes (Karshmer, 2003).

2.3.4 PROGRAMAÇÃO MODELAR

Um sistema modelar pode ser definido como uma parte distinta logicamente de um programa maior. Um programa completo pode assim ser considerado como um conjunto de módulos. Um módulo construído adequadamente aceita uma entrada bem definida quanto ao seu conteúdo e estrutura, realiza um conjunto bem definido de ações de transformação, e produz uma saída que está bem definida quanto ao seu conteúdo e estrutura. Neste sentido, um módulo de construção adequado tem um ponto de entrada e saída única. Se é uma sub-rotina, retorna apenas a instrução seguinte que foi executada. Os módulos, neste sentido, são uma espécie de decomposição funcional, frequentemente apelidado de abstração procedimental. Em muitas linguagens, uma sub-rotina (procedimento) pode servir como um módulo (Barker, McDowell, & Kalahar, 2009).

2.3.5 LITERATURA DE PROGRAMAÇÃO

Um programador que cria documentação para um programa depois de estar desenvolvido muitas vezes tem dificuldade em fazer a documentação corresponder ao programa. Literatura de programação é um método de combinação entre programa e documentação, de modo que pode ser realizado em conjunto. Processamento especial permite que um programa ao ser escritos em qualquer ordem possa melhorar e simplificar a exposição; macros são automaticamente numeradas de modo que a sua utilização seja facilmente cruzada. Um

sistema de programação letrada converte a documentação e código literatura sem nenhum esforço adicional por parte do programador. Todos os recursos se combinam para simplificar e incentivar o processo de documentação e para mantê-lo em estreita correspondência com o programa atual (Fuqua, Slagle, & Gini, 2003).

2.3.6 PROGRAMAÇÃO LÓGICA

Programação em Lógica foi introduzida, no início de 1970, a partir de uma convergência de trabalhos para provar o teorema da automatização, a inteligência artificial e as linguagens formais.

Em 1972, Colmerauer e Rousell agregando ideias procederam à implementação da primeira linguagem de programação lógica - Prolog. Warren, em Edimburgo, mostrou posteriormente que Prolog podia ser implementado com uma eficiência semelhante à do Lisp e fez a implementação de um compilador para Prolog, o Prolog -lo.

Kowalski & Clark, (1978), estenderam a interpretação processual para permitir condições negativas. O que permitiu criar as condições negativas foi o método teorema. A prova resultante da implementação desse método é o SLDNF, em que o SLD significa uma negação por falha. Da implementação do método teorema surgiu então o SLDNF que se tornou a base para a maioria dos trabalhos em lógica de programação até aos dias de hoje (Kowalski & Clark, 2003).

2.3.7 MICROPROGRAMAÇÃO

Microprogramação é uma técnica usada por programadores para implementar funções de controlo de um computador. Geralmente usa memória read-only (ROM) que implementa instruções de descodificação. Este tipo de programação é uma técnica específica que, *“to provide a systematic approach and an orderly approach to designing the control section of any computing system.”* (Flynn, 2003, p. 1).

2.3.8 PROGRAMAÇÃO ESTRUTURADA

Programação estruturada pode ser definida como um estilo metodológico pelo qual a programação é construída por concatenação ou subunidades lógicas coerentemente juntas. Que podem ser os próprios programas estruturados ou então são-no a partir de um ou de outro um pequeno número de estruturas de controlo bem definidas (Reilly, 2003).

2.3.9 PROGRAMAÇÃO AUTOMÁTICA

O objetivo da programação automática é permitir que os programadores possam especificar o que o programa faz e deixar que o sistema de programação gere um código que diga como o fazer. Mais tarde os programadores escreveram os seus programas utilizando código binário. A programação utilizando a linguagem máquina está longe de ser removida por um nível em que os humanos conceberam os algoritmos para resolver problemas. O compilador foi uma das primeiras ferramentas de programação automática desenvolvida permitindo o conhecimento para a codificação em linguagem como o Fortran em fórmulas algébricas e estruturas de controlo de alto nível (Lowry, 2003).

2.3.10 LISTA DE PROCESSOS

Os dois elementos de um programa de computador, consistem em cálculos, as ações a executar e dados, coisas que queremos que as ações façam. Os cálculos são definidos utilizando uma linguagem de computador, combinados para formar procedimentos, os quais por sua vez são combinados para formar compostos, procedimentos e eventualmente programas. A capacidade de combinar expressões simples em procedimentos é a chave para o uso de programas de computador para modelar processos do mundo real (Fuqua et al., 2003).

2.3.11 PROGRAMAÇÃO MÁQUINA - ASSEMBLY

A linguagem de máquina, tradicionalmente, significa que a representação particular de instruções e dados que é diretamente utilizável. A unidade de processamento central da máquina em questão. Era uma linguagem de baixo nível, de fato, o mais baixo possível, com

o corolário de que reflete a estrutura interna da máquina muito mais do que os fins a que a maioria dos utilizadores gostaria de colocar o computador. Também foi hard-wired-intencionados que era a língua nativa do hardware, e não precisou de tradutor software (Halpern, 2003).

2.3.12 PROGRAMAÇÃO ORIENTADA A OBJETOS (POO)

A programação orientada a objetos, serve para modelar sistemas de entidades e foi criada com o objetivo de separar a estrutura interna das interações externas. Facilita a ocultação e ou o encapsulamento do estado "interno" das entidades e da especificação de propriedades interativas de entidades por uma interface de operações. Faz a separação do funcionamento interno das entidades - como bancos, aviões ou pessoas - do seu funcionamento externo - interação com outras entidades. A divisão é feita tendo como base a divisão de um sistema de entidades em pedaços associados com objetos de modo que cada parte é responsável pela sua própria proteção contra o acesso de operações não autorizadas. Num ambiente em simultâneo, os objetos protegem-se automaticamente contra o acesso assíncrono, removendo a carga de sincronização, a partir de processos que acedem aos dados do objeto (Rumbaugh, 2003).

2.3.13 PROGRAMAÇÃO DE SISTEMAS

A programação de sistemas está preocupada com os sistemas operacionais, programas utilitários e software de biblioteca necessários para manter os sistemas de computadores a funcionar. Ao contrário das áreas tradicionais da ciência de computação, como compiladores, algoritmos ou estruturas de dados, os tópicos incluídos na programação de sistemas são menos focados e mais diversificados. Em suma, programação de sistemas pode ser descrita como a junção do hardware de um computador ao seu sistema operacional, utilitários, como compiladores, editores e outros aspetos do funcionamento adequados do computador no dia-a-dia (Narten, 2003).

Tabela 3: Linguagens vs Paradigmas

Paradigma/ Linguagem	Programação aplicacional	Programação funcional	Programação lógica	Programação máquina	Programação orientada a objetos - POO
Java					X
C	X				
Assembly				X	
Python	X	X			X
ASP			X		
PHP	X	X			X
C++					X
HTML/ JavaScript		X			X

```

/* Hello World program */
#include<stdio.h>

main() {
    printf("Hello World");
}

```

Figura 1: Linguagens C – Hello World

```

public class Helloworld {
    public static void main(String[ ] args) {
        System.out.println("Hello World!");
    }
}

```

Figura 2: Linguagens Java – Hello World

```
<html>
  <script>
    alert('Hello World!')
  </script>
</html>
```

Figura 3: Linguagens JavaScript – Hello World

Como se pode constatar, em apenas três linguagens um simples ‘Hello world’ apresenta-se de forma diferente em todas elas. Sendo as três tão diferentes entre si, gera a problemática de as saber gerir. Para que seja possível, o manuseamento correto e claro, destas das linguagens, o programador tem de as dominar. Esta é uma das provações para os novos alunos de programação – adquirir e consolidar conhecimentos.

2.4 Plano curricular dos cursos de Informática

2.4.1 Planos curriculares nacionais

De universidade para universidade o plano curricular é apresentado de formas diferentes. No fundo os planos curriculares têm o mesmo fundamento mas são apresentados de maneiras distintas dependendo das universidades (ACM/IEEE, 2013). Os vários cursos completam-se entre si o que gera uma grande comunidade para partilha de informação.

Ao observar o plano curricular das quatro universidades portuguesas (ISCTE-IUL – ISCTE Instituto Universitário de Lisboa (“ISCTE-IUL”); Instituto Superior Técnico (IST) (“Ensino | Técnico Lisboa”); ISEL – Instituto Superior de Engenharia de Lisboa (“ISEL - ADEETC - Candidatos”); FCUL – Faculdade de Ciências da Universidade de Lisboa (“Engenharia Informática | Faculdade de Ciências da Universidade de Lisboa”)), retirei algumas conclusões preliminares.

As quatro universidades, apresentam planos curriculares semelhantes, quanto às linguagens de programação lecionadas ao longo dos cursos. Todas elas têm como base à introdução de programação a linguagem C, variando entre outras linguagens como o Python e para orientação a objeto optam pela linguagem Java.

Além disto observei que nenhum destes cursos apresentava unidades curriculares de segurança. Ao aprofundar a minha pesquisa pelos vários documentos para a realização deste trabalho deparei-me com uma informação interessante, as linguagens de programação aconselhadas para introdução á aprendizagem de programação são também C e Java e para as linguagem de scripting aconselham as linguagens de Python ou Javascript (ACM/IEEE, 2013).

Tabela 4: Linguagens de faculdades portuguesas

Paradigmas:			Programação Maquina	Programação Estrutural		Programação Orientada a Objetos		Programação Estrutural
Universidades	Institutos / Escolas / Faculdade	Cursos	assembly	C	Python	Java	C++	HTML / JavaScript
ISCTE - IUL	ISTA	EI				X		
UL	IST	EIC		X	X	X		
	FCUL	EI	X			X		
UBI		IW		X		X		X
UM		EI				X	X	
Ualg		EI		X		X		
UE	ECT	EI	Não são especificadas as linguagens lecionadas					
UP	FE	EIC		X		X	X	X
UC	FCT	EI		X		X		
UA	FE	EI			X	X		X
UTAD	ECT	EI	Não são especificadas as linguagens lecionadas					
UM		EI	Não são especificadas as linguagens lecionadas					
UAçores		IRM	Não são especificadas as linguagens lecionadas					
UNL	FCT	MIEI				X		

2.4.2 Computer Science Curricula 2013 (CS2013)

ACM e IEEE-Computer Society têm uma longa história de juntar esforços para estabelecer diretrizes curriculares internacionais para cursos de graduação em computação.

O CS2013 é a mais recente série de diretrizes curriculares. Devido ao fato do campo da computação crescer e se diversificar existiu a necessidade de criar várias recomendações curriculares. Existem neste momento vários volumes curriculares para Engenharia da Computação, Sistemas de Informação, Tecnologia da Informação, Engenharia de Software e Ciência da Computação. Estes volumes curriculares são atualizados regularmente com o objetivo de manter os currículos da computação modernos e relevantes.

As orientações do CS2013 incluem um corpo redefinido do conhecimento, que é resultado de repensar os fundamentos necessários para um currículo de Ciência da Computação. Ele também procura identificar exemplos de cursos e programas reais para fornecer orientações concretas sobre a estrutura e desenvolvimento curricular numa variedade de contextos institucionais. O desenvolvimento de diretrizes curriculares para a Ciência da Computação tem sido sempre um desafio, dada a rápida evolução e expansão do campo. A crescente diversidade de tópicos potencialmente relevantes para uma educação na Ciência da Computação e a crescente integração da programação com outras disciplinas cria desafios específicos para este esforço.

Como resultado, o Comitê de Direção do CS2013 tem feito um esforço considerável para envolver mais amplamente a comunidade de educação ciência da computação numa caixa de diálogo para melhor entender as novas oportunidades e necessidades locais e para identificar modelos bem-sucedidos de computação curriculares – quer estabelecidos ou novos (ACM/IEEE, 2013).

2.5 Fatores de Continuidade e confirmação na aprendizagem

Para analisar com maior profundidade a continuidade dos estudos nos alunos do ensino superior, analisei as teorias como a difusão da inovação (Davis, 1993), a aceitação do modelo de tecnologia (Fishbein & Ajzen, 1991) e o comportamento planeado (Rogers, 1983) (Ajzen, 1991), são estudos sobre as variantes individuais para a aceitação dos sistemas de informação e quais as atitudes.

O artigo tem como objetivo a análise da atividade cognitiva e a influência da intenção, na decisão de continuar ou não continuar a utilizar os sistemas de informação. Através da teoria da confirmação da expectativa, que foi adaptada a partir da literatura do comportamento do consumidor e integrada com resultados teóricos e empíricos da pesquisa anterior, é usada para teorizar um modelo de intenção de continuação (IS continuance intention). Como público-alvo, a pesquisa utilizou os serviços bancários on-line. O estudo revelou que a vontade de continuar (IS continuance intention) é determinada pelo grau de satisfação do utilizador. A satisfação do utilizador, por sua vez, é influenciada pela sua confirmação da expectativa de antes conjugado com a utilidade que retira do produto. O estudo direcciona a sua atenção para as diferenças substanciais entre comportamentos de aceitação e continuidade. (Bhattacharjee, 2001)

2.5.1 Aplicação do modelo de estudo

Vai-se estudar se as teoria de difusão da inovação (Rogers, 1995), de aceitação do modelo de tecnologia (Fishbein & Ajzen, 1991) e o comportamento planeado (Davis, 1993) (Ajzen, 1991), e verifica-se no âmbito da aprendizagem da programação. O aluno inicialmente toma conhecimento de todas as ofertas do ensino superior. Após tomar consciência de toda a oferta o aluno opta por aquela que acha adequar-se mais à sua necessidade. Posteriormente, já com o decorrer do ano letivo o aluno vai validando se o curso está a enquadrar-se nas suas expectativas e com a qualidade do produto esperada (curso, no caso em questão), é então aqui que se dá a confirmação do curso e a intenção de continuar, está comprado pois já o frequenta e pretende terminar. Lógico que o final só poderá ser designado ao final de cada semestre, aí sim já terá experiência suficiente para saber se recompra o produto, que é o mesmo que

decidir se continua o curso - se efetua mais uma matrícula - ou se abandona e se decide por outro produto/curso (Bhattacharjee, 2001).

A aplicação do modelo foi estudada no comportamento dos compradores. No estágio do conhecimento e persuasão os compradores avaliam a expectativa do produto ou serviço. No estágio da decisão os compradores aceitam o uso do produto ou serviço. No estágio da implementação os compradores avaliam a percepção e desempenho versus a sua expectativa inicial e se a expectativa foi confirmada. No último estágio, o da aceitação o comprador decide se compra ou não o produto chegando então à confirmação (Bhattacharjee, 2001).

2.6 Síntese da revisão da literatura

Há necessidade de solucionar falhas que existem no ensino da programação como os problemas prático que são essencialmente vocacionados para a vertente industrial, a falta de ginástica mental para a resolução de problemas, o tempo vs programação, a idade vs. género vs programação (Murphy et al., 2006) que pode limitar em diversos aspetos como a dificuldade em dominar (Gomes et al., 2008) os conceitos base e ainda a motivação social que em certos casos delimita o sucesso dos alunos.

Há necessidade de combater as problemáticas na programação já existentes por isso existem paradigmas de programação (Gomes et al., 2008). Os paradigmas de programação, são ferramentas que afetam positivamente o sucesso dos programadores (Barker et al., 2009).

O CS2013 é a mais recente série de diretrizes curriculares. Devido ao fato do campo da computação crescer e se diversificar existiu a necessidade de criar várias recomendações curriculares. Existem neste momento vários volumes curriculares para Engenharia da Computação, Sistemas de Informação, Tecnologia da Informação, Engenharia de Software e Ciência da Computação. Estes volumes curriculares são atualizados regularmente com o objetivo de manter os currículos da computação modernos e relevantes (ACM/IEEE, 2013).

3 Proposta do modelo de investigação

3.1 Definição dos constructos adaptados à aprendizagem da programação.

Com base na revisão da literatura realizada sobre os diversos estudos de modelos de intenção e continuidade, propôs-se estudar a realidade da continuidade de aprendizagem de programação recorrendo a diversos constructos teóricos. Assim tentando colmatar a falta de estudos na área de aprendizagem de programação, nomeadamente para entender as intenções dos alunos em continuar a estudar programação depois de ingressarem em licenciaturas de informática, propôs-se o estudo das relações entre os seguintes constructos: desempenho individual, reputação da universidade percebida pelo aluno, suporte académico, fatores familiares, utilidade percebida pelo curso, expectativa do aluno, desempenho individual, intenção de continuar a estudar no curso. A Tabela 5 apresenta as definições de cada constructo, que vão servir de base ao modelo proposto.

Tabela 5: Definição dos constructos

Variável latente	Códigos	Definição	Autor
Desempenho Individual	SE	O sucesso que o aluno teve no curso até ao momento.	(Bandura, A., 1981)
Reputação Percebida	RP	A importância da reputação da universidade e das disciplinas de programação para os alunos.	(Veloutsou et al. 2004)
Suporte Académico	AC	Condições universitárias.	
Fatores ponderadores da entrada	E	A importância da escolha por parte dos novos alunos, de um curso numa universidade ou politécnico.	

Variável latente	Códigos	Definição	Autor
Família	F	A pressão exercida pelo seio familiar na escolha do curso mediante o nível de empregabilidade futura.	
Intenção de continuação	Int	Demonstra a intenção de dar continuidade ao curso e à aprendizagem da programação.	(Mathieson, 1991)
Satisfação	Sat	A satisfação dos alunos com o curso e com a aprendizagem da programação.	(Spreng et al., 1996)
Utilidade	PU	Mostra a utilidade que os alunos dão ao curso e à aprendizagem da programação.	(Bhattacharjee, 2001)
Confirmação	Conf	A confirmação positiva relativa ao curso e à aprendizagem de programação, por parte dos novos alunos do ensino superior.	(Bhattacharjee, 2001)
Expetativa	Ex	A expetativa é positiva nos alunos dos cursos de ensino superior e na aprendizagem da programação.	(Bhattacharjee, 2001)
Desempenho Individual	DI	A percepção de que estão a desenvolver atividades no curso e a desenvolver a aprendizagem da programação cria um desempenho individual positivo nos alunos.	(Bhattacharjee, 2001)

3.1.1 Proposta das hipóteses

Bhattacharjee (2001) afirma que existem 5 estágios para a decidir algo - Conhecimento, persuasão, decisão, implementação e a fase de aceitação. No final da sua tese, reforçou a necessidade de se criar um momento de reflexão para que se reavalie a decisão tomada.

A importância dada à reputação da universidade e às disciplinas de programação lecionadas para os novos alunos torna-se uma forte premissa na escolha de um curso, para a escolha do curso têm em conta as condições universitárias oferecidas para obtenção de bons resultados académicos (Bhattacharjee, 2001; Kopanidis, 2008). Daqui resultará a seguinte hipótese de investigação:

Hipótese 1 (H1) - Há uma relação positiva entre a reputação percebida e o suporte acadêmico.

A importância dada às condições universitárias oferecidas para obtenção de bons resultados acadêmicos aliado à expectativa positiva por parte dos novos alunos dos cursos de ensino superior e na aprendizagem da programação (Gomes et al., 2008), Bhattacharjee (2001) afirma que a expectativa é importante pois sofre mutações ao longo do tempo. Daqui resultará a seguinte hipótese de investigação:

Hipótese 2 (H2) - Há uma relação positiva entre o suporte acadêmico e expectativa.

A importância que os alunos dão às condições universitárias afetam positivamente a obtenção de bons resultados acadêmicos facilitando assim um bom desempenho no curso e na aprendizagem de programação (Gomes et al., 2008), por parte dos novos alunos do ensino superior. Daqui resultará a seguinte hipótese de investigação:

Hipótese 3 (H3) - Há uma relação positiva entre suporte acadêmico e a confirmação.

A percepção de que os novos alunos estão a desenvolver atividades no curso e a desenvolver a aprendizagem da programação cria um desempenho individual positivo nos alunos aliado à confirmação positiva relativa ao curso e à aprendizagem de programação pode ser um bom incentivo, Bhattacharjee (2001) defende que a confirmação avalia os utilizadores para determinar a sua resposta avaliativa ou de satisfação. Daqui resultará a seguinte hipótese de investigação:

Hipótese 4 (H4) - Há uma relação positiva entre o desempenho individual e a confirmação.

A confirmação positiva relativa ao curso e à aprendizagem de programação, por parte dos novos alunos do ensino superior em que Bhattacharjee (2001) defende que a confirmação avalia os utilizadores para determinar a sua resposta avaliativa ou de satisfação. Enquanto Davis et al. (1989) define que a utilidade influencia a intenção do utilizador de continuar a utilizar diversas fases temporais. Daqui resultará a seguinte hipótese de investigação:

Hipótese 5 (H5) - Há uma relação positiva entre a confirmação e a utilidade.

A confirmação positiva relativa ao curso e à aprendizagem de programação, por parte dos novos alunos do ensino superior. Bhattacharjee (2001) defende que a confirmação está positivamente relacionada à satisfação com a utilização. Pois a confirmação avalia os utilizadores para determinar a sua resposta avaliativa ou satisfação. Em paralelo satisfação dos alunos com o curso e com a aprendizagem da programação Spreng (1996) define que a satisfação é a chave para construir e manter uma base lealdade ao longo prazo. Daqui resultará a seguinte hipótese de investigação:

Hipótese 6 (H6) - Há uma relação positiva entre a confirmação e a satisfação.

Os alunos quando percebem que o curso tem utilidade podem aumentar o seu grau de satisfação. Davis et al.'s (1989) define que a utilidade influencia a intenção do utilizador de continuar a utilizar diversas fases temporais. Enquanto, que Spreng (1996) define que a satisfação é a chave para construir e manter uma base leal de consumidores a longo prazo. Daqui resultará a seguinte hipótese de investigação:

Hipótese 7 (H7) - Há uma relação positiva entre a utilidade e a satisfação.

Os alunos quando percebem que o curso tem utilidade podem aumentar o seu grau de satisfação. Davis et al. (1989) define que a utilidade influencia a intenção do utilizador de continuar a utilizar diversas fases temporais. A intenção de continuação demonstra a intenção de dar continuidade ao curso e à aprendizagem da programação, Mathieson (1991) leva em conta as distinções entre o comportamento de aceitação e de continuidade. Daqui resultará a seguinte hipótese de investigação:

Hipótese 8 (H8) - Há uma relação positiva entre a utilidade e a intenção de continuação.

A expectativa é a confirmação positiva relativa ao curso e à aprendizagem de programação, por parte dos novos alunos do ensino superior, para Bhattacharjee (2001) a expectativa sofre mutações ao longo do tempo. A intenção de continuação demonstra a intenção de dar

continuidade ao curso e à aprendizagem da programação, Mathieson (1991) leva em conta as distinções entre o comportamento de aceitação e de continuidade. Daqui resultará a seguinte hipótese de investigação:

Hipótese 9 (H9) - Há uma relação positiva entre a expectativa e a intenção de continuação.

A satisfação dos alunos com o curso e com a aprendizagem da programação, Davis et al. (1989) define que a utilidade influencia a intenção do utilizador de continuar a utilizar diversas fases temporais. Enquanto, que Spreng (1996) define que a satisfação é a chave para construir e manter uma base leal de consumidores a longo prazo. A intenção de continuação demonstra a intenção de dar continuidade ao curso e à aprendizagem da programação, Mathieson (1991) leva em conta as distinções entre o comportamento de aceitação e de continuidade. Daqui resultará a seguinte hipótese de investigação:

Hipótese 10 (H10) - Há uma relação positiva entre a satisfação e a intenção de continuação.

3.2 Modelo de investigação

Após a revisão da literatura e definidas as hipóteses de investigação, com base na teoria do comportamento planeado de Fishbein & Ajzen (1991) propõe-se para este estudo o modelo de investigação ilustrado na Figura 4: Modelo de Investigação.

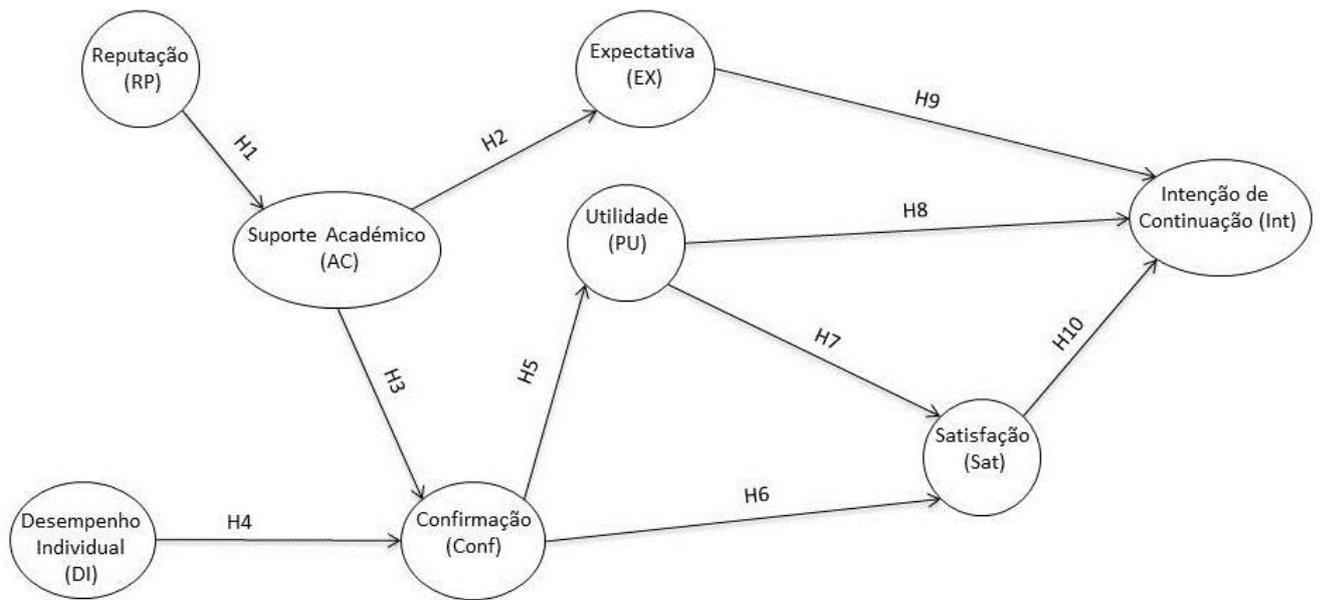


Figura 4: Modelo de Investigação

O modelo conceptual possui como objetivo explicitar os fatores dos conceitos estudados em detalhe na revisão da literatura. A variável dependente do modelo é a intenção de continuar a estudar. Como variáveis independentes apresenta uma dimensão de Veloutsou et al. (2004) e Soutar and Turner (2002): reputação (RP). E uma dimensão de Bhattacharjee (2001) – desempenho individual (DI). Como variáveis dependentes apresenta Veloutsou et al. (2004) e Soutar and Turner (2002): universidade (AC), apresenta Mathieson's (1991): intenção de continuação (Cont. Int.), apresenta Spreng et al.'s (1996): satisfação (Sat.), apresenta Bhattacharjee (2001): confirmação (Cont) e expetativa (EX). O modelo proposto utiliza as dimensões de desempenho individual do aluno, de expetativas do aluno quanto ao curso, nível de utilidade e nível de satisfação para explicar a intenção de continuar a estudar no cursos de programação.

Tabela 6: Perguntas realizadas aos alunos

Variável latente	Código	Tradução	Autor
Desempenho Individual	SE1	Tenho a certeza que vou entender o conteúdo da programação seja ela fácil ou difícil.	(Bandura, 1981)
	SE2	Eu não me sinto confiante para entender todos os conceitos de programação.	
	SE3	Eu tenho a certeza que vou ter sucesso na programação.	
	SE4	Por mais que me esforce, não sou capaz de aprender programação.	
	SE5	Quando as atividades de programação são muito difíceis, eu desisto ou só faço as partes fáceis.	
	SE6	Durante os exercícios de programação, prefiro pedir as respostas a outras pessoas em vez de pensar por mim mesmo.	
	SE7	Quando eu encontro conteúdo de programação difícil, não me esforço por aprendê-lo.	
Reputação Percebida	RP1	Reputação da universidade	(Veloutsou et al., 2004)
	RP2	Ajustamento ao curso	
	RP3	Reputação do curso	
	RP4	Prestígio e status da universidade	
Suporte Académico	AC1	Custo das propinas	
	AC2	Recursos da universidade (biblioteca, laboratórios de informática, e / ou salas de aula, Wi-Fi).	
	AC3	Atividades extra curricular e outras infraestruturas disponíveis.	
	AC4	Experiência e a qualificação dos professores.	
Fatores ponderadores da entrada	E1	Os requisitos de entrada são importantes na escolha do curso.	
	E2	Considera importante a gama de cursos disponíveis.	
	E3	Considera importante o tipo de instituição (ex. Universidade, Politécnico).	
	E4	Considera importante localização da universidade/instituto.	

Variável latente	Código	Tradução	Autor
Família	F1	A opinião da família foi importante na escolha do curso.	(Veloutsou et al., 2004)
	F2	Eu escolhi o curso e sua percentagem de empregabilidade	
Intenção de continuação	Int1	É minha intenção continuar com o curso.	(Mathieson, 1991)
	Int2	Se eu pudesse, eu interrompia o curso.	
	Int3	Tenho intenção de continuar a aprofundar os conhecimentos de programação	
	Int4	Se eu pudesse, deixava de estudar programação.	
	Int5	Eu tenho intenção de continuar a estudar nesta universidade/instituto	
	Int6	Eu não pretendo continuar a estudar nesta universidade/instituto	
Satisfação		Como me sinto em relação à minha experiência no curso	(Spreng et al., 1996)
	Sat1	Estou satisfeito em relação à minha experiência no curso.	
	Sat2	Estou agradado com a minha experiência no curso.	
	Sat3	É gratificante a minha experiência no curso.	
	Sat4	Estou muito entusiasmado com a minha experiência no curso.	
	Sat5	Estou satisfeito em relação à minha experiência na programação.	
	Sat6	Estou agradado com a minha experiência na programação.	
	Sat7	É gratificante a minha experiência na programação.	
	Sat8	Estou muito entusiasmado com a minha experiência na programação	
Utilidade	PU1	Este curso melhorou o meu desempenho na programação	(Davis et al., 1989)
	PU2	Este curso aumentou a minha produtividade na programação.	
	PU3	Este curso aprimorou a minha eficácia na programação.	
	PU4	No geral, o curso é útil na programação	
Confirmação	Conf1	A minha experiência com o curso superou as minhas expectativas.	(Bhattacharjee, 2001)
	Conf2	O nível de suporte prestado no curso foi melhor do que eu esperava.	

Variável latente	Código	Tradução	Autor
Confirmação	Conf3	No geral, as minhas expetativas foram confirmadas na realização do curso.	(Bhattacharjee, 2001)
	Conf4	A minha experiência nas disciplinas de programação superou as minhas expetativas.	
	Conf5	O nível de suporte prestado nas disciplinas de programação foi melhor do que eu esperava.	
	Conf6	No geral, as minhas expetativas foram confirmadas na realização das disciplinas de programação.	
Expetativa		Tinha no início expetativas elevadas com:	(Bhattacharjee, 2001)
	EX1	A experiência no curso	
	EX2	O suporte prestado no curso	
	EX3	O geral do curso	
	EX4	A minha experiência das disciplinas do curso	
	EX5	O suporte prestado na disciplina de programação	
Desempenho Individual		Foi atingido um bom desempenho:	(Bhattacharjee, 2001)
	DI1	A experiência no curso	
	DI2	O suporte prestado no curso	
	DI3	O geral do curso	
	DI4	A minha experiência das disciplinas do curso	
	DI5	O suporte prestado na disciplina de programação	
	DI6	O geral da disciplina de programação	

Da Tabela 5 resultou um questionário que foi realizado através de um inquérito on-line a alunos do ensino superior português e que frequentam pelo menos o segundo ano de licenciatura de um curso de informática. O inquérito foi realizado através de uma plataforma para inquéritos GoogleDrive Forms, o referido questionário encontra-se no Anexo K.

4 Análise de dados e resultados

4.1 Análise da amostra

A amostra do presente estudo incide numa população adulta em frequência no ensino superior tanto em licenciaturas, mestrados e doutoramentos. A análise do comportamento tomado pelos novos alunos do ensino superior na escolha do curso e na aprendizagem da programação nesta investigação, assenta no comportamento individual numa organização tanto pública como privada. Assim, os indivíduos que frequentam o ensino superior foram considerados na amostra, uma vez que as instituições de ensino são também elas organizações.

O inquérito foi realizado a 170 indivíduos, todos foram considerados válidos. Cohen (1992) defende que a dimensão da amostra recomendada, com a utilização do software SMART-PLS V2, para um poder estatístico de 80% é de 170 amostras para um número máximo de três setas apontadas a um constructo. Assim, com a amostra de 170 indivíduos que se possui, o nível de significância esperada é de 1% e um R² mínimo de 0,10, como se pode constatar na Tabela 7.

Tabela 7: Definição da dimensão mínima da amostra Cohen (1992).

Nível de Significância												
Número Máximo de Setas Apontadas a um Constructo	1%				2%				3%			
	R ² Mínimo				R ² Mínimo				R ² Mínimo			
	0.10	0.25	0.50	0.75	0.10	0.25	0.50	0.75	0.10	0.25	0.50	0.75
2	158	75	47	38	110	52	33	26	88	41	26	21
3	176	84	53	42	124	59	38	30	100	48	30	25
4	191	91	58	46	137	65	42	33	111	53	34	27
5	205	98	62	50	147	70	45	36	120	58	37	30
6	217	103	66	53	157	75	48	39	128	62	40	32
7	228	109	69	56	166	80	51	41	136	66	42	35
8	238	114	73	59	174	84	54	44	143	69	45	37
9	247	119	76	62	181	88	57	46	150	73	47	39
10	256	123	79	64	189	91	59	48	156	76	49	41

4.1.1 Caracterização da amostra

A amostra conta um maior número de elementos do género masculino (81.8%) e a sua grande maioria está entre a idade dos 17 aos 22 anos.

Tabela 8: Amostra - Género e idades

Descrição da Variável	N	Percentagem
Masculino	139	81.8%
Feminino	31	18.2%
Idade	170	100%
17 - 22	108	63.53%
23 - 26	30	17.65%
27 - 33	9	5.29%
> 31	23	13.53%

A grande maioria da amostra é proveniente de Lisboa (40%), seguido Setúbal (24.71%), Leiria (8.82%), Santarém (7.65%). Equiparados estão os alunos provenientes de Beja (2.94%) e da Madeira (2.94%), seguidos dos Açores (2.35%) e do Porto (2.35%), seguidos de Évora (1.76%) e Coimbra (1.76%), seguidos de Aveiro (1.18%) e Faro (1.18%) e com percentagens mais baixas alunos provenientes de Bragança (0.59%), Castelo Branco (0.59%), Portalegre (0.59%) e Guarda (0.59%).

Tabela 9: Amostra - Região de proveniência

Descrição da Variável	N	Percentagem
Região / Distrito de proveniência	170	100%
Açores	4	2.35%
Aveiro	2	1.18%
Beja	5	2.94%
Bragança	1	0.59%
Castelo Branco	1	0.59%
Coimbra	3	1.76%
Évora	3	1.76%
Faro	2	1.18%
Guarda	1	0.59%
Leiria	15	8.82%
Lisboa	68	40%
Madeira	5	2.94%

Portalegre	1	0.59%
Porto	4	2.35%
Santarém	13	7.65%
Setúbal	42	24.71%

A maior percentagem frequenta o curso em Lisboa (50%), seguido de Setúbal (33.53%), Leiria (8.24%), Évora (2.35%), equiparados estão o Porto (1.18%), Coimbra (1.18%), Braga (1.18%) e Beja (1.18%). Também equiparados estão Açores (0.59%) e Aveiro (0.59%).

Tabela 10: Amostra - Região do curso

Descrição da Variável	N	Percentagem
Região / Distrito do curso	170	100%
Açores	1	0.59%
Aveiro	1	0.59%
Beja	2	1.18%
Braga	2	1.18%
Coimbra	2	1.18%
Évora	4	2.35%
Leiria	14	8.24%
Lisboa	85	50%
Porto	2	1.18%
Setúbal	57	33.53%

A maior parte dos alunos frequenta o 2º Ano de Licenciatura (33.53%) seguido de 3º Ano de Licenciatura (32.35%), 2º Ano Mestrado (12.94%), 1º Ano Mestrado (8.24%), 1º Ano Licenciatura (7.65%), 4º Ano Licenciatura (4.12%), 3º Ano Doutoramento (1.18%)

Tabela 11: Amostra - Ano que frequenta

Descrição da Variável	N	Percentagem
Ano que frequenta	170	100%
1º Ano - Licenciatura	13	7.65%
2º Ano - Licenciatura	57	33.53%
3º Ano - Licenciatura	55	32.35%
4º Ano - Licenciatura	7	4.12%
1º Ano - Mestrado	14	8.24%
2º Ano - Mestrado	22	12.94%
3º Ano - Doutoramento	2	1.18%

A maior percentagem de escolaridade da Mãe é do Ensino básico (30.59%) seguido do ensino secundário (28.24%), licenciatura (27.06%), pós-graduação (6.47%), doutoramento (4.71%) e outros (2.94%).

Tabela 12: Amostra - Escolaridade da Mãe

Descrição da Variável	N	Percentagem
Escolaridade da Mãe	170	100%
Ensino básico	52	30.59%
Ensino secundário	48	28.24%
Licenciatura	46	27.06%
Pós-graduação	11	6.47%
Doutoramento	8	4.71%
Outros	5	2.94%

A maior percentagem de escolaridade do Pai é o Ensino Secundário (37.06%) seguido do ensino básico (28.82%), licenciatura (21.18%), pós-graduação (6.47%), doutoramento (5.29%) e outros (1.18%).

Tabela 13: Amostra - Escolaridade do Pai

Descrição da Variável	N	Percentagem
Escolaridade do Pai	170	100%
Ensino básico	49	28.82%
Ensino secundário	63	37.06%
Licenciatura	36	21.18%
Pós-graduação	11	6.47%
Doutoramento	9	5.29%
Outros	2	1.18%

A maior percentagem de área da escolaridade dos alunos é de Ciências e tecnologia (83.53%) seguido de Formação profissional (11.18%), economia (2.35%), com a mesma percentagem Humanidades (1.18%) e Outros (1.18%). Artes (0.59%) e Desporto (0.59%) apresentam também a mesma percentagem.

Tabela 14: Amostra - Escolaridade alunos

Descrição da Variável	N	Percentagem
Área de escolaridade 12º	170	100%
Ciências e tecnologias	142	83.53%

Humanidades	2	1.18%
Artes	1	0.59%
Desporto	1	0.59%
Economia	4	2.35%
Formação profissional	19	11.18%
Outros	2	1.18%

A maioria dos alunos tem mais conhecimento sobre a linguagem de programação Java (88.82%) seguido de C (87.65%), assembly (65,29%), HTML/JavaScript (62.94%), C++ (37.06%), PHP (32.35%), Python (31.76%), outros (23.53%), ASP (12.35%) e nenhuma (2.35%).

Tabela 15: Amostra - Linguagens de programação

Descrição da Variável	N	Percentagem
Linguagens de programação que tem conhecimento	170	100%
Java	151	88.82%
C	149	87.65%
assembly	111	65.29%
Python	54	31.76%
ASP	21	12.35%
PHP	55	32.35%
C++	63	37.06%
HTML / JavaScript	107	62.94%
Nenhuma	4	2.35%
Outros	40	23.53%

4.2 Resultados da Avaliação do modelo de medida

Na Avaliação do Modelo de Medida (Tabela 16) são analisadas a Validade Convergente, a Fiabilidade e a Validade Discriminante do modelo de investigação.

Tabela 16: Caracterização da amostra

Variáveis latentes	Itens	Loadings	Indicador de Fiabilidade	AVE	Fiabilidade Composta (CR)	Alpha de Cronbach	Validade Discriminante
Suporte	AC1	0,780	0,609	0,615	0,827	0,709	0,615

Variáveis latentes	Itens	Loadings	Indicador de Fiabilidade	AVE	Fiabilidade Composta (CR)	Alpha de Cronbach	Validade Discriminante
Académico	AC2	0,717	0,514				
	AC3	0,850	0,722				
Confirmação	Conf1	0,891	0,794	0,776	0,954	0,942	0,776
	Conf2	0,850	0,722				
	Conf3	0,902	0,813				
	Conf4	0,863	0,745				
	Conf5	0,873	0,761				
	Conf6	0,905	0,818				
Intenção de Continuação	Int1	0,891	0,794	0,708	0,878	0,792	0,708
	Int3	0,872	0,761				
	Int5	0,753	0,568				
Expetativa	EX1	0,832	0,691	0,725	0,940	0,924	0,725
	EX2	0,866	0,750				
	EX3	0,881	0,776				
	EX4	0,801	0,641				
	EX5	0,887	0,786				
	EX6	0,839	0,704				
Desempenho Individual	DI1	0,918	0,843	0,848	0,971	0,964	0,848
	DI2	0,909	0,826				
	DI3	0,944	0,891				
	DI4	0,890	0,792				
	DI5	0,924	0,854				
	DI6	0,941	0,885				
Utilidade	PU1	0,966	0,934	0,907	0,975	0,966	0,907
	PU2	0,956	0,913				
	PU3	0,970	0,940				
	PU4	0,918	0,842				
Reputação	R1	0,893	0,797	0,703	0,904	0,860	0,703
	R2	0,780	0,609				
	R3	0,804	0,646				
	R4	0,873	0,761				
Satisfação	Sat1	0,912	0,832	0,827	0,975	0,970	0,827
	Sat2	0,923	0,852				
	Sat3	0,903	0,815				
	Sat4	0,900	0,810				
	Sat5	0,916	0,839				
	Sat6	0,912	0,831				
	Sat7	0,912	0,832				
	Sat8	0,897	0,804				

A consistência interna do modelo de medida esta garantida uma vez que o $CR > 0,7$ para todas as variáveis latentes. Os loadings dos itens apresentam valores altos, acima do limiar de 0.70, isto significa que os constructos teóricos têm fiabilidade estatística, ou seja medem de um modo correto a realidade de todas as variáveis latentes, segundo Fornell e Larcker (1981). Os AVE também estão acima de 0.50 assegurando que existe validade convergente nos constructos teóricos, ou seja, a garante a variância dos itens. A Tabela 16 mostra os resultados do teste da validade discriminante, verificando-se que os loadings são maiores que os cross loadings e que a diagonal (representa a raiz quadrada do AVE) também passou o teste de Fornel e Larker (1981).

Tabela 17: Análise de Cross-Loadings

	AC	Conf	Int	EX	DI	PU	RP	Sat
AC1	0,780	0,195	0,123	0,370	0,125	0,028	0,204	0,059
AC2	0,717	0,182	0,100	0,216	0,119	0,071	0,176	0,110
AC3	0,850	0,345	0,181	0,324	0,369	0,161	0,509	0,260
Conf1	0,336	0,891	0,548	0,354	0,781	0,692	0,469	0,731
Conf2	0,181	0,850	0,491	0,197	0,752	0,586	0,384	0,626
Conf3	0,345	0,902	0,538	0,317	0,814	0,668	0,445	0,718
Conf4	0,302	0,863	0,580	0,349	0,796	0,666	0,430	0,705
Conf5	0,281	0,873	0,502	0,259	0,735	0,543	0,388	0,640
Conf6	0,279	0,905	0,527	0,277	0,762	0,711	0,405	0,701
EX1	0,327	0,262	0,313	0,832	0,294	0,290	0,257	0,267
EX2	0,373	0,232	0,281	0,866	0,234	0,180	0,203	0,195
EX3	0,373	0,316	0,369	0,881	0,373	0,343	0,252	0,321
EX4	0,339	0,181	0,252	0,801	0,245	0,236	0,164	0,230
EX5	0,345	0,305	0,302	0,887	0,363	0,285	0,158	0,274
EX6	0,245	0,406	0,388	0,839	0,450	0,403	0,191	0,429
Int1	0,132	0,499	0,891	0,317	0,548	0,438	0,368	0,543
Int3	0,140	0,509	0,872	0,307	0,575	0,462	0,303	0,611
Int5	0,199	0,527	0,753	0,329	0,528	0,427	0,378	0,396
DI1	0,304	0,788	0,619	0,389	0,918	0,691	0,541	0,759
DI2	0,248	0,814	0,564	0,330	0,909	0,635	0,450	0,670
DI3	0,310	0,848	0,659	0,410	0,944	0,685	0,517	0,768
DI4	0,320	0,781	0,587	0,343	0,890	0,613	0,455	0,723
DI5	0,207	0,805	0,570	0,323	0,924	0,645	0,423	0,710
DI6	0,264	0,818	0,613	0,330	0,941	0,690	0,478	0,775
PU1	0,127	0,694	0,511	0,332	0,689	0,966	0,334	0,656

	AC	Conf	Int	EX	DI	PU	RP	Sat
PU2	0,148	0,716	0,509	0,332	0,674	0,956	0,383	0,621
PU3	0,152	0,709	0,518	0,310	0,695	0,970	0,382	0,645
PU4	0,048	0,680	0,460	0,327	0,673	0,918	0,317	0,613
RP1	0,340	0,351	0,336	0,188	0,430	0,309	0,893	0,364
Rp2	0,430	0,489	0,407	0,309	0,500	0,351	0,780	0,441
Rp3	0,293	0,377	0,281	0,061	0,394	0,308	0,804	0,318
Rp4	0,335	0,352	0,317	0,200	0,384	0,262	0,873	0,342
Sat1	0,203	0,791	0,587	0,313	0,809	0,689	0,436	0,912
Sat2	0,184	0,755	0,597	0,353	0,789	0,641	0,450	0,923
Sat3	0,244	0,734	0,581	0,342	0,765	0,600	0,473	0,903
Sat4	0,226	0,734	0,552	0,317	0,745	0,574	0,424	0,900
Sat5	0,127	0,667	0,577	0,263	0,685	0,598	0,355	0,916
Sat6	0,150	0,656	0,541	0,283	0,673	0,567	0,362	0,912
Sat7	0,170	0,669	0,568	0,290	0,660	0,602	0,375	0,912
Sat8	0,191	0,664	0,524	0,276	0,651	0,557	0,360	0,897

Tabela 18: Análise do Critério de Fornell e Larcker

	AC	Conf	Int	EX	DI	PU	RP	Sat
AC	0,784							
Conf	0,328	0,881						
Int	0,182	0,604	0,841					
EX	0,393	0,334	0,374	0,851				
DI	0,299	0,879	0,654	0,385	0,921			
PU	0,126	0,735	0,525	0,341	0,717	0,953		
RP	0,427	0,478	0,410	0,241	0,519	0,372	0,839	
Sat	0,207	0,782	0,623	0,336	0,797	0,665	0,447	0,909

Após a realização dos testes ao modelo de medida conclui-se assim, que as variáveis latentes (universidade, confirmação, intenção de continuação, expectativa, desempenho individual, utilidade, reputação e satisfação) são válidas e consistentes.

4.3 Avaliação do Modelo Estrutural

Os resultados do modelo estrutural estão reportados na Tabela 16 e nos Anexos G e H. Todas as hipóteses foram validadas. O modelo estrutural explica 43% da variação da expectativa, da

utilidade e da satisfação. A satisfação é explicada pelo modelo estrutural em 63% pela variação da utilidade percebida do curso e pela confirmação. A utilidade percebida do curso é explicada em 53% pela confirmação. A expectativa é explicada em 15% diretamente pelo suporte acadêmico e indiretamente pela reputação da universidade.

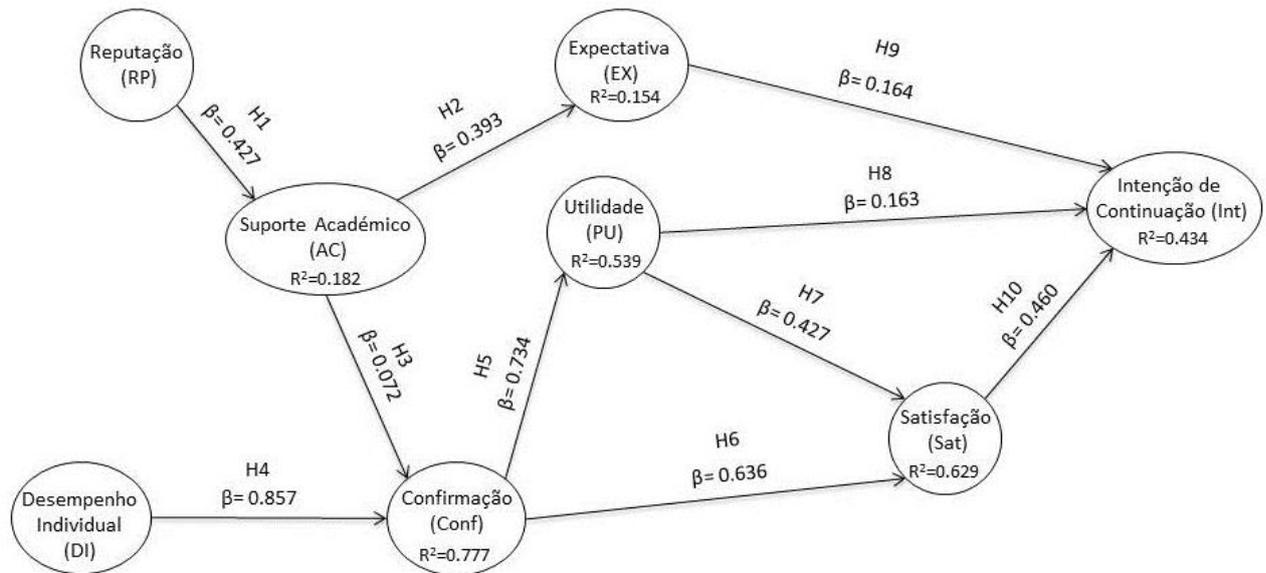


Figura 5: Resultados do Modelo de Investigação

A reputação percebida da universidade explica positivamente o suporte acadêmico ($\beta = 0,43$; $p < 0,01$), isto confirma a hipótese 1. O suporte acadêmico explica positivamente a expectativa ($\beta = 0,39$; $p < 0,01$), isto confirma a hipótese 2. O suporte acadêmico explica positivamente a confirmação ($\beta = 0,07$; $p > 0,01$), isto confirma a hipótese 3. O desempenho explica positivamente a confirmação ($\beta = 0,86$; $p < 0,01$), isto confirma a hipótese 4. A confirmação explica positivamente a utilidade ($\beta = 0,73$; $p < 0,01$), isto confirma a hipótese 5. A confirmação explica positivamente a satisfação ($\beta = 0,64$; $p < 0,01$), isto confirma a hipótese 6. A utilidade explica positivamente a satisfação ($\beta = 0,2$; $p > 0,01$), isto confirma a hipótese 7. A utilidade explica positivamente a intenção de continuação ($\beta = 0,16$; $p > 0,01$), isto confirma a hipótese 8. A expectativa explica positivamente a intenção de continuação ($\beta = 0,16$; $p > 0,01$), isto confirma a hipótese 9. A satisfação explica positivamente a intenção de continuação ($\beta = 0,46$; $p < 0,01$), isto confirma a hipótese 10.

Tabela 19: Avaliação do modelo estrutural

	Hipóteses	Efeito	β	T Test	p-value	Significância
H1	RP -> Acad	+	0,43	7,2082	0,000	***
H2	Acad -> EX	+	0,39	5,8049	0,000	***
H3	Acad -> Conf	+	0,07	1,991	0,048	*
H4	DI -> Conf	+	0,86	29,4913	0,000	***
H5	Conf -> PU	+	0,73	15,5191	0,000	***
H6	Conf -> Sat	+	0,64	8,1246	0,000	***
H7	PU -> Sat	+	0,2	2,356	0,020	**
H8	PU -> Cont_Int	+	0,16	1,9578	0,052	*
H9	EX -> Cont_Int	+	0,16	1,9139	0,057	*
H10	Sat -> Cont_Int	+	0,46	4,4664	0,000	***

5 Discussão de Resultados

A amostra do presente estudo tem em conta uma população adulta em frequência no ensino superior tanto em licenciaturas, mestrados e doutoramentos.

A análise do comportamento tomado pelos novos alunos do ensino superior na escolha do curso e na aprendizagem da programação nesta investigação, assenta no comportamento individual numa organização tanto pública como privada. Assim, os indivíduos que frequentam o ensino superior foram considerados na amostra, uma vez que as instituições de ensino são também elas organizações.

Este estudo incide na satisfação do curso e na utilidade da aprendizagem da programação. Com a investigação de resultados obtidos através de um inquérito realizado segundo os modelos de aceitação do modelo de tecnologia (Fishbein & Ajzen, 1991) e o comportamento planeado (Ajzen, 1991; Davis, 1993). Foi possível observar que a amostra de alunos é maioritariamente do género masculino (81.8%), sendo o seu local de estudo em Lisboa (50%).

A maioria dos alunos frequenta o 2º Ano de Licenciatura (33.53%) e as três linguagens de programação de que têm mais conhecimentos são Java (88.82%), C (87.65%) e assembly (65,29%).

O modelo estrutural explica 43% da variação da expectativa, da utilidade e da satisfação. A satisfação é explicada pelo modelo estrutural em 63% pela variação da utilidade percebida do curso e pela confirmação. A utilidade percebida do curso é explicada em 53% pela confirmação. A expectativa é explicada em 15% diretamente pelo suporte académico e indiretamente pela reputação da universidade.

Com a análise dos resultados foram validadas as seguintes hipóteses de investigação:

- **Hipótese 1 (H1)** - *Há uma relação positiva entre a reputação percebida e o suporte académico.*

A reputação percebida tem um efeito positivo sobre o suporte académico com um alto nível de significância positiva.

- **Hipótese 2 (H2)** - *Há uma relação positiva entre o suporte académico e expectativa.*

O suporte acadêmico tem um efeito positivo sobre a expectativa, com um alto nível de significância positiva.

- **Hipótese 3 (H3)** - *Há uma relação positiva entre suporte acadêmico e a confirmação.*

O suporte acadêmico tem um efeito positivo sobre a confirmação com um baixo nível de significância positiva.

- **Hipótese 4 (H4)** - *Há uma relação positiva entre o desempenho individual e a confirmação.*

O desempenho individual tem um efeito positivo sobre a confirmação com um alto nível de significância positiva.

- **Hipótese 5 (H5)** - *Há uma relação positiva entre a confirmação e a utilidade.*

A confirmação tem um efeito positivo sobre a utilidade, com um alto nível de significância positiva.

- **Hipótese 6 (H6)** - *Há uma relação positiva entre a confirmação e a satisfação.*

A confirmação tem um efeito positivo sobre a satisfação, com um alto nível de significância positiva.

- **Hipótese 7 (H7)** - *Há uma relação positiva entre a utilidade e a satisfação.*

A utilidade tem um efeito positivo sobre a satisfação, com um médio nível de significância positiva.

- **Hipótese 8 (H8)** - *Há uma relação positiva entre a utilidade e a intenção de continuação.*

A utilidade tem um efeito positivo sobre a intenção de continuação, com um baixo nível de significância positiva.

- **Hipótese 9 (H9)** - *Há uma relação positiva entre a expectativa e a intenção de continuação.*

O expectativa tem um efeito positivo sobre a intenção de continuação, com um baixo nível de significância positiva.

- **Hipótese 10 (H10)** - *Há uma relação positiva entre a satisfação e a intenção de continuação.*

A satisfação tem um efeito positivo sobre a intenção de continuação, com um alto nível de significância positiva.

6 Conclusões

6.1 Conclusões

A informática, conseguiu juntar numa só ferramenta (computador) a capacidade de desenvolver e criar novas evoluções em vários ramos do trabalho, sociedade e ciências.

Com o avanço das tecnologias de informação, a procura de jovens programadores está em constante crescimento. Formar bons programadores é uma problemática das universidades que lecionam cursos com unidades curriculares de programação. Atualmente construir e consolidar as bases de conhecimento para se tornar num bom programador requer vontade e tempo por parte de quem ensina e de quem aprende. É um esforço que tem de ser conjunto, caso contrário vai traduzir-se num crescente aumento de insucessos dos estudantes.

Há necessidade de enfrentar vários problemas de entre os quais a falta de problemas prático que são essencialmente vocacionados para a vertente industrial, a falta de ginástica mental para a resolução de problemas, o tempo vs programação, a idade vs género vs programação que pode limitar em diversos aspetos como a dificuldade em dominar os conceitos base e ainda a motivação social que em certos casos delimita o sucesso dos alunos.

As universidades, enfrentam estes e outros problemas no ensino da programação mas começam a surgir algumas medidas para tentar combater estas problemáticas cada vez mais preocupantes. Os paradigmas de programação, são ferramentas que afetam positivamente o sucesso dos programadores. Todos os paradigmas de programação que existem são a programação aplicacional, programação concorrente, programação funcional, programação modelar, literatura da programação, programação lógica, Microprogramação (Flynn, 2003), programação estruturada, programação automática, lista de processos, programação máquina – Assembly, programação orientada a objetos - POO, programação de sistemas. O CS2013 é a mais recente série de diretrizes curriculares. Devido ao fato do campo da computação crescer e se diversificar existiu a necessidade de criar várias recomendações curriculares. Existem neste momento vários volumes curriculares para Engenharia da Computação, Sistemas de Informação, Tecnologia da Informação, Engenharia de Software e Ciência da Computação. Estes volumes curriculares são atualizados regularmente com o objetivo de manter os currículos da computação modernos e relevantes. Com este estudo realizado em Portugal

junto de alunos de licenciaturas de informática, foi possível validar um modelo teórico proposto que explica em 43% as intenções dos alunos na continuidade dos estudos de informática. Essa intenção de continuidade resulta fortemente do nível de satisfação que sentem em relação aos cursos que frequentam. A intenção também é devida ao grau de utilidade, que os alunos percebem, que o curso lhes vai proporcionar, bem como também é devida pela expectativa positiva em relação ao curso. O grau de satisfação sentido pelos alunos é explicado pela utilidade percebidas e pela confirmação positiva do seu desempenho individual académica. O desempenho individual académico é um fator de grande importância na explicação da confirmação positiva que por sua vez leva a maior satisfação do aluno.

A presente dissertação tem algumas limitações, nomeadamente não foi realizado o estudo cruzado entre o que é ensinado nas universidades e as normas internacionais recomendadas pelas principais associações da área. Como limitação temporal registou-se também a necessidade de recolher mais respostas.

Assim como principais contribuições desta dissertação podem ser enumeradas as seguintes a identificação bibliográfica sobre as dificuldades de aprendizagem da programação. O estudo de estado de arte em Portugal sobre os cursos universitários que lecionam programação. A proposta e validação empírica de um modelo que explica os fatores determinantes que levam o aluno a ter intenção de continuar a estudar programação em contexto universitário.

7 Referências Bibliográficas

- ACM/IEEE. (2013, June 22). ACM/IEEE. (2013). Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. ACM. Retrieved June 22, 2014, from <http://www.acm.org/education/CS2013-final-report.pdf>
- Ajzen, I. (1991). THE THEORY OF PLANNED BEHAVIOR.
- Barker, L. J., McDowell, C., & Kalahar, K. (2009). Exploring factors that influence computer science introductory course students to persist in the major. *ACM SIGCSE Bulletin*, 41(1), 153–157.
- Bhattacharjee, A. (2001). Understanding information systems continuance: an expectation-confirmation model. *MIS Quarterly*, 351–370.
- Carter, J., & Jenkins, T. (1999). Gender and programming: What's going on? In *ACM SIGCSE Bulletin* (Vol. 31, pp. 1–4). ACM. Retrieved from <http://dl.acm.org/citation.cfm?id=305824>
- Cody, W. J. (2003). Approximation theory. Retrieved from <http://dl.acm.org/citation.cfm?id=1074131>
- Costa, C. J., Aparicio, M., & Cordeiro, C. (2012). A solution to support student learning of programming. In *Proceedings of the Workshop on Open Source and Design of Communication* (pp. 25–29). ACM. Retrieved from <http://dl.acm.org/citation.cfm?id=2316942>
- Davis. (1993). User acceptance of information technology: system characteristics, user perception and behavioral impacts.

Denning, P. J. (2003). Computer science.

Engenharia Informática | Faculdade de Ciências da Universidade de Lisboa. (n.d.). Retrieved June 22, 2014, from <http://www.fc.ul.pt/pt/pagina/3328/engenharia-inform%C3%A1tica>

Ensino | Técnico Lisboa. (n.d.). Retrieved June 22, 2014, from <http://tecnico.ulisboa.pt/pt/ensino/>

Fishbein, & Ajzen. (1991). Theory of Reasoned Action as Applied to Moral Behavior: A Confirmatory Analysis.

Flynn, M. (2003). Microprogramming. In *Encyclopedia of Computer Science* (pp. 1169–1170). UK. Retrieved from http://delivery.acm.org/10.1145/1080000/1074603/p1169-flynn.pdf?ip=193.136.189.5&id=1074603&acc=ACTIVE%20SERVICE&key=2E5699D25B4FE09E.544572B3C38C6BD2.4D4702B0C3E38B35.4D4702B0C3E38B35&CFID=611334776&CFTOKEN=64488745&__acm__=1421495868_d444879f8f2f997992b99f06bd31bb9c

Fornel, L. (1981). Evaluating structural equation models with Unobservable Variables and measurement error.

Fuqua, P., Slagle, J. R., & Gini, M. L. (2003). List processing. Retrieved from <http://dl.acm.org/citation.cfm?id=1074544>

Gomes, A., Henriques, J., & Mendes, A. (2008). Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. *Educação, Formação & Tecnologias-ISSN 1646-933X*, 1(1), 93–103.

- Gomes, A., & Mendes, A. J. (2007). Learning to program-difficulties and solutions. In *International Conference on Engineering Education–ICEE* (Vol. 2007). Retrieved from <http://ineer.org/Events/ICEE2007/papers/411.pdf>
- Halpern, M. (2003). Machine-and assembly-language programming. Retrieved from <http://dl.acm.org/citation.cfm?id=1074561>
- Hemmendinger, D. (2003). Concurrent programming. Retrieved from <http://dl.acm.org/citation.cfm?id=1074281>
- ISCTE-IUL. (n.d.). Retrieved June 22, 2014, from http://www.iscte-iul.pt/cursos/licenciaturas.aspx?utm_source=Google&utm_medium=Licenciaturas&utm_campaign=Extention
- ISEL - ADEETC - Candidatos. (n.d.). Retrieved June 22, 2014, from <http://www.adeetc.isel.pt/pt/candidatos>
- Karshmer, A. I. (2003). Front-end processor. Retrieved from <http://dl.acm.org/citation.cfm?id=1074414>
- Kopanidis, F. Z. (2008). *An investigation of undergraduate choice behaviour of a preferred program, discipline and university: a conceptual model*. Royal Melbourne Institute of Technology. Retrieved from <https://researchbank.rmit.edu.au/eserv/rmit:6153/Kopanidis.pdf>
- Kowalski, R., & Clark, K. L. (2003). *Logic programming*. John Wiley and Sons Ltd. Retrieved from <http://dl.acm.org/citation.cfm?id=1074553>
- Lowry, M. R. (2003). Automatic programming. Retrieved from <http://dl.acm.org/citation.cfm?id=1074152>

- Murphy, L., Richards, B., McCauley, R., Morrison, B. B., Westbrook, S., & Fossum, T. (2006). Women catch up: gender differences in learning programming concepts. In *ACM SIGCSE Bulletin* (Vol. 38, pp. 17–21). ACM. Retrieved from <http://dl.acm.org/citation.cfm?id=1121350>
- Narten, T. (2003). Systems programming. Retrieved from <http://dl.acm.org/citation.cfm?id=1074850>
- Reilly, E. D. (2003). Structured programming. Retrieved from <http://dl.acm.org/citation.cfm?id=1074835>
- Rogers, E. M. (1983). *Diffusion of innovations* (3rd ed). New York : London: Free Press ; Collier Macmillan.
- Rogers, E. M. (1995). DIFFUSION OF INNOVATIONS.
- Rumbaugh, J. (2003). Object-oriented analysis and design (OOAD). Retrieved from <http://dl.acm.org/citation.cfm?id=1074650>

Anexos

Anexo A: Modelo de medida Resultados Overview

	AVE	Composite Reliability	R Square	Cronbachs Alpha	Communality	Redundancy
Acad	0.6148	0.8266	0.1824	0.7090	0.6148	0.0922
Conf	0.7756	0.9540	0.7769	0.9421	0.7756	0.0327
Cont_Int	0.7076	0.8784	0.4336	0.7922	0.7076	0.0682
EX	0.7248	0.9404	0.1545	0.9238	0.7248	0.1116
DI	0.8484	0.9711	0.0000	0.9642	0.8484	0.0000
PU	0.9074	0.9751	0.5394	0.9658	0.9074	0.4895
RP	0.7032	0.9043	0.0000	0.8597	0.7032	0.0000
Sat	0.8270	0.9745	0.6293	0.9701	0.8270	0.4854

Anexo B: Cross Loadings

	Acad	Conf	Cont_Int	EX	DI	PU	RP	Sat
A2	0.7804	0.1945	0.1234	0.3699	0.1247	0.0282	0.2035	0.0588
A3	0.7166	0.1821	0.0998	0.2163	0.1191	0.0706	0.1759	0.1099
A4	0.8496	0.3446	0.1809	0.3244	0.3693	0.1612	0.5092	0.2598
C1	0.3360	0.8911	0.5480	0.3538	0.7808	0.6916	0.4686	0.7314
C2	0.1809	0.8495	0.4912	0.1971	0.7519	0.5864	0.3840	0.6258
C3	0.3451	0.9017	0.5378	0.3173	0.8137	0.6675	0.4449	0.7180
C4	0.3017	0.8631	0.5799	0.3486	0.7956	0.6655	0.4297	0.7054
C5	0.2807	0.8725	0.5022	0.2592	0.7348	0.5427	0.3881	0.6399
C6	0.2791	0.9046	0.5271	0.2771	0.7617	0.7109	0.4047	0.7005
EX1	0.3270	0.2615	0.3125	0.8315	0.2937	0.2903	0.2567	0.2670
EX2	0.3731	0.2322	0.2805	0.8663	0.2343	0.1795	0.2032	0.1949
EX3	0.3725	0.3156	0.3687	0.8811	0.3730	0.3429	0.2520	0.3206
EX4	0.3392	0.1807	0.2518	0.8005	0.2446	0.2361	0.1640	0.2298
EX5	0.3447	0.3050	0.3022	0.8867	0.3626	0.2851	0.1582	0.2735
EX6	0.2454	0.4055	0.3884	0.8389	0.4496	0.4029	0.1905	0.4290
Int1	0.1322	0.4989	0.8913	0.3170	0.5481	0.4378	0.3679	0.5432
Int3	0.1395	0.5091	0.8722	0.3066	0.5754	0.4616	0.3028	0.6108
Int5	0.1985	0.5271	0.7534	0.3289	0.5279	0.4270	0.3775	0.3955
DI1	0.3037	0.7878	0.6190	0.3889	0.9181	0.6912	0.5411	0.7594
DI2	0.2475	0.8135	0.5637	0.3295	0.9089	0.6348	0.4501	0.6702
DI3	0.3101	0.8481	0.6587	0.4103	0.9438	0.6853	0.5170	0.7679
DI4	0.3196	0.7811	0.5869	0.3433	0.8900	0.6129	0.4553	0.7228
DI5	0.2066	0.8053	0.5702	0.3228	0.9242	0.6452	0.4232	0.7102
DI6	0.2640	0.8182	0.6128	0.3301	0.9405	0.6896	0.4781	0.7750
PU1	0.1267	0.6940	0.5112	0.3316	0.6886	0.9664	0.3338	0.6559
PU2	0.1480	0.7155	0.5090	0.3321	0.6736	0.9557	0.3826	0.6209
PU3	0.1522	0.7086	0.5184	0.3099	0.6953	0.9697	0.3815	0.6448
PU4	0.0481	0.6803	0.4597	0.3267	0.6730	0.9176	0.3172	0.6134
R1	0.3399	0.3512	0.3363	0.1878	0.4298	0.3087	0.8928	0.3636
R2	0.4296	0.4892	0.4068	0.3089	0.4995	0.3511	0.7801	0.4409
R3	0.2927	0.3770	0.2811	0.0613	0.3941	0.3075	0.8036	0.3182
R4	0.3351	0.3517	0.3172	0.2001	0.3837	0.2622	0.8726	0.3421
S1	0.2034	0.7913	0.5874	0.3133	0.8087	0.6886	0.4360	0.9123
S2	0.1839	0.7553	0.5970	0.3534	0.7890	0.6405	0.4500	0.9233
S3	0.2442	0.7338	0.5809	0.3416	0.7653	0.6002	0.4727	0.9025
S4	0.2264	0.7340	0.5519	0.3170	0.7447	0.5737	0.4243	0.9001
S5	0.1265	0.6669	0.5769	0.2633	0.6854	0.5983	0.3553	0.9159
S6	0.1501	0.6562	0.5410	0.2827	0.6732	0.5669	0.3615	0.9116
S7	0.1695	0.6685	0.5682	0.2903	0.6596	0.6015	0.3749	0.9124
S8	0.1914	0.6638	0.5243	0.2757	0.6514	0.5570	0.3598	0.8969

Anexo C: Path Coefficients

	Acad	Conf	Cont_Int	EX	DI	PU	RP	Sat
Acad	0.0000	0.0725	0.0000	0.3930	0.0000	0.0000	0.0000	0.0000
Conf	0.0000	0.0000	0.0000	0.0000	0.0000	0.7345	0.0000	0.6364
Cont_Int	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
EX	0.0000	0.0000	0.1642	0.0000	0.0000	0.0000	0.0000	0.0000
DI	0.0000	0.8571	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
PU	0.0000	0.0000	0.1629	0.0000	0.0000	0.0000	0.0000	0.1980
RP	0.4271	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Sat	0.0000	0.0000	0.4598	0.0000	0.0000	0.0000	0.0000	0.0000

Anexo D: Latent variable Scores

	Acad	Conf	Cont_Int	EX	DI	PU	RP	Sat
Case 0	3	3	3	3	3	3	3	3
Case 1	6	5.9914	7	5.3376	6.1761	5.7494	6.7543	5.8823
Case 2	1	5.8341	7	6	6.1706	7	2	7
Case 3	4.3616	5.1663	6	4.3539	4.664	6	3.9755	4.8739
Case 4	1	2.1685	6.261	5.5181	2.1689	3.0136	1	5.6094
Case 5	5.7035	5.6786	7	5.5236	6.5004	4.9975	6.2665	6.7397
Case 6	4.6808	6.5042	5.9918	5	6.5012	6.2532	5.242	6
Case 7	5	4.8315	5.3727	5.0192	5.3298	5.7468	5	5.4877
Case 8	4.6808	5.6652	4.2381	4.6871	5	6	5	6
Case 9	2.3419	3	5.5221	3.6914	4.0089	3.9899	3	3.9928
Case 10	3.0511	6	7	6	7	6	7	6
Case 11	4.7006	4.679	6.3727	5.0462	5.5076	6	5.242	5
Case 12	4.6639	5.4957	4.7372	4.98	5.4995	7	6.2637	5.8746
Case 13	4.6639	4.3344	5.1346	4.2148	4.6658	5.2542	5.758	4.8779
Case 14	2.3674	3.0169	2.1182	2.6689	3	4	5	3.5082
Case 15	4.661	5	6.0082	5	5	4	5.4892	5
Case 16	2.3558	3.6656	6.0147	3	4.3306	5.2542	3.5048	5.3547
Case 17	4.6808	5.333	6.2545	3.7138	4.6729	6.4962	4.9717	4.5082
Case 18	3.2541	2.6678	6.2545	7	3.0188	5.5073	3	3.0124
Case 19	4.7035	3.501	7	4.8506	5.6702	2.2532	6.758	5.3532
Case 20	5.7035	5.6768	6.2463	5.8351	6.0054	6.4962	6	5.8738
Case 21	1.3419	4.3385	6.6273	5.1687	4.3404	5	5.758	4.2628
Case 22	1	5.0001	5.8818	4.7078	4.3237	6.7494	5	4
Case 23	5	6.1685	4	5	5	7	5	5
Case 24	2.339	3.3432	4.7537	4	4	4.4952	4.2509	5
Case 25	4.6581	5.3492	5.5074	4.5226	4.6702	5.5073	3.9717	5.2634
Case 26	4.068	4.1685	5.5156	6.3329	4.8364	4.7458	7	4.3876
Case 27	6	3	5.619	4	3	3	6	1.9835

	Acad	Conf	Cont_Int	EX	DI	PU	RP	Sat
Case 28	4	4.4936	5.2545	4.8303	5.3423	5	5.7528	5.8739
Case 29	5.0058	3.864	6.5074	4.672	3.8392	6.2628	4	5.0125
Case 30	4.0651	6.166	6.0082	5.494	6	6.5073	6	5.2793
Case 31	4.0029	4.6729	6.0082	4.5481	4.1768	4.2542	3.4929	4
Case 32	1	5.3385	4.381	4.0174	4.4968	6	4.4915	5
Case 33	4.068	5.1356	6.2545	6.0769	6.0083	4.0126	7	4.2812
Case 34	4.6581	2.6763	4.6273	4.6702	3.0116	4.2542	6	2.8959
Case 35	3.6384	4.1756	5.381	4.98	5.1616	5.0111	4.7543	4.3645
Case 36	5.0169	4.3237	4.7537	4.1886	4.3325	4.5073	4	4.1314
Case 37	4.6808	4.5024	6.381	5	5.3325	6.758	6	5.487
Case 38	1	5.3442	5.4844	5.2788	5.6605	7	5	4.517
Case 39	4	6	7	6.169	6.664	7	7	6
Case 40	2.9971	5.3249	6.5074	4.7325	6.3298	5.7468	5.9986	6.6139
Case 41	4.3134	3.2872	6.5074	3.5815	4.6577	4.5159	4.4453	2.7585
Case 42	3.3361	5.4993	5.8801	5.1904	5.6605	7	6	5
Case 43	3.3361	5.8291	7	4	5.6656	7	6.2226	5.9835
Case 44	3.3558	4.186	6.261	6	3.5085	7	5.4877	4.6247
Case 45	3.3419	3	2.381	3	3	4	4	3
Case 46	2.3587	3.8449	5.8883	4	4	6	2.9986	4
Case 47	2.6779	5.4844	6.6273	3.9534	5.8401	5.5063	3.7052	5.0393
Case 48	1	5.4743	7	5.4877	6	7	3.968	6.7558
Case 49	4.0256	3.5043	6.261	5.8424	4.5093	5	6.032	5.7598
Case 50	4.0227	4.5069	5.7537	5.4549	4.5093	4.4927	4.9963	4.1221
Case 51	2.3616	3.8315	5.6273	4	4	4	3.2406	4
Case 52	3.3674	4.8455	5.8883	5.3129	5.6702	5	6	6
Case 53	4.0256	6.3271	7	4.6693	6	7	5.4863	6
Case 54	1	1	2.619	4	2	2.2532	3.758	1.4886
Case 55	1	2.3271	7	5.1909	3.3503	6	5.5085	4.0007
Case 56	1	5.4824	7	1.9808	6.3334	7	7	7
Case 57	5	4.8315	6.381	4.6169	4.0062	5	6	5.9835
Case 58	5.7035	6.1685	7	6	6	7	6.758	6
Case 59	2	2.0086	4.0147	1.8433	1	3.2532	4.2637	2
Case 60	4.0622	4.8484	5.5221	6.8313	5	5.0025	4.9538	5.7289
Case 61	3.9744	4.3518	6	5.8313	4.6658	6	5	3.897
Case 62	5.6837	6.8337	7	6.6915	6.6613	7	7	6.0047
Case 63	2.7035	5.8063	7	6	7	7	6.274	5.7588
Case 64	2.3616	2.6428	3.2463	5.1497	3.3326	5.7569	2.758	1.1314
Case 65	4.0425	7	7	5.7068	6	7	6	7
Case 66	2.3674	6.3406	7	5.973	6.3341	6.758	6	6.636
Case 67	4.3419	4.4922	5.7619	5	4.3306	7	5.4735	5.4918
Case 68	4.3616	4.6729	7	5.6671	5.5012	6.5073	5.516	6.2386

	Acad	Conf	Cont_Int	EX	DI	PU	RP	Sat
Case 69	3.0511	6	5.1364	3.7182	5.1689	7	7	4
Case 70	4.661	4	6.381	4	4	4	6.0245	4
Case 71	4.6808	6.4874	7	5.6616	5.8221	7	6.2457	6.2262
Case 72	6.0425	5.315	6.6273	6.1904	5.6675	6.4937	7	5.3836
Case 73	4.661	5.3276	7	5.6668	5.8311	7	5.7543	5.6248
Case 74	2.0227	5.5132	7	4.4978	5.4995	7	6	6.4918
Case 75	3.339	3	5.7537	3	3	3	6	2.1341
Case 76	5.661	6.3348	5.3874	5.8558	5.5005	6.2532	5.7528	6.5078
Case 77	6.3616	7	6.5074	7	7	6.2653	7	7
Case 78	2	4.3519	6.5074	5.831	5	7	5.516	4.1554
Case 79	4.0227	4.5123	6.7537	6.001	5.6736	7	5.2637	5.6194
Case 80	5	4.9799	5.8736	5.4967	6	7	5.7349	6
Case 81	5.7035	5.1952	6.619	6.4981	5.8408	6.2532	3.4811	3.4113
Case 82	1.339	4.6473	7	4.4962	5	6	5	4.2828
Case 83	2.9744	4.1551	5	4.4789	4.1636	5	4	4
Case 84	4.325	6	6.7537	1	6	7	7	6
Case 85	2.7006	4.3271	6.2381	4.1687	4	7	4.7811	4
Case 86	4	5.5141	6.381	5.8558	6	7	6	5.5257
Case 87	4.3843	5.5043	5.5156	3.0773	5	5.5038	6	4.1313
Case 88	3.3419	4.8649	7	5.6671	6	7	6.2509	6.5082
Case 89	2.0169	1	4	3.0689	1	1	5.0075	1.3752
Case 90	3.3674	5.8449	6	5.6715	5.6728	6	6	6
Case 91	2.3674	7	5.8801	3.5821	7	7	5.7335	6.6248
Case 92	4.0453	5.1577	5.5221	6.1657	5.6702	6.0036	5.4929	6
Case 93	4.6808	6	7	5.5185	5	7	6	5.3892
Case 94	4	6.3303	7	4.6614	4.664	4	4	4
Case 95	2.0256	2.4957	5.5221	5	4	7	6	5
Case 96	1.6581	4.1573	4.8818	3.0466	3.5102	5.9914	5.4929	3.5247
Case 97	6	6	6.381	6	6	6	6	6
Case 98	3.3616	7	7	4.7555	5.8311	7	5.475	6
Case 99	5.4041	5.6642	7	6.0466	5.5032	6.7494	7	6.4926
Case 100	5.3419	5.8372	6.6273	5.5455	6.504	7	5.9948	5.7629
Case 101	3	5.9943	7	4.689	5.8302	7	6.2457	7
Case 102	4.3558	1	1	7	1	4	1	1
Case 103	2.3192	5.8337	6.7537	5	5.5023	6	5.7386	5
Case 104	1	1	1	1	1.3395	2.0126	4.4929	1
Case 105	5	5.8449	6.2545	6.3129	5.6648	7	7	5.6284
Case 106	3.3901	6.8449	6.7537	6.5673	6.8294	7	5.4826	7
Case 107	5.0198	6	4.8965	6	6	7	6	5.5082
Case 108	4.068	6.02	7	7	7	7	7	7
Case 109	4.2907	6.3237	7	4.8794	5	7	4.9806	6.1066

	Acad	Conf	Cont_Int	EX	DI	PU	RP	Sat
Case 110	6.0227	6.172	7	6	6.6702	7	6.758	6
Case 111	3.7006	3.5012	2.5074	6	3.3407	4	6	3.9828
Case 112	3.6837	5.4957	6.6273	5.6653	5.3272	7	5.242	5.7345
Case 113	5.3052	5.7032	7	5.1372	6.4907	7	6.516	7
Case 114	6.6808	6.3385	6.6273	6.169	6.336	6.4962	6.4877	6.5074
Case 115	5.3419	4	5.1346	4	4	4	6.274	4
Case 116	2.7093	6.1551	5.6355	5.6959	6.8302	6.5073	6	5.5139
Case 117	4.6837	5	7	4.3809	5	7	5.5085	5.2631
Case 118	1.6808	4.6678	5.3727	5.3591	5.0054	7	6	5.0055
Case 119	5.3192	5.172	4.3727	6	5.3342	6	6	5.6367
Case 120	3.0453	2.8305	6.6273	7	3.8346	7	7	3
Case 121	5.6808	4.3694	7	7	4.6605	7	5.7543	4.2526
Case 122	1	4	7	6.3329	4	4	4.2383	6.0028
Case 123	2.6581	4.8132	6.6273	5.7082	5.4924	4.7605	5.026	5.8655
Case 124	4.7431	7	7	6.4981	7	7	6.0171	7
Case 125	4.0425	4.679	5.8883	4	5	6.5038	5.5057	5.8794
Case 126	1	3.172	5.642	4.5085	5.3271	5.2774	5.2548	5.9143
Case 127	4.7262	3	7	5	4	4	6.274	3.402
Case 128	5.6779	7	7	7	7	7	3.9717	4
Case 129	2.7093	3.9978	6.261	4	4.8266	3.5184	4.4646	5
Case 130	3.3419	6	6.7537	6	6	6	5.4877	5
Case 131	4.0482	4.5127	6.381	5.2138	5.4916	7	4.7269	5.7258
Case 132	5	5.679	6.6273	6.1435	5.8408	6.5073	4.7491	5.8763
Case 133	5	5.679	6.6273	6.1435	5.8408	6.5073	4.7491	5.8763
Case 134	5.6779	4.4761	7	5.5695	4	4	5.484	4.7346
Case 135	5.3645	3.8146	6.381	5.3543	5	4.2481	6.2637	4
Case 136	2	5.6678	6.6273	6	6	6.2532	5.758	5.8835
Case 137	7	1	3.6273	7	1	1.4962	4.2152	1.8686
Case 138	3.3843	6.1551	6.6273	6	6	6.2532	6.4929	5.2331
Case 139	3	4	6.381	4	4	4	5.516	4
Case 140	3.7093	4.6965	6.1346	3.7196	4.8496	7	7	5.6279
Case 141	1.6779	6	6.5074	3.6933	6	7	4.7335	6.8794
Case 142	5.3419	4.5105	5.642	6	5	6	5.268	5.4958
Case 143	3.3419	6.0113	6.7537	6.5029	5.8364	7	6	7
Case 144	5	4.6652	5.0082	4	4	4	5.4877	5
Case 145	4	4.8365	7	5.7377	5.3387	6.0025	5.5123	5.133
Case 146	4.3192	5.9888	6.261	5.5429	6	7	4.758	4.6288
Case 147	1	3.5043	5.642	5	3	5	4	4
Case 148	4.3872	6.828	7	4.3702	6.6694	7	6.2665	6.2567
Case 149	2.6808	2.8449	3.7537	2.3591	3	4	3.0037	3
Case 150	4.0087	5.6813	5.8883	5.1487	5.6702	7	6.2651	6.1277

	Acad	Conf	Cont_Int	EX	DI	PU	RP	Sat
Case 151	4.3616	6.1747	7	5.8572	5.6702	7	6.0283	7
Case 152	5.339	5.3317	6.7537	5.004	5.1901	4.9843	6	6.1505
Case 153	3.6442	4.3275	4.7537	1	5	5.7468	4.5183	4.2592
Case 154	3.986	5.3505	4.7537	5.5473	6.0008	7	6.2665	6
Case 155	6	5.1867	7	6	5.8347	7	7	5.9945
Case 156	4	6	7	6	3.5075	6	6.4877	6
Case 157	4.3221	5.8449	7	5.3583	6.6667	7	4.7297	6.6344
Case 158	3.6837	5.3379	7	5.6915	5.3369	6	6.2457	5.9865
Case 159	5.661	5.679	7	6.6679	5.1698	6.7458	5.7349	6.3656
Case 160	5.9744	7	7	7	7	7	7	7
Case 161	1	5.4724	5.8883	5.5717	5.8302	7	4.7566	6.6367
Case 162	5.3163	5.8449	6.7537	6	5.6729	6.758	6.7349	6
Case 163	6	6	6.619	6.1497	7	7	6	7
Case 164	1	5.6568	7	2.4789	6.1503	7	5.0245	5.9835
Case 165	3.3645	3.3159	4.2463	5.3376	3.6668	3.5073	5.758	2.7658
Case 166	4	4.6362	4	4	4	6.2739	4.9806	7
Case 167	4.3616	6	7	7	7	7	6	6
Case 168	5.339	5.3492	6	5.831	5.8337	6.4952	5.9821	6
Case 169	4.6808	4.851	5.1264	4.5284	4.6667	5	5.4877	5.1221

Anexo F: Total Effects

	Acad	Conf	Cont_Int	EX	DI	PU	RP	Sat
Acad	0.0000	0.0725	0.0993	0.3930	0.0000	0.0532	0.0000	0.0567
Conf	0.0000	0.0000	0.4791	0.0000	0.0000	0.7345	0.0000	0.7818
Cont_Int	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
EX	0.0000	0.0000	0.1642	0.0000	0.0000	0.0000	0.0000	0.0000
DI	0.0000	0.8571	0.4106	0.0000	0.0000	0.6295	0.0000	0.6701
PU	0.0000	0.0000	0.2540	0.0000	0.0000	0.0000	0.0000	0.1980
RP	0.4271	0.0310	0.0424	0.1678	0.0000	0.0227	0.0000	0.0242
Sat	0.0000	0.0000	0.4598	0.0000	0.0000	0.0000	0.0000	0.0000

Anexo G: Outer Loadings (Bootstrap)

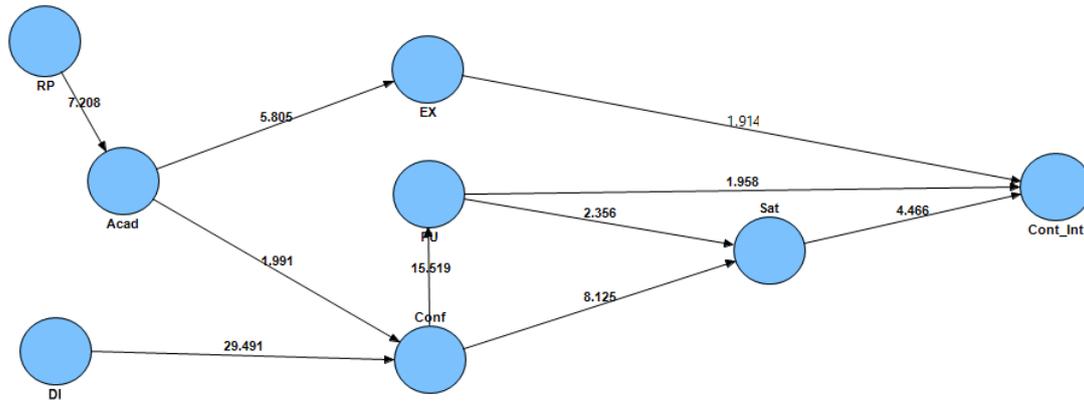
	Original Sample (O)	Sample Mean (M)	Standard Deviation (STDEV)	Standard Error (STERR)	T Statistics (O/STERR)
A2 <- Acad	0.7804	0.7743	0.0614	0.0614	12.719
A3 <- Acad	0.7166	0.7075	0.0759	0.0759	9.4471
A4 <- Acad	0.8496	0.8507	0.0351	0.0351	24.1951
C1 <- Conf	0.8911	0.8883	0.0184	0.0184	48.5201
C2 <- Conf	0.8495	0.847	0.0265	0.0265	32.0611
C3 <- Conf	0.9017	0.9001	0.0197	0.0197	45.802

	Original Sample (O)	Sample Mean (M)	Standard Deviation (STDEV)	Standard Error (STERR)	T Statistics (O/STERR)
C4 <- Conf	0.8631	0.8609	0.0287	0.0287	30.0865
C5 <- Conf	0.8725	0.8717	0.0264	0.0264	33.0761
C6 <- Conf	0.9046	0.903	0.0274	0.0274	33.0029
EX1 <- EX	0.8315	0.8327	0.0433	0.0433	19.1912
EX2 <- EX	0.8663	0.8654	0.0261	0.0261	33.1718
EX3 <- EX	0.8811	0.8813	0.0206	0.0206	42.7921
EX4 <- EX	0.8005	0.7982	0.0446	0.0446	17.9686
EX5 <- EX	0.8867	0.8838	0.0242	0.0242	36.5778
EX6 <- EX	0.8389	0.8366	0.0504	0.0504	16.6328
Int1 <- Cont_Int	0.8913	0.8876	0.026	0.026	34.3323
Int3 <- Cont_Int	0.8722	0.8699	0.0279	0.0279	31.3098
Int5 <- Cont_Int	0.7534	0.7511	0.055	0.055	13.7051
DI1 <- DI	0.9181	0.9161	0.018	0.018	51.027
DI2 <- DI	0.9089	0.9068	0.02	0.02	45.3622
DI3 <- DI	0.9438	0.9425	0.0115	0.0115	81.7621
DI4 <- DI	0.89	0.8867	0.024	0.024	37.0364
DI5 <- DI	0.9242	0.9225	0.0181	0.0181	51.0483
DI6 <- DI	0.9405	0.9388	0.0126	0.0126	74.9104
PU1 <- PU	0.9664	0.9657	0.0092	0.0092	104.9817
PU2 <- PU	0.9557	0.955	0.0111	0.0111	86.2798
PU3 <- PU	0.9697	0.9695	0.0076	0.0076	128.1676
PU4 <- PU	0.9176	0.9162	0.0236	0.0236	38.8818
R1 <- RP	0.8928	0.8884	0.0292	0.0292	30.5659
R2 <- RP	0.7801	0.7791	0.0526	0.0526	14.822
R3 <- RP	0.8036	0.7978	0.0524	0.0524	15.3291
R4 <- RP	0.8726	0.8667	0.0341	0.0341	25.6197
S1 <- Sat	0.9123	0.9106	0.0155	0.0155	58.8347
S2 <- Sat	0.9233	0.9219	0.0155	0.0155	59.4546
S3 <- Sat	0.9025	0.8997	0.0198	0.0198	45.5709
S4 <- Sat	0.9001	0.8978	0.0199	0.0199	45.2113
S5 <- Sat	0.9159	0.913	0.0161	0.0161	56.7562
S6 <- Sat	0.9116	0.9086	0.02	0.02	45.5363
S7 <- Sat	0.9124	0.9095	0.0197	0.0197	46.2395
S8 <- Sat	0.8969	0.8934	0.0217	0.0217	41.3207

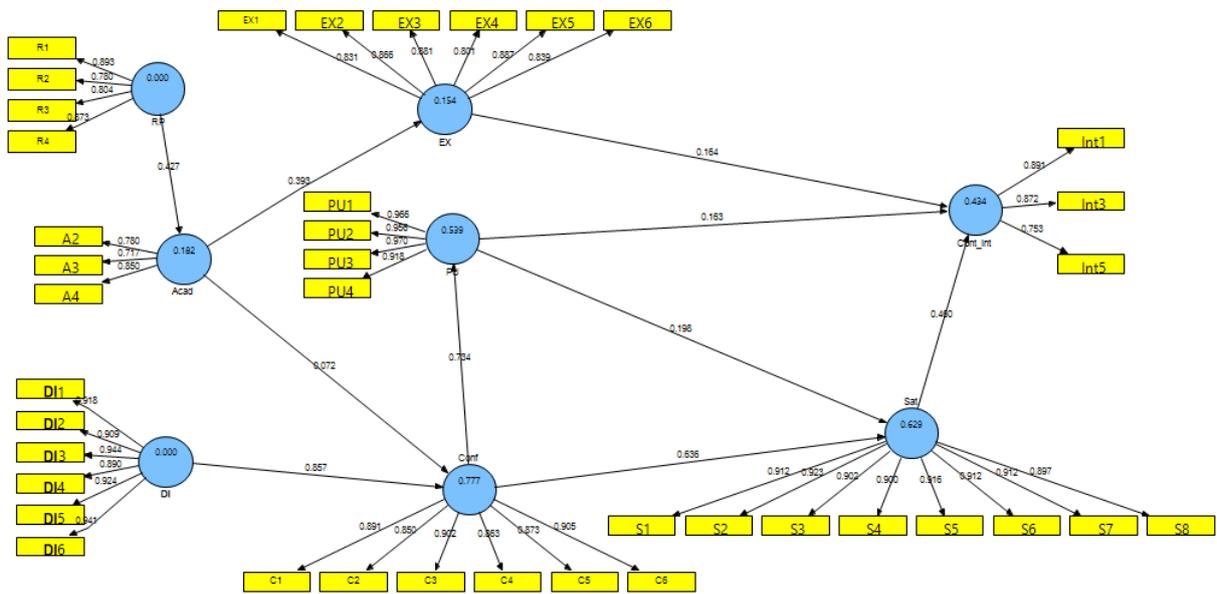
Anexo H: Total Effects (Bootstrap)

	Original Sample (O)	Sample Mean (M)	Standard Deviation (STDEV)	Standard Error (STERR)	T Statistics (O/STERR)
Acad -> Conf	0.0725	0.0771	0.0366	0.0366	1.9789
Acad -> Cont_Int	0.0993	0.1044	0.0382	0.0382	2.5988
Acad -> EX	0.393	0.3972	0.0661	0.0661	5.9482
Acad -> PU	0.0532	0.0562	0.0264	0.0264	2.0171
Acad -> Sat	0.0567	0.0596	0.0277	0.0277	2.0493
Conf -> Cont_Int	0.4791	0.4709	0.0718	0.0718	6.6772
Conf -> PU	0.7345	0.7314	0.0451	0.0451	16.2684
Conf -> Sat	0.7818	0.778	0.0411	0.0411	19.0122
EX -> Cont_Int	0.1642	0.1702	0.0821	0.0821	1.9992
DI -> Conf	0.8571	0.8532	0.0293	0.0293	29.2421
DI -> Cont_Int	0.4106	0.4028	0.0689	0.0689	5.9601
DI -> PU	0.6295	0.6246	0.0509	0.0509	12.3632
DI -> Sat	0.6701	0.6645	0.0505	0.0505	13.2566
PU -> Cont_Int	0.254	0.2531	0.0883	0.0883	2.8759
PU -> Sat	0.198	0.2013	0.0856	0.0856	2.3123
RP -> Acad	0.4271	0.4394	0.0579	0.0579	7.3722
RP -> Conf	0.031	0.0336	0.0162	0.0162	1.9135
RP -> Cont_Int	0.0424	0.046	0.0184	0.0184	2.3009
RP -> EX	0.1678	0.1742	0.0359	0.0359	4.6763
RP -> PU	0.0227	0.0245	0.0118	0.0118	1.9211
RP -> Sat	0.0242	0.026	0.0123	0.0123	1.9614
Sat -> Cont_Int	0.4598	0.4476	0.0975	0.0975	4.716

Anexo I: Printscreen do Bootstrap



Anexo J: Printscreen do PLS



Anexo K: Printscreen do questionário

Considerações acerca do ensino superior.

O presente questionário insere-se no âmbito de uma investigação académica e tem como objectivo estudar as opiniões dos alunos de ensino superior que ingressam em cursos de informática.

A sua resposta irá dar um importante contributo à academia e permitirá compreender melhor os factores que influenciam o grau de satisfação dos alunos. Todos os dados recolhidos encontram-se ao abrigo de um escrupuloso critério de confidencialidade e anonimato. O questionário levará aproximadamente 10 minutos a ser respondido.

Muito obrigado pela sua colaboração.

***Required**

Género *

- Masculino
 Feminino

Idade *

Região / Distrito de proveniência *

Região / Distrito do curso *

Ano que frequenta *

Escolaridade da Mãe *

- Ensino básico
 Ensino secundário
 Licenciatura
 Pós-graduação
 Doutoramento
 Other:

Escolaridade do Pai *

- Ensino básico
 Ensino secundário
 Licenciatura
 Pós-graduação
 Doutoramento
 Other:

Área de escolaridade 12^o *

- Ciências e tecnologias
 Humanidades
 Artes
 Desporto
 Economia
 Formação profissional

Linguagens de programação que tem conhecimento *

- Java
- C
- assembly
- Python
- ASP
- PHP
- C++
- HTML / JavaScript
- Nenhuma
- Other:

1 - Eficácia *

1 - Discordo totalmente 7 - Concordo totalmente

	1	2	3	4	5	6	7
Tenho a certeza que vou entender o conteúdo da programação seja ela fácil ou difícil.	<input type="radio"/>						
Eu não me sinto confiante para entender todos os conceitos de programação.	<input type="radio"/>						
Eu tenho a certeza que vou ter sucesso na programação.	<input type="radio"/>						
Por mais que me esforce, não sou capaz de aprender programação.	<input type="radio"/>						
Quando as atividades de programação são muito difíceis, eu desisto ou só faço as partes fáceis.	<input type="radio"/>						
Durante os exercícios de programação, prefiro pedir as respostas a outras pessoas em vez de pensar por mim mesmo.	<input type="radio"/>						
Quando eu encontro conteúdo programação difícil, não me esforço por aprendê-lo.	<input type="radio"/>						

2 - Importância da reputação na sua escolha: *

1 - Nada importante 7 - Muito importante

	1	2	3	4	5	6	7
Reputação da universidade	<input type="radio"/>						
Ajustamento ao curso	<input type="radio"/>						
Reputação do curso	<input type="radio"/>						
Prestigio e status da universidade	<input type="radio"/>						

3 - Na minha escolha da universidade tive em consideração estes aspectos: *

1 - Nada importante 7 - Muito importante

	1	2	3	4	5	6	7
Custo das propinas	<input type="radio"/>						
Recursos da universidade (biblioteca, laboratórios de informática, e / ou salas de aula, Wi-Fi).	<input type="radio"/>						
Actividades extra curriculares e outras infra-estruturas disponíveis.	<input type="radio"/>						
Experiência e as qualificação dos professores.	<input type="radio"/>						

4 - Entrada *

1 - Nada importante 7 - Muito importante

	1	2	3	4	5	6	7
Os requisitos de entrada são importantes na escolha do curso.	<input type="radio"/>						
Considera importante a gama de cursos disponíveis.	<input type="radio"/>						
Considera importante o tipo de instituição (ex. Universidade, Politécnico)	<input type="radio"/>						
Considera importante localização do universidade/instituto.	<input type="radio"/>						

5 - Família *

1 - Nada importante 7 - Muito importante

	1	2	3	4	5	6	7
A opinião da família foi importante na escolha do curso.	<input type="radio"/>						
Eu escolhi o curso sua percentagem de empregabilidade	<input type="radio"/>						

6 - Intenção de continuar *

1 - Nada importante 7 - Muito importante

	1	2	3	4	5	6	7
É minha intenção continuar com o curso.	<input type="radio"/>						
Se eu pudesse, eu interrompia o curso.	<input type="radio"/>						
Tenho intenção de continuar a aprofundar os conhecimentos de programação	<input type="radio"/>						
Se eu pudesse, deixava de estudar programação.	<input type="radio"/>						
Eu tenho intenção de continuar a estudar nesta universidade/instituto	<input type="radio"/>						
Eu não pretendo continuar a estudar nesta universidade/instituto	<input type="radio"/>						

7 - Como me sinto em relação à minha experiência no curso: *

1 - Nada importante 7 - Muito importante

	1	2	3	4	5	6	7
Estou satisfeito em relação à minha experiência no curso.	<input type="radio"/>						
Estou agradado com a minha experiência no curso.	<input type="radio"/>						
É gratificante a minha experiência no curso.	<input type="radio"/>						
Estou muito entusiasmado com a minha experiência no curso.	<input type="radio"/>						
Estou satisfeito em relação à minha experiência na programação.	<input type="radio"/>						
Estou agradado com a minha experiência na programação.	<input type="radio"/>						
É gratificante a minha experiência na programação.	<input type="radio"/>						
Estou muito entusiasmado com a minha experiência na programação.	<input type="radio"/>						

8 - Utilidade *

1 - Discordo totalmente 7 - concordo totalmente

	1	2	3	4	5	6	7
Este curso melhorou o meu desempenho na programação	<input type="radio"/>						
Este curso aumentou a minha produtividade na programação.	<input type="radio"/>						
Este curso aprimorou a minha eficácia na programação.	<input type="radio"/>						
No geral, o curso é útil na programação	<input type="radio"/>						

9 - Confirmação *

1 - Discordo totalmente 7 - Concordo totalmente

	1	2	3	4	5	6	7
A minha experiência com o curso superou as minhas expectativas.	<input type="radio"/>						
O nível de suporte prestado no curso foi melhor do que eu esperava.	<input type="radio"/>						
No geral, as minhas expectativas foram confirmadas na realização do curso.	<input type="radio"/>						
A minha experiência nas disciplinas de programação superou as minhas expectativas.	<input type="radio"/>						
O nível de suporte prestado na disciplinas de programação foi melhor do que eu esperava.	<input type="radio"/>						
No geral, as minhas expectativas foram confirmadas na realização das disciplinas de programação.	<input type="radio"/>						

10 - Tinha no início expectativas elevadas com: *

1 - Discordo totalmente 7 - Concordo totalmente

	1	2	3	4	5	6	7
A experiencia no curso	<input type="radio"/>						
O suporte prestado no curso	<input type="radio"/>						
O geral do curso	<input type="radio"/>						
A minha experiencia das disciplinas do curso	<input type="radio"/>						
O suporte prestado na disciplina de programação	<input type="radio"/>						
O geral da disciplina de programação	<input type="radio"/>						

11- Foi atingido um bom desempenho: *

1 - Discordo totalmente 7 - Concordo totalmente

	1	2	3	4	5	6	7
A experiencia no curso	<input type="radio"/>						
O suporte prestado no curso	<input type="radio"/>						
O geral do curso	<input type="radio"/>						
A minha experiencia das disciplinas do curso	<input type="radio"/>						
O suporte prestado na disciplina de programação	<input type="radio"/>						
O geral da disciplina de programação	<input type="radio"/>						

Observações