

ENSINO DE PROGRAMAÇÃO EM CURSOS DE ENGENHARIA: INTERFACES CONSOLE X INTERFACES GRÁFICAS

Sergio Vicente Denser Pamboukian - sergiop@mackenzie.br

Raquel Cymrot - raquelc@mackenzie.br

Lincoln Cesar Zamboni - lincoln.zamboni@mackenzie.br

Oswaldo Ramos Tsan Hu - osvaldo.hu@mackenzie.br

Edson de Almeida Rego Barros - edson.barros@mackenzie.br

Universidade Presbiteriana Mackenzie

Rua da Consolação, 930

CEP 01302-907 - Consolação - São Paulo - SP – Brasil

Resumo: *Este trabalho apresenta uma visão geral de como os conceitos de Computação, Algoritmos e Programação são ensinados atualmente nos cursos de engenharia. Discute alguns dos diversos paradigmas de programação (estruturada, orientada a objetos e orientada a eventos) e os tipos de interfaces com o usuário mais comuns (console e gráfica). Descreve a forma como as disciplinas de computação são ministradas na EEUPM (Escola de Engenharia da Universidade Presbiteriana Mackenzie) e como as mesmas são complementadas por atividades de extensão como, por exemplo, fóruns de discussão presenciais. Analisa os resultados de uma pesquisa de satisfação efetuada com os alunos dos fóruns de Interfaces Gráficas realizados em 2009 e 2010.*

Palavras-chave: *Computação, Programação, Interfaces, Educação.*

1 INTRODUÇÃO

As ferramentas auxiliares para a resolução de problemas de Engenharia evoluíram muito nos últimos anos. Até a década de 1960, profissionais de engenharia utilizavam ferramentas como régua de cálculo e tabelas de logaritmos para elaborar seus projetos. A partir da década de 1970, estes instrumentos foram gradativamente substituídos pelas calculadoras eletrônicas.

Atualmente, com o aparecimento e a rápida evolução dos microcomputadores, as calculadoras também estão sendo gradualmente substituídas pelos mesmos. As maiores vantagens do uso do computador em substituição às calculadoras são a rapidez com que os cálculos são efetuados, a precisão destes cálculos e principalmente a capacidade que os computadores têm de executar automaticamente tarefas e cálculos repetitivos poupando tempo e trabalho aos seus usuários.

No mercado atual, não existe mais espaço para empresas que desenvolvam projetos sem o auxílio de computadores, pois os custos são elevados e os prazos são muito longos. Desta forma, o computador se tornou um instrumento indispensável para empresas que desejam competir neste mercado. Os alunos de graduação em engenharia devem ter esta visão da importância da programação para a vida acadêmica e para a carreira profissional.

O Conselho Nacional de Educação (CNE) no artigo 4º de sua resolução nº11 de 2002 (CONSELHO NACIONAL DE EDUCAÇÃO, 2002) trata das habilidades gerais do Engenheiro e, dentre as inúmeras lá citadas, quase todas estão direta, ou indiretamente,

relacionadas com disciplinas que envolvem computação, algoritmos e programação. A mesma resolução em seu artigo 6º trata dos núcleos de conteúdos e em tais núcleos encontramos relacionamentos com disciplinas que também envolvem computação, algoritmos e programação. O Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP) em sua portaria nº 160 de 2005 (INSTITUTO NACIONAL DE ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA, 2005) cita diretamente as mesmas habilidades da resolução anterior quanto à aplicação de conteúdo do Exame Nacional de Desempenho de Estudantes (ENADE). O Conselho Federal de Engenharia, Arquitetura e Agronomia (CONFEA) no Anexo II de sua resolução nº 1.010 de 2005 (CONSELHO FEDERAL DE ENGENHARIA, ARQUITETURA E AGRONOMIA, 2005) salienta os campos de atuação profissional do Engenheiro em suas várias modalidades e estes são inevitavelmente controlados e automatizados por computadores.

Desta forma, os cursos de engenharia devem possuir obrigatoriamente disciplinas relacionadas aos conceitos de Computação, Algoritmos e Programação. Em geral, o enfoque destas disciplinas está na Lógica de Programação e sua aplicação direta em uma Linguagem de Programação Científica (C++, Pascal e outras). Em alguns cursos o aprendizado é feito através da Programação Estruturada (PE) e a geração de Aplicações Console e, em outros cursos, são ensinados os conceitos de Programação Orientada a Objetos (POO) e Programação Orientada a Eventos (POE) utilizando Interfaces Gráficas. No caso específico da Escola de Engenharia da Universidade Presbiteriana Mackenzie (EEUPM), alguns cursos tem enfoque apenas na PE e aplicações Console, enquanto outros cursos possuem os conceitos de POO, POE e Interfaces Gráficas. Para os cursos que possuem apenas enfoque na PE são oferecidas atividades de extensão (opcionais e gratuitas) como fóruns de discussão presenciais para que os alunos aprendam os conceitos de POO, POE e Interfaces Gráficas.

1.1 Objetivos

Este trabalho tem como objetivos:

- descrever sucintamente algumas formas de ensino de programação nas escolas de engenharia e, em particular, na Escola de Engenharia da Universidade Presbiteriana Mackenzie (EEUPM);
- salientar as características e aplicações dos diferentes tipos de interface (Console e Gráfica) e dos diferentes paradigmas de programação (PE, POO e POE);
- avaliar os resultados de uma pesquisa de satisfação realizada com os participantes dos Fóruns de Interfaces Gráficas oferecidos pela EEUPM.

2 INTERFACES HOMEM-MÁQUINA

É chamado de interface o meio utilizado pelo usuário para se comunicar com a máquina e realizar suas tarefas. Uma interface pode ser definida como o componente responsável pela tradução das ações do usuário em uma ou mais solicitações à funcionalidade da aplicação. Também é responsável por fornecer ao usuário um *feedback* sobre os resultados e conseqüências de sua ação.

No princípio, a interface do computador com os usuários era o próprio *hardware*, mas com o passar do tempo as interfaces foram evoluindo e passando por diversas etapas: interfaces puramente textuais, interfaces com menus, interfaces orientadas a janelas e, finalmente, as interfaces gráficas que conhecemos hoje.

As interfaces gráficas utilizadas atualmente na maioria das aplicações tornam o *software* mais atraente, mais fácil de aprender e mais fácil de utilizar. Para o desenvolvimento destas aplicações, os projetistas utilizam uma série de diretrizes que são essenciais para a criação de

um *software* amigável. Algumas destas diretrizes, descritas por diversos autores (PREECE, 2005; PRESSMAN, 2011; SOMMERVILLE, 2007; HECKEL, 1993), são:

- manter a consistência da aplicação, permitindo que o usuário generalize o seu conhecimento a respeito do comportamento da mesma;
- manter coerência entre a aplicação e a interface do ambiente operacional. Desta maneira, ao iniciar a aprendizagem de um novo *software* o usuário terá que se preocupar em aprender apenas o modelo de funcionamento da nova aplicação, pois a interface padronizada do ambiente já será de seu conhecimento;
- fornecer a realimentação (*feedback*) adequada às ações do usuário através de modificação de objetos da interface, mensagens coerentes, sinais sonoros e outros recursos;
- projetar adequadamente o sistema de prevenção e tratamento de erros para que o usuário se sinta seguro na utilização da aplicação;
- estruturar a interface baseada no conhecimento do usuário, sua formação, sua personalidade, seu comportamento e outros fatores;
- utilizar metáforas para facilitar o entendimento da interface (lixeira, por exemplo);
- utilizar elementos gráficos (imagens, ícones) em substituição ao texto sempre que possível para facilitar a memorização, o reconhecimento e o melhor entendimento da interface.

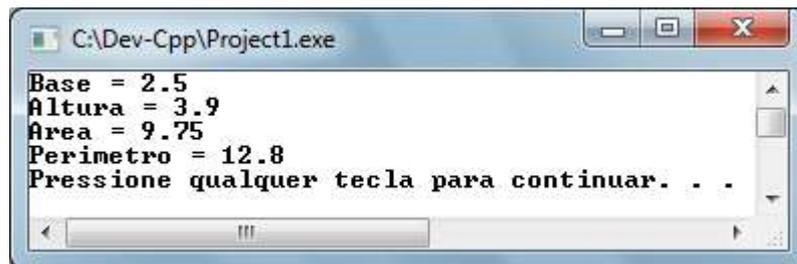


Figura 1 – Aplicação simples com Interface Console.

A “Figura 1” mostra uma aplicação simples com Interface Console que calcula a área e o perímetro de um retângulo.

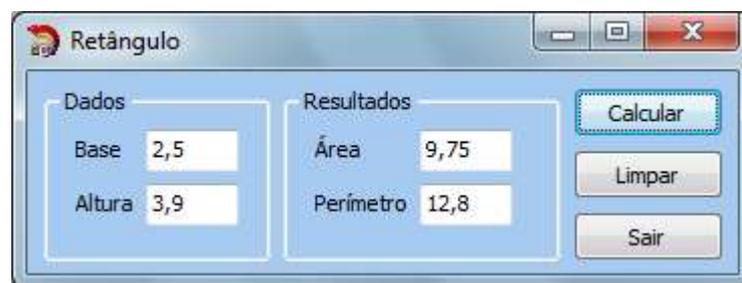


Figura 2 – Aplicação simples com Interface Gráfica.

A “Figura 2” mostra a mesma aplicação utilizando uma Interface Gráfica. Podemos notar que as duas aplicações, embora utilizem interfaces distintas, resolvem o problema proposto. Porém a Interface Gráfica se mostra muito mais atrativa e fácil de interagir.

3 ENSINO DE PROGRAMAÇÃO EM CURSOS DE ENGENHARIA

Pode-se observar que, atualmente, os cursos de engenharia utilizam diversos recursos para o ensino de Lógica de Programação como algoritmos, fluxogramas e diagramas N-S (Nassi-Schneiderman) para Programação Estruturada (PE) e UML (*Unified Modelling Language*) para Programação Orientada a Objetos (POO).

As principais Linguagens de Programação utilizadas para o ensino de Programação Estruturada com Interface Console são C, C++ e Pascal. Para o ensino de Programação Orientada a Objetos e Programação Orientada a Eventos tem-se utilizado ambientes integrados de desenvolvimento (IDE – *Integrated Development Environment*) para as linguagens Object Pascal (Delphi), C++ (C++ Builder e Visual C++) e Java (NetBeans e Eclipse).

Na Escola de Engenharia da Universidade Presbiteriana Mackenzie (EEUPM) utiliza-se a seguinte seqüência de aprendizagem:

- Programação Estruturada com Interface Console e Linguagem C++ (compilador Dev-C++);
- Programação Orientada a Objetos utilizando Interface Console e Linguagem C++ (compilador Dev-C++);
- Programação Orientada a Objetos e Programação Orientada a Eventos com Interface Gráfica utilizando C++ (ambiente integrado C++ Builder).

O ensino de Lógica de Programação tem início com a Programação Estruturada, pois é um estilo de programação que facilita o entendimento dos conceitos básicos de programação (estruturas seqüenciais, condicionais e repetitivas, funções e procedimentos, vetores e matrizes, etc.). A interface Console também é utilizada por ser bem simples e permitir que o aluno se concentre no código da aplicação e não na sua aparência. Em uma segunda etapa, são introduzidos os conceitos de Programação Orientada a Objetos (classes, objetos, encapsulamento, hereditariedade, polimorfismo, etc.) ainda com aplicações Console. O enfoque nesta etapa é o reuso de classes através da criação de bibliotecas. A etapa seguinte é a introdução dos conceitos de Programação Orientada a Eventos e o desenvolvimento de Interfaces Gráficas utilizando bibliotecas de componentes já existentes no ambiente integrado de programação.

Estas etapas são oferecidas de maneira diferenciada para os diversos cursos da EEUPM. A Engenharia Elétrica possui o curso de programação mais completo, abrangendo todos os conceitos descritos anteriormente em quatro disciplinas regulares. Outros cursos, como as Engenharias Civil, Mecânica e de Materiais, possuem apenas os conceitos básicos de Programação Estruturada com Interfaces Console. Para complementar o aprendizado destes, em geral, são oferecidas atividades de extensão gratuitas na forma de Fóruns de Discussão Presencial abrangendo conceitos de POO, POE e Interfaces Gráficas.

4 FÓRUNS TÉCNICOS SOBRE INTERFACES GRÁFICAS EM C++

Para suprir as necessidades de alguns cursos da EEUPM, foram oferecidos nos anos de 2009 e 2010, dois Fóruns de Discussão Presenciais gratuitos. Cada fórum era constituído de dez encontros presenciais semanais com duração de duas horas com turmas reduzidas (cerca de 20 alunos) em Laboratório de Informática utilizando o ambiente de programação C++ Builder.

Os fóruns foram apoiados pelo Ambiente Virtual de Aprendizagem (AVA) gratuito Moodle (MOODLE, 2010), no qual foram utilizados recursos para disponibilização de material didático, entrega de exercícios, avaliação, questionário, pesquisa e outros.

Com o objetivo de avaliar a aceitação destes fóruns, foi realizada com os alunos aprovados uma pesquisa de opinião após o encerramento dos mesmos. A metodologia e a análise dos resultados desta pesquisa são discutidos a seguir.

4.1 Metodologia

Para estudar a percepção dos alunos quanto aos resultados obtidos devido a sua participação no Fórum de C++ e sua importância no contexto acadêmico e profissional foi construído um instrumento de pesquisa em forma de questionário. As primeiras oito questões caracterizavam o respondente, enquanto as de números 9 a 25 focaram as variáveis específicas para análise deste evento de extensão. Na questão 26 foi dada a oportunidade ao respondente de comentar algo mais sobre esta pesquisa ou sobre o Fórum de C++.

Após a elaboração do questionário, a pesquisa foi submetida e aprovada pela Comissão de Ética da Escola de Engenharia da Universidade Presbiteriana Mackenzie.

A população sobre a qual foi realizada a inferência é a população amostrada e não a população-alvo, uma vez que esta não inclui os elementos que não responderam à solicitação de preenchimento e envio do questionário por meio da Plataforma Moodle. Nestas condições é necessário procurar as possíveis diferenças entre a população amostrada e população-alvo. Utilizou-se a variável gênero para validar a amostra. A amostragem, portanto, não foi probabilística. Entretanto, como a seleção da amostra independeu do pesquisador, esta pode ser considerada criteriosa (BOLFARINE; BUSSAB, 2005).

Depois de encerrado o prazo para envio das respostas, os dados foram obtidos por meio da própria plataforma Moodle que os fornece de forma já tabelada no formato ASCII. A análise estatística foi então realizada construindo-se tabelas e gráfico de Boxplot e realizando-se inferências a partir do cálculo de intervalos com 95% de confiança para médias e proporções de interesse e realização de testes de hipótese pertinentes.

O Boxplot é um gráfico que apresenta o menor valor que não é uma observação discrepante, o primeiro quartil, a mediana, o terceiro quartil e o maior valor que não é observação discrepante, dando uma visão tanto das medidas de posição, como da variabilidade envolvida (MONTGOMERY; RUNGER, 2009).

O intervalo de confiança para a média foi calculado após teste de aderência à distribuição Normal e o para cálculo de intervalo de confiança para proporção foi utilizado o método exato (NEWCOMBE, 1998). Devido ao pequeno tamanho da amostra os testes de hipóteses de aderência à distribuição Normal e de independência entre pares de variáveis aleatórias pertinentes foram realizados utilizando-se um nível de significância de 10%.

Como as suposições para uso do teste de independência Quiquadrado não foram respeitadas, utilizou-se o teste exato de Fisher o qual é usado para testar independência entre pares de variáveis aleatórias com dois níveis cada uma (tabela de contingência 2 x 2). O teste exato de Fisher é baseado na distribuição hipergeométrica. Calcula-se a probabilidade do resultado ocorrido acrescida das probabilidades dos resultados ainda mais extremos. Este será o nível descritivo do teste e a hipótese de independência das variáveis aleatórias deverá ser rejeitada se seu valor for inferior ao nível de significância do teste (SIEGEL; CASTELLAN, 2008).

Os dados foram analisados com o auxílio do programa estatístico Minitab, disponível no Laboratório de Simulação e Processos, do prédio 6 da Escola de Engenharia da Universidade Presbiteriana Mackenzie.

4.2 Resultados

A única variável disponível para caracterizar o respondente na população foi o gênero, variável esta utilizada para validar a amostra. A hipótese de independência entre as variáveis

gênero e ter participado da pesquisa, testada por meio do teste de independência de Fisher, não foi rejeitada ($P = 0,459$), sugerindo não haver vício de amostragem, embora a amostra não tenha sido sorteada da população.

Vinte alunos concordaram em participar da pesquisa e enviaram o questionário preenchido. Destes 15 (75%) são do gênero masculino. Quanto ao curso de origem, 13 (65%) são alunos do curso de Engenharia Elétrica, 5 (25%) de Engenharia Civil e 2 (10%) de Engenharia de Produção.

Quanto ao semestre, tem-se que os alunos cursavam majoritariamente o 3º semestre (14 alunos – 70%), dois alunos cursavam o 5º semestre (10%) e houve um aluno de cada semestre, 2º, 4º, 7º e 8º, correspondendo cada um a 5% da amostra.

Quanto à caracterização da amostra 65,00% foi formada por alunos do curso de Engenharia Elétrica (I.C. = [40,78; 84,61]) e 75,00% era de alunos que cursavam entre a 1ª e a 3ª etapa do curso (I.C. = [50,90; 91,34]).

A porcentagem de alunos que acreditam ser possível em algumas situações concorrer com outras empresas do mercado sem utilizar os recursos computacionais em seus projetos de Engenharia foi igual a 20,00% (I.C. = [5,73; 43,66]). Pode-se perceber que as ferramentas computacionais se tornaram imprescindíveis no desenvolvimento de projetos de engenharia.

A porcentagem de respondentes que indicou ser a Interface Gráfica importante ou muito importante como fator para motivar um usuário a gostar de utilizar um *software* foi igual a 95,00% (I.C. = [75,13; 99,87]) e para facilitar o aprendizado de um *software* e torná-lo mais fácil de usar foi igual a 90,00% (I.C. = [68,30; 98,77]). As aplicações console servem para resolver problemas específicos encontrados pelo engenheiro no seu dia a dia. Porém, quando se pensa no desenvolvimento de aplicações para usuários finais, a interface gráfica é de extrema importância.

Já ao se comparar o desenvolvimento de aplicações com Interfaces Gráficas com a criação de aplicações Console, 95,00% dos alunos indicaram ser mais fácil o trabalho com a utilização da Interface Gráfica (I.C. = [75,13; 99,87]). A princípio, o desenvolvimento de interfaces gráficas parece mais complicado do que o de interfaces console. Porém, conforme o programador vai se envolvendo com a linguagem e com o ambiente de programação, as tarefas se tornam cada vez mais fáceis devido à grande quantidade de componentes e funções já existentes.

Na avaliação direta do aproveitamento do fórum, 90,00% dos alunos (I.C. = [68,30; 98,77]) acharam que este foi importante ou muito importante para se perceber que os problemas matemáticos e de Engenharia podem ser resolvidos facilmente com o auxílio de uma Linguagem de Programação, criando um *software* amigável, fácil de aprender e fácil de utilizar. Coerentemente, 80,00% dos participantes (I.C. = [56,34; 94,27]) pretendem utilizar os conceitos aprendidos neste fórum para resolver seus futuros problemas de engenharia e elaborar seus projetos.

Ao serem indagados se as disciplinas de programação do curso de Engenharia em que eles estão matriculados são suficientes para fornecer todo o conhecimento necessário para a utilização adequada das linguagens de programação, a fim de resolver problemas/projetos durante o curso, 60,00% dos alunos responderam ser estas suficientes ou adequadas (I.C. = [36,05; 80,88]). Já ao responder se o curso de engenharia deveria ter uma disciplina regular sobre Interfaces Gráficas, 90,00% responderam afirmativamente (I.C. = [68,30; 98,77]).

Ao analisar o Fórum de C++, 65,00% dos alunos (I.C. = [40,78; 84,61]) acharam ser a carga horária, que foi de 20 horas, adequada. A totalidade dos que não a acharam adequada aumentaria a carga horária. Quanto ao método de ensino utilizado, 95,00% dos alunos (I.C. = [75,13; 99,87]) o acharam adequado ou muito adequado. De forma geral também 95,00% dos alunos (I.C. = [75,13; 99,87]) acharam que o fórum atendeu as expectativas, foi acima das

expectativas ou foi muito acima das expectativas, sendo que 60,00% dos alunos (I.C.= [36,05; 80,88]) acharam que o fórum foi acima ou muito acima das expectativas.

A nota média atribuída para o Fórum de C++, sendo 0 a menor valor de uma nota e 5 a maior valor, foi igual a 4,55. Como a amostra tem tamanho menor que 30, não se pode utilizar o Teorema do Limite Central para supor a distribuição Normal da média da variável nota atribuída. Embora a distribuição das notas tenha sido a assimétrica (coeficiente de assimetria igual a menos um), ao se testar a aderência desta distribuição à distribuição Normal pelo método de Kolmogorov-Smirnov, a hipótese não foi rejeitada, ao nível de significância de 10% ($P > 0,15$), podendo-se então utilizar a distribuição t-de-Student para construir um intervalo com 95% de confiança para a nota média, a saber: I.C. = [4,27; 4,83]. Tal resultado indica uma grande satisfação com a participação neste fórum. Além disso, 100% dos participantes disseram que indicariam fóruns deste tipo para seus colegas. A “Figura 3” apresenta o gráfico boxplot para as notas atribuídas e ilustra o resultado acima descrito.

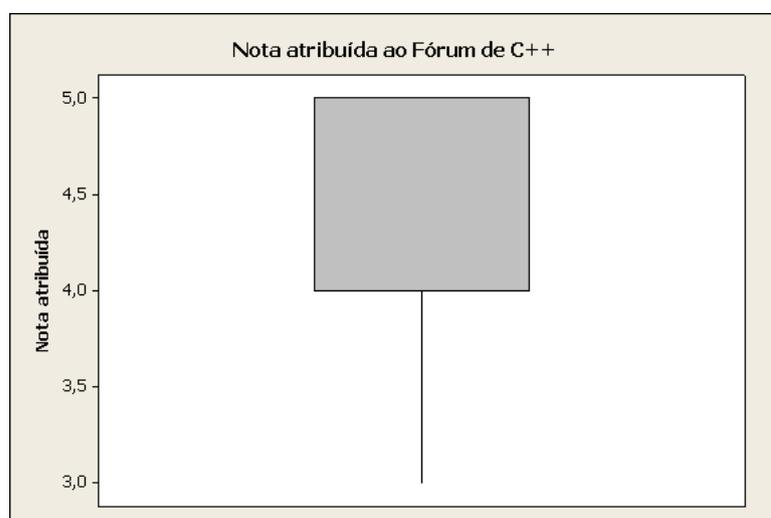


Figura 3 – Boxplot para notas atribuídas.

Ao nível de significância de 10% houve dependência entre as variáveis cursar Engenharia Elétrica e achar que as disciplinas de programação do curso de Engenharia em que eles estão matriculados são suficientes para fornecer todo o conhecimento necessário para a utilização adequada das linguagens de programação a fim de resolver problemas/projetos durante o curso ($P = 0,062$), indicando que proporcionalmente os alunos do curso de Engenharia Elétrica responderam mais afirmativamente serem as disciplinas matriculadas suficientes. Tal resultado vem de acordo com o fato de a Engenharia Elétrica dar mais ênfase às disciplinas de programação.

O teste de independência entre as variáveis cursar Engenharia Elétrica e achar que o seu curso de Engenharia deveria ter uma disciplina regular sobre Interfaces Gráficas teve nível descritivo igual a 0,111, superior portanto a 10%, levando a não rejeição da hipótese de independência. Como, entretanto, o tamanho da amostra foi muito pequeno ($n = 20$), há indicação de que novas pesquisas com tamanho maior de amostra possam confirmar a tendência aqui verificada de que alunos de outros cursos que não o de Engenharia Elétrica desejam proporcionalmente mais uma disciplina de regular sobre Interfaces Gráficas. Vale salientar que na EEUPM o único curso que possui uma disciplina regular de Interfaces Gráficas é a Engenharia Elétrica, porém nenhum aluno do fórum havia cursado a mesma.

Ao nível de significância de 10% houve também dependência entre as variáveis parte do curso na qual o aluno está majoritariamente matriculado (de 1ª a 3ª etapa ou de 4ª a 10ª) e

acreditar ser possível concorrer com outras empresas do mercado sem utilizar os recursos computacionais em seus projetos de Engenharia ($P = 0,032$), concluindo-se, ao nível de significância de 10%, que alunos que cursam etapas superiores a terceira proporcionalmente acharam mais ser possível concorrer com outras empresas do mercado sem utilizar os recursos computacionais em seus projetos de Engenharia. Isto pode ser devido ao fato dos alunos das primeiras etapas estarem mais envolvidos com as disciplinas de programação.

Ao se testar a independência entre a turma na qual o aluno participou do fórum e acreditar ser possível concorrer com outras empresas do mercado sem utilizar os recursos computacionais em seus projetos de Engenharia, a hipótese de independência foi rejeitada ($P = 0,061$), sendo que o alunos que participaram da primeira turma acharam mais ser possível concorrer com outras empresas do mercado sem utilizar os recursos computacionais em seus projetos de Engenharia. Tal resultado pode estar relacionado a outro. Também foi rejeitada a hipótese de independência entre turma na qual o aluno participou do fórum e parte do curso (de 1ª a 3ª etapa ou de 4ª a 10ª) na qual o aluno está majoritariamente matriculado ($P = 0,000$), sendo que o primeiro fórum teve proporcionalmente mais alunos de etapas superiores à terceira. Tal fato pode ter causado um confundimento, estando os últimos resultados em concordância com a conclusão já obtida de que alunos que cursam etapas superiores a terceira proporcionalmente acharam mais ser possível concorrer com outras empresas do mercado sem utilizar os recursos computacionais em seus projetos de Engenharia.

A variável frequência de participação nos encontros do fórum foi independente de todas as variáveis estudadas nesta pesquisa.

5 CONSIDERAÇÕES FINAIS

Utilizam-se muito as aplicações com Interface Console para que o aluno de engenharia, nas etapas iniciais do curso, possa focar mais as estruturas lógicas de uma determinada linguagem em detrimento as concepções visuais de uma aplicação com Interface Gráfica. Não se deve deixar, entretanto, de trazer ao aluno de engenharia as Interfaces Gráficas, uma vez que é um elemento motivador e agregador para o mesmo.

Em termos gerais, os fóruns sobre C++, valendo-se de Interfaces Gráficas, são de extrema importância para os estudantes de engenharia quanto à organização, manipulação e visualização de dados para seus projetos. Tais estudantes já possuem, mesmo os que não são de engenharia, uma pré-disposição para se motivarem mais no quesito visualização. Observa-se que a própria linguagem de programação é composta por elementos simbólicos visuais.

Tais fóruns trouxeram outra dimensão no aprendizado para o estudante de engenharia, principalmente no tocante à empolgação e motivação visual. Lembra-se aqui que aplicações com Interfaces Gráficas estão presentes em aplicações com Banco de Dados e Internet, não só em computadores, como também em outros equipamentos como *smartphones*, *tablets* etc.

REFERÊNCIAS BIBLIOGRÁFICAS

- BOLFARINE, H.; BUSSAB, W. O. **Elementos de amostragem**. ABE-Projeto Fisher, São Paulo: Edgard Blücher, 2005. 274 p.
- CONSELHO FEDERAL DE ENGENHARIA, ARQUITETURA E AGRONOMIA (CONFEA). **Resolução nº 1.010 de 2005**. Disponível em: <<http://www.abepro.org.br/arquivos/websites/1/1010-05.pdf>> Acesso em: 05 jun. 2011
- CONSELHO NACIONAL DE EDUCAÇÃO (CNE). **Resolução CNE/CES 11, de 11 de março de 2002**. Disponível em: <<http://portal.mec.gov.br/cne/arquivos/pdf/CES112002.pdf>> Acesso em: 05 jun. 2011.
- HECKEL, Paul. **Software Amigável**. Rio de Janeiro: Campus, 1993. 312 p.

- INSTITUTO NACIONAL DE ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA (INEP). **Portaria nº 160 de 2005**. Disponível em: <<http://www.pucrs.br/autoavaliacao/portaria160.pdf>> Acesso em: 05 jun. 2011
- MONTGOMERY, D. C.; RUNGER, G. C. **Estatística Aplicada e Probabilidade para Engenheiros**. 4 ed. Rio de Janeiro: LTC, 2009. 493 p.
- MOODLE. **Site dedicado ao software Moodle**, 2010. Disponível em: <<http://moodle.org/>> Acesso em: 05 jun. 2011
- NEWCOMBE, R. G. Two-sided confidence intervals for the single proportion: comparison of seven methods. **Statistics in Medicine** v. 17, p. 857-872, 1998. Disponível em: <<http://www.stats.org.uk/statistical-inference/Newcombe1998.pdf>>. Acesso em: 4 jun. 2011.
- PREECE, Jennifer; ROGERS, Yvonne; SHARP, Helen. **Design de Interação**. Porto Alegre: Bookman, 2005. 548 p.
- PRESSMAN, Roger S. **Engenharia de Software**. 7. ed. Mcgraw Hill – Artmed, 2011. 776 p.
- SIEGEL; S.; CASTELLAN JR., N. J. **Estatística não-paramétrica para ciências do comportamento**. Métodos de Pesquisa. 2 ed. Porto Alegre: Bookman, 2008. 448 p.
- SOMMERVILLE, Ian. **Engenharia de Software**. 8. ed. Addison Wesley Bra, 2007. 568 p.

TEACHING OF PROGRAMMING IN ENGINEERING COURSES: CONSOLE INTERFACES X GRAPHICAL INTERFACES

***Abstract:** This work presents a general vision of how Computation, Algorithms and Programming concepts are taught currently in engineering courses. It discusses some of the several programming paradigms (structured, object oriented and event oriented) and the types of more common user interfaces (console and graphical). It describes the way the courses of computation are given in the EEUPM (Escola de Engenharia da Universidade Presbiteriana Mackenzie) and how they are complemented by extension activities as, for example, discussion forums. It analyzes the results of a satisfaction research made with the students of the forums of Graphical interfaces accomplished in 2009 and 2010.*

Key-words: Computation, Programming, Interfaces, Education.