

O uso do Linux para entregas de trabalhos no sistema SuSy

Cristiano José Furlan, Gabriel Velasco Fernandes, Guilherme Masao Tsuyukubo, Leonardo Polizel, Lorenzo Chaves, Mateus Soraggi, Matheus Cumpian e Matheus Padovani.

Faculdade de Tecnologia – Universidade Estadual de Campinas

ST008 – Metodologia do Trabalho Científico

Abstract. *This work responds to a question of whether or not there is an advantage over the use of the Linux system in detriment of the Windows system for deliveries works of SI100 discipline in the SuSy system of Unicamp. By collecting data through the results of the codes sent in SuSy systems, we find that there is no difference in using a Linux system within the scope of the SI100 discipline, because the comparison of results was unbiased if only the standard C language libraries were used.*

Resumo. *Este trabalho responde a questão de se existe ou não vantagem da utilização do sistema Linux em detrimento do sistema Windows para entregas de trabalhos da disciplina SI100 no sistema SuSy da Unicamp. Por meio da coleta de dados referente aos resultados dos códigos enviados nos dois sistemas para o SuSy, verificamos que não existe diferenças entre utilizar um sistema Linux no âmbito dos trabalhos da disciplina, pois a comparação de resultados se mostrou imparcial se utilizados apenas as bibliotecas padrões da linguagem C.*

1. Introdução

No começo da vida acadêmica dos alunos de Sistemas de Informação e Análise e Desenvolvimento de Sistemas da Faculdade de Tecnologia da Unicamp, os alunos irão se deparar com a matéria de algoritmos e programação de computadores, conhecida como SI100, logo em seu primeiro dia de aula irão receber a recomendação de utilizar um sistema Linux durante o desenvolvimento da disciplina.

O sistema SuSy é uma ferramenta onde os alunos submetem os códigos das atividades realizadas na disciplina, onde o sistema efetua testes automáticos para avaliação do código do aluno, tal sistema está rodando em uma distribuição Linux utilizando o compilador GCC [1]. Tal fato pode indicar que alunos que utilizem um sistema Windows para fazer e testar seus códigos podem ter problemas com a compilação do código pelo sistema susy, já que o compilador MinGW (porte do gcc para windows) não oferece nativamente o suporte para compilação *cross-platform* [2].

Neste âmbito, esse trabalho tem como objetivo verificar se existe alguma vantagem na utilização de um sistema Linux para entregas de trabalho da disciplina SI100 no sistema SuSy.

2. Métodos

Durante o período de desenvolvimento do trabalho a pesquisa foi realizada utilizando dois tipos de sistemas operacionais, dois compiladores e o sistema SuSy da Unicamp, os sistemas operacionais usados foram o Ubuntu 16.04 e o Windows 10, como compiladores foram utilizados o MinGW64 (4.6.1) para Windows e o GCC 7.6.0 para Linux, ademais os uploads dos códigos produzidos para a disciplina foram enviados ao SuSy em sua versão 9.8.

Foram utilizados 12 códigos, referentes a 12 trabalhos realizados na matéria SI100B durante o período de desenvolvimento desta pesquisa, a estratégia utilizada para teste consistia em codificar e testar o código em um ambiente Windows (Windows 10), utilizando o compilador MinGW64 (4.6.1), após isso o código era submetido ao sistema SuSy e era observado o relatório de submissão, também utilizamos um ambiente Linux (Ubuntu 16.04) para programar as tarefas e testar o código com o compilador GCC 7.6.0, assim podemos comparar os resultados.

3. Resultados

Ao decorrer da disciplina SI100, nós coletamos resultados de códigos corretos enviados ao SuSy utilizando o Windows e também utilizando o Linux, assim temos uma visão sobre os resultados do SuSy para um aluno utilizando Windows e um utilizando Linux, segue a tabela de resultados:

| Trabalho | | Data Entrega | Resultado | | |
|----------|-------------------------|--------------|-----------|---------|--|
| | | | Linux | Windows | |
| Tópico | Enunciado | | GCC | MinGW | |
| Loops | Top4Ex1 | 11/abril | Ok | Ok | |
| Loops | Top4Ex2 | 11/abril | Ok | Ok | |
| Loops | Top4Ex3 | 11/abril | Ok | Ok | |

| | | | | | |
|-------------------|----------------------------|---------|----|-------------------|--|
| Strings | Top7Ex1 | 10/maio | Ok | Ok | |
| Strings | Top7Ex2 | 10/maio | Ok | Ok | |
| | | | | Erro ¹ | |
| Strings | Top7Ex3 | 10/maio | Ok | Ok ² | |
| Registros | Top8Ex1 | 17/maio | Ok | Ok | |
| Registros | Top8Ex2 | 17/maio | Ok | Ok | |
| Registros | Top8Ex3 | 17/maio | Ok | Ok | |
| Vetores | Trabalho 3 | 17/maio | Ok | Ok | |
| Modularizaçã o | Top9aEx 1 | 24/maio | Ok | Ok | |

Tabela 1 – Comparação de resultados

Como visto na tabela, obtivemos o mesmo resultado pelo sistema SuSy na grande maioria dos trabalhos realizados, porém no *Top7Ex3* tivemos um erro de compilação, pois usamos a função *strrev()*, que não é uma função padrão da biblioteca *string.h* da linguagem C [3], a biblioteca *string.h* do MinGW possui funções que não estão presentes no GCC, assim tivemos que fazer uma mudança no código para depois enviarmos novamente, e assim obtivemos um resultado positivo. Isso se dá ao fato do MinGW conter algumas funções não padrões da linguagem C, algo que deve ser alvo de atenção do aluno, pois a disciplina apenas pede funções padrões da linguagem nos trabalhos. Portanto, mostramos que no âmbito da disciplina SI100, não há vantagem no uso do sistema Linux (GCC) para envios de trabalhos ao sistema SuSy, o aluno da disciplina SI100 pode optar por utilizar o Windows sem que acarrete em dificuldades no envio dos trabalhos ao SuSy.

4. Discussão e conclusão

Os resultados dessa pesquisa mostram que a grande maioria dos códigos feitos e testados em um ambiente Windows foram dados como corretos pelo sistema SuSy, algo que também foi mostrado é que o MinGW não é exatamente equivalente ao GCC no âmbito da realização dos trabalhos da disciplina SI100, pois a biblioteca padrão *string.h* do MinGW contém funções que não estão presentes no GCC, podendo assim conduzir o aluno que está realizando seus trabalhos no Windows ao erro, já que o SuSy não aceita funções que não estão presentes nas bibliotecas padrões do GCC.

Uma das hipóteses levantadas antes dos experimentos foi a de que a codificação de caracteres padrão do Windows (Windows-1252), poderia influenciar na saída do programa, já que o SuSy utiliza como padrão o UTF-8, que tem a tabela ASCII como referência, guardando 1 byte para cada carácter da posição 0 a 127 da tabela [4], já o padrão Ansi utilizado pelo sistema Windows, pode guardar posições além da tabela ASCII com 1 byte de memória, equivalente a um CHAR da linguagem C. Então caso o

aluno esteja utilizando o padrão Windows-1252, ele poderá representar um 'Ç' utilizando um tipo char em C, algo que no GCC (UTF-8) não poderá representar na saída do programa [5]. Porém essa hipótese é inválida, já que a compilação dos códigos dessa disciplina feita pelo susy utiliza as seguintes flags do GCC: *-std=c99 -pedantic -Wall -lm -o*. Com a flag *-std=c99* o compilador irá aceitar linguagens de padrão ANSI[6] e usado em conjunto com a flag *-pedantic* ele irá ignorar certos caracteres que não estejam de acordo com o padrão[7], portanto o aluno não terá problemas quanto a codificação que está usando.

Ademais, para apontarmos precisamente se existe algum tipo de saída em que o aluno utilizando o Windows pode ser prejudicado pelo sistema SuSy, teríamos que realizar testes com saídas específicas, fugindo do escopo de trabalhos da disciplina SI100, porém não obtivemos permissão para criação de trabalhos no sistema pela Unicamp, algo que já tinha sido acordado anteriormente porém fomos barrados depois.

Portanto, podemos concluir com este trabalho que os alunos utilizando Linux não terão vantagens no âmbito da entrega de trabalhos da disciplina SI100 no sistema SuSy, os alunos que utilizam Windows poderão entregar seus códigos que estão corretos normalmente, desde que não utilizem funções que não estão contidas nas bibliotecas padrões do GCC.

Referências

- [1] Sistema SuSy - Instituto de Computação da UNICAMP. Disponível em: <<https://www.ic.unicamp.br/~susy/>>. Acesso em: 30 de maio de 2018.
- [2] Using MinGW for cross-hosted development. Disponível em: <http://www.mingw.org/Welcome_to_MinGW_org>. Acesso em: 30 de maio de 2018.
- [3] C – strrev() function. Disponível em: < <https://fresh2refresh.com/c-programming/c-strings/c-strrev-function/>>. Acesso em: 31 de maio de 2018.
- [4] UTF-8 Encoding. Disponível em: < <https://www.fileformat.info/info/unicode/utf8.htm>>. Acesso em: 31 de maio de 2018.
- [5] Entendendo o Unicode e os Characters Encodings. Disponível em: < <http://blog.caelum.com.br/entendendo-unicode-e-os-character-encodings/>>. Acesso em: 31 de maio de 2018.
- [6] Options Controlling C Dialect. Disponível em: <<https://gcc.gnu.org/onlinedocs/gcc/C-Dialect-Options.html#C-Dialect-Options>>. Acesso em: 31 de maio de 2018.
- [7] Some useful things to remember about the gcc compiler. Disponível em: <http://web.mit.edu/10.001/Web/Tips/tips_on_gcc.html> . Acesso em: 31 de maio de 2018.