



# AVALIAÇÃO – PRIMEIRO BIMESTRE

## Linguagem de Programação

Comandos de Entrada e Saída / Comandos de Condição / Laços de Repetição

Paranaguá, 04 de Maio de 2017

### Instruções para resolução da Avaliação:

1. A prova é resolvida de forma individual ou em dupla, não havendo troca de informações entre os alunos, nem mesmo troca de material como cadernos, livros ou *pen drive*.
2. As saídas dos programas a serem desenvolvidos devem, obrigatoriamente, ser idênticas as apresentadas nos exemplos de cada questão, caso contrário o programa não será considerado correto.
3. Não existem questões parcialmente corretas, ou seja, como cada questão equivale a 20% do conceito parcial, ou o aluno atinge esses 20% na questão ou esta é desconsiderada. O aluno deve escolher 5 das 6 questões para serem resolvidas;
4. O professor não tirará nenhuma dúvida com relação a possíveis erros de compilação ou sintaxe, nem mesmo com relação a interpretação das questões, por isso todas elas possuem exemplos de entrada com sua correspondente saída.
5. Para resolução das questões, devem ser utilizados, obrigatoriamente, apenas os conceitos de linguagem C vistos até o momento na disciplina, ou seja, vetores, funções, entre outros não serão considerados.
6. A cada questão resolvida o aluno ou a dupla de alunos deve chamar o professor para que o mesmo verifique a corretude do código-fonte e já contabilize a questão como resolvida.

## QUESTÕES

1. **(Fácil)** Codifique um programa em C que solicite ao usuário um caractere qualquer. Após isso crie a forma apresentada a seguir, utilizando para tal o caractere especificado pelo usuário. Vejo exemplo:

Obs.: não há necessidade do uso do `'\t'`

```
g113du4rd0@asus-ultrabook-g1l:~/Documentos/Prova/EMI x
Arquivo Editar Ver Pesquisar Terminal Ajuda
Digite um caractere: o
o          ooooooo
o          o      o
o          o      o
o          ooooooo
o          o
o          o
o          o
o          o
ooooo     o
[g113du4rd0@asus-ultrabook-g1l EMI]$
```

2. **(Fácil)** Faça um programa C que possua 4 variáveis locais inteiras, com os nomes `"v1"`, `"v2"`, `"v3"` e `"v4"`. Agora solicite que o usuário digite o valor inicial (entre 0-100)

da variável “v1”. Após isso, crie um código: contendo apenas 4 linhas; utilizando apenas as funções matemáticas “/” e “+” e os valores contidos nas variáveis; no qual ao final da sua execução os novos valores das variáveis sejam: v1=21, v2=28, v3=3 e v4=4 . Veja o exemplo:

```
g1l3du4rd0@asus-ultrabook-g1l:~/Documentos/Prova/EMI x
Arquivo Editar Ver Pesquisar Terminal Ajuda
Digite o valor de v1: 99

Valor de 'v1': 21
Valor de 'v2': 28
Valor de 'v3': 3
Valor de 'v3': 4

[g1l3du4rd0@asus-ultrabook-g1l EMI]$
```

3. **(Fácil)** Codifique um programa em C que solicite ao usuário três números. Considere que o usuário sempre digitará números variando entre 0-100. Após armazenar os três números apresente-os em ordem decrescente. Veja exemplo:

```
g1l3du4rd0@asus-ultrabook-g1l:~/Documentos/Prova/EMI x
Arquivo Editar Ver Pesquisar Terminal Ajuda
Digite três números: 80 95 23

Ordem decrescente: 95 80 23

[g1l3du4rd0@asus-ultrabook-g1l EMI]$
```

4. **(Médio)** Codifique um programa em C que receba um número inteiro e identifique se ele é perfeito. Os números perfeitos são números inteiros onde a soma de todos os seus divisores positivos (excluindo ele mesmo) é igual ao próprio número. Veja o exemplo abaixo:

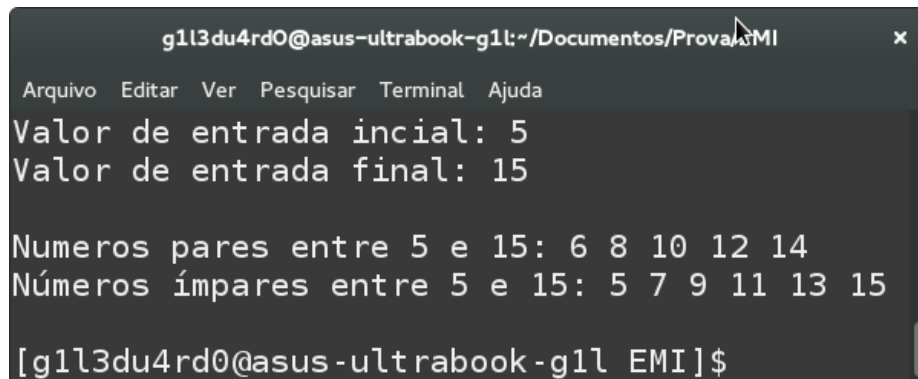
```
g1l3du4rd0@asus-ultrabook-g1l:~/Documentos/Prova/EMI x
Arquivo Editar Ver Pesquisar Terminal Ajuda
Digite um número: 6
Resposta: 1 + 2 + 3 = 6 logo [PERFEITO]

[g1l3du4rd0@asus-ultrabook-g1l EMI]$ ./a.out
Digite um número: 12
Resposta: 1 + 2 + 3 + 4 + 6 = 16 logo [IMPERFEITO]

[g1l3du4rd0@asus-ultrabook-g1l EMI]$
```

Obs.: lembre-se, a saída do seu programa deve ser idêntica a apresentada no exemplo da questão.

5. **(Difícil)** Codifique um programa em C que solicite ao usuário dois valores inteiros. Após isso apresente todos os números pares e ímpares existentes entre esses dois valores (incluindo-os). Utilize, obrigatoriamente, **apenas um laço de repetição** para resolver o exercício. Veja o exemplo:



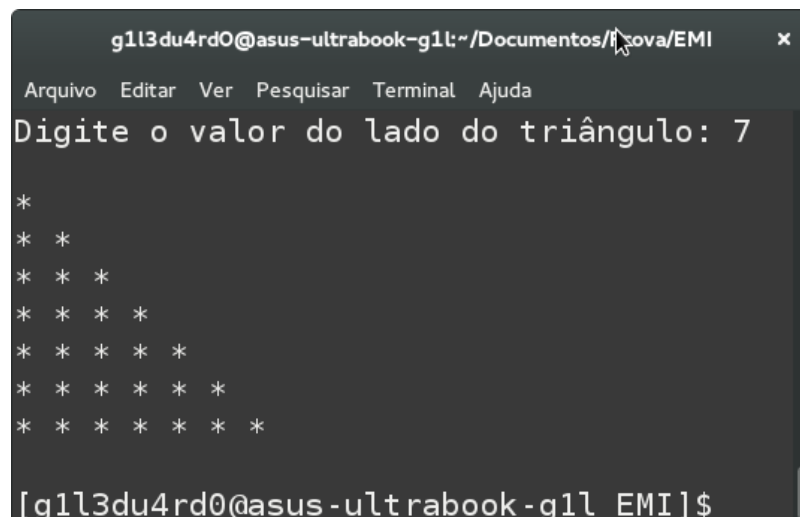
```
g1l3du4rd0@asus-ultrabook-g1l:~/Documentos/Prova/EMI
Arquivo Editar Ver Pesquisar Terminal Ajuda
Valor de entrada inicial: 5
Valor de entrada final: 15

Numeros pares entre 5 e 15: 6 8 10 12 14
Números ímpares entre 5 e 15: 5 7 9 11 13 15

[g1l3du4rd0@asus-ultrabook-g1l EMI]$
```

Obs.: lembre-se, a saída do seu programa deve ser idêntica a apresentada no exemplo da questão.

6. **(Difícil)** Codifique um programa em C que solicite ao usuário um valor inteiro referente ao tamanho do lado de um triângulo equilátero (possui todos os lados iguais). Através desse valor desenhe o triângulo usando o caractere asterisco “\*”, onde o tamanho do lado é representado pelo número de asteriscos que ele contém. Veja o exemplo:



```
g1l3du4rd0@asus-ultrabook-g1l:~/Documentos/Prova/EMI
Arquivo Editar Ver Pesquisar Terminal Ajuda
Digite o valor do lado do triângulo: 7

*
* *
* * *
* * * *
* * * * *
* * * * *
* * * * *

[g1l3du4rd0@asus-ultrabook-g1l EMI]$
```