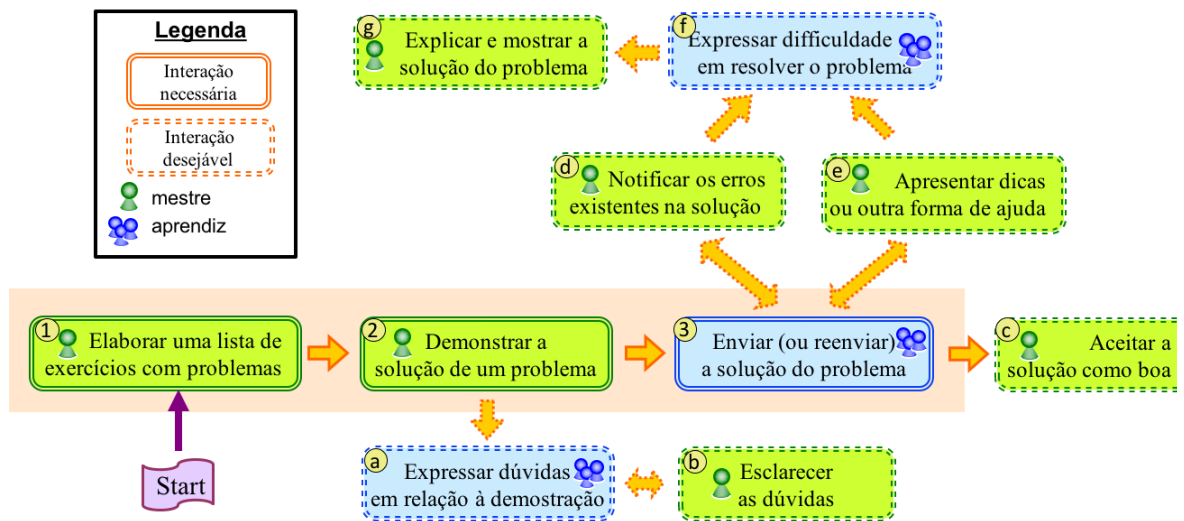


# Atividade Colaborativa 01



A figura de acima mostra o fluxo de passos necessários e desejáveis que serão efetuados nesta atividade colaborativa.

No primeiro passo, o estudante com papel de **mestre** deve **elaborar uma lista de exercícios com problemas** de programação. Os problemas da lista deve estar relacionados ao tópico "estruturas condicionais (if-else)" e deve consistir de três (03) problemas: um problema de nível fácil, um problema de nível médio, e um problema de nível difícil. Para facilitar sua tarefa, você pode selecionar os problemas da seguinte lista:

- Lista de Exercícios - IF.pdf (<https://www.dropbox.com/s/zrps4f14b0sp2i3/Lista%20de%20Exerc%C3%ADcios%20IF.pdf?dl=0>)

Quando a lista estiver pronta, ela deve ser enviada pelo estudante com papel de **mestre** mediante uma mensagem no fórum fazendo click no link "**Elaborar uma lista de exercícios com problemas (1)**".

No segundo passo, o estudante com papel de **mestre** deverá **demonstrar a solução de um problema** da lista de exercícios. Para isso, o estudante com papel de **mestre** enviará uma mensagem no fórum fazendo click no link "**Demonstrar a solução de um problema (2)**" apresentando os detalhes de como ele chegou à solução. O uso de comentários no código são muito úteis para completar esta tarefa.

- Depois da demonstração de como resolver o problema, o estudante com papel de **aprendiz** poderá expressar suas dúvidas fazendo click no link "**Expressar dúvidas em relação à demonstração (a)**".
- Logo, o estudante com papel de **mestre** poderá esclarecer as dúvidas do aprendiz fazendo click no link "**Esclarecer as dúvidas (b)**".

No terceiro passo, o estudante com papel **aprendiz** deverá **enviar a solução para a lista** de exercícios. Para cada problema da lista, ele deverá enviar uma mensagem no fórum fazendo click no link "**Enviar (ou reenviar) a solução de um problema (3)**". O estudante com papel de

aprendiz deve enviar esta mensagem na maior brevidade possível para obter uma revisão rápida do mestre. Não é necessário enviar a solução final, é suficiente enviar um esboço da solução ou uma mensagem solicitando ajuda para resolver o problema.

- Depois de enviar a mensagem com a solução de um problema, esboço ou ideia inicial, o estudante com papel **mestre** poderá:
  - "**Aceitar a solução como boa (c)**" se não encontrar erros na solução apresentada pelo aprendiz,
  - "**Notificar os erros existentes na solução (d)**" se encontrar erros na solução apresentada pelo aprendiz, ou
  - "**Apresentar dicas ou outra forma de ajuda (e)**" para a resolução do problema se o aprendiz enviar um esboço ou uma mensagem solicitando ajuda.
- Após a **notificação dos erros existentes na solução apresentada pelo aprendiz** ou **apresentação dicas ou outra forma de ajuda**, o estudante com papel de **aprendiz** poderá:
  - **Reenviar a solução do problema** para ser revisado pelo estudante com papel de mestre, fazendo click no link "**Enviar (ou reenviar) a solução de um problema (3)**" ou
  - "**Expressar dificuldade em resolver o problema (f)**", se ele não conseguir resolver o problema depois de conhecer os erros cometidos ou receber dicas para resolver o problema
- Finalmente, se o estudante com papel de aprendiz **expressar sua dificuldade em não conseguir resolver o problema**, o estudante com papel de **mestre** poderá "**Explicar e mostrar a solução do problema (g)**" que não consegue ser resolvido pelo aprendiz.

---

Permalink | Edit

---

 Moodle Docs for this page

You are logged in as Admin User (Log out)  
SSC0600 - Introdução à Ciência de Computação I

## SSC0600 - Introdução à Ciência de Computação I

### Lista de Exercícios (SE-ENTÃO)

1) Ler um número fornecido pelo usuário, e se este número for positivo, calcular a raiz quadrada do número. Se o número for negativo, exibir uma mensagem dizendo que o número é inválido. **(Nível: Fácil)**

2) Escreva um algoritmo que verifique se um valor inteiro é par ou ímpar. Dica: Utilize o operador de RESTO % (e.g.  $4\%2 \Rightarrow 0$ ;  $5\%2 \Rightarrow 1$ ). **(Nível: Fácil)**

3) Faça um programa que leia o sexo e a altura de uma pessoa e calcule o seu peso ideal, utilizando as seguintes fórmulas: (Nível: Fácil)

- para homem:  $(72.7 \times h) - 58$
- para mulher:  $(62.1 \times h) - 44.7$

4) Faça um algoritmo que leia o valor da temperatura, e depois da unidade de medida que está esta temperatura ("c" ou "f"). Se a unidade da temperatura fornecida for em graus Celsius (°C), converta para graus Fahrenheit (°F), exibindo o resultado na tela. Se a unidade da temperatura fornecida for em graus Fahrenheit (°F), converta para graus Celsius (°C), exibindo o resultado na tela. **(Nível: Fácil)**

Nota:  $^{\circ}\text{F} = ^{\circ}\text{C} \times 1,8 + 32$

$^{\circ}\text{C} = (^{\circ}\text{F} - 32) / 1,8$

5) Uma empresa decide dar um aumento aos seus funcionários de acordo com uma tabela que considera o salário atual e o tempo de serviço de cada funcionário. Os funcionários com menor salário terão um aumento proporcionalmente maior do que os funcionários com um salário maior, e conforme o tempo de serviço na empresa, cada funcionário irá receber um bônus adicional de salário.

Faça um programa que leia: (1) o valor do salário atual do funcionário; (2) o tempo de serviço deste funcionário na empresa (nro. de anos de trabalho na empresa). Use as tabelas abaixo para calcular o salário reajustado deste funcionário e imprima o valor do salário final reajustado, ou uma mensagem caso o funcionário não tenha direito a nenhum aumento. **(Nível: Difícil)**

Sal atual	Reajuste	Tempo	Bonus
Até 500,00	25,00%	menos 1 ano	Nada
Até 1000,00	20,00%	1 a 3 anos	100
Até 1500,00	15,00%	4 a 6 anos	200

Até 2000,00	10,00%	7 a 10 anos	300
Mais de 2000,00	Nada	Mais de 10 anos	500

6) Ler três números inteiros e exibir o maior deles. Se forem iguais, exibir qualquer um dos três (não utilizar o comando PARA/for e nem vetores para armazenar variáveis). **(Nível: Médio)**

7) Ler três números inteiros e exibir na tela valores dispostos em ordem crescente. (não utilizar o comando PARA/for e nem vetores para armazenar variáveis) **(Nível: Difícil)**

~~8) Escreva um algoritmo que verifica se um número fornecido é primo.~~

9) Dado três valores, A, B e C, construa um algoritmo para verificar se estes valores podem ser valores dos lados de um triângulo. Em caso positivo verificar se o triângulo é escaleno, equilátero ou isósceles. **(Nível: Médio)**

10) Crie um programa que consegue adivinhar um número no raio de 1 a 15 em no máximo 4 tentativas (assuma que o jogador diga se o número é maior ou menor a cada tentativa incorreta). **(Nível: Médio)**

11) Escrever um algoritmo que lê a hora de início e hora de término de um jogo, ambas subdivididas em dois valores distintos : horas e minutos. Calcular e escrever a duração do jogo, também em horas e minutos, considerando que o tempo máximo de duração de um jogo é de 24 horas e que o jogo pode iniciar em um dia e terminar no dia seguinte. **(Nível: Médio)**

12) A jornada de trabalho semanal de um funcionário é de 40 horas. O funcionário que trabalhar mais de 40 horas receberá hora extra, cujo cálculo é o valor da hora regular com um acréscimo de 50%. Escreva um algoritmo que leia o número de horas trabalhadas em um mês e o salário por hora. Com estes dados o algoritmo deve calcular o salário total do funcionário, que deverá ser acrescido das horas extras, caso tenham sido trabalhadas e imprimir o resultado (considere que o mês possua 4 semanas exatas). **(Nível: Fácil)**