	<b>UNIVERSIDADE ESTADUAL DA PARAÍBA</b>	
	Curso:	<b>Ciência da Computação</b>
	Disciplina:	<b>Laboratório de Programação 1</b>
	Professor:	<b>Danilo Abreu Santos</b>
	<b>Laboratório 01</b>	Data 04/09/2019

**Critérios a serem avaliados:** *entendimento e implementação da questão; compreensão dos comandos de entrada e saída; domínio das estruturas condicionais; domínio das estruturas de repetições;*

### Como devo fazer o laboratório?

Você poderá utilizar qualquer editor de códigos para escrever o seu programa. Até mesmo um processador de textos simples (bloco de notas).

### O que deverei enviar?

Um único arquivo com a extensão c. Você deverá criar um arquivo seguindo as seguintes **REGRAS:** seuNomeLab1.c (Exemplo: daniloAbreuLab1.c). Os demais laboratórios irão seguir este mesmo padrão.

### Como deverei enviar esta atividade?

Pelo google sala de aula. Não envie nada por e-mail.

### Os prazos?

Estarão na descrição da tarefa no google sala de aula. Após o horário de entrega: penalidade de 50% da nota. Acima de 12 horas do prazo de entrega: penalidade de 75% da nota. A partir de 18 horas após o prazo de entrega a atividade será zerada.

### Como será avaliado o meu programa?

Primeiramente, se o programa funciona adequadamente. Depois, se foi utilizado as técnicas de forma eficaz e eficiente (Gambiarras não são bem-vindas). Utilize nomes de variáveis que deixem claro a sua finalidade (ex.: numero1, soma). Insira comentários em seu código. Faça um código indentado.

Evite constrangimentos, não compartilhe/copie códigos de outros colegas. Código iguais: (nota/quantidade de trabalhos iguais) - 1.5.

Não adianta compartilhar o código e mudar as variáveis. Faça o seu! 😊

### Não entendi bem o laboratório, o que devo fazer?

Primeiro, procure os monitores da disciplina. Se não for suficiente, entre em contato com o professor. Não deixe para tirar dúvidas na reta final do laboratório. Quanto mais cedo forem sanadas, mais refinado poderá ficar o seu código.

## O LABORATÓRIO

Usando apenas as técnicas apresentadas até então na nossa disciplina, escreva um programa que mostre e calcule as seguintes opções:

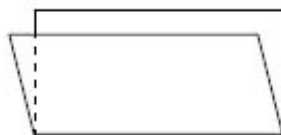
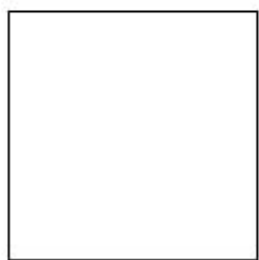
1. Sequência de Fibonacci
  2. Sequência de termos
  3. Operações aritmética
  4. Dobras
  5. Sair do programa
1. Leia um número positivo do usuário, então, calcule e imprima a sequência Fibonacci até o primeiro número superior ao número lido. Exemplo: se o usuário informou o número 30, a sequência a ser impressa será 0 1 1 2 3 5 8 13 21 34.
  2. Sequência de termos: Faça um programa que peça um número 'n' ao usuário, e que gere um novo n de acordo com a seguinte regra:  
$$\text{se } n \text{ é par, } n = n / 2$$
$$\text{se } n \text{ é ímpar, } n = 3 * n + 1$$
$$\text{imprime } n$$

O programa deve parar quando n tiver o valor igual a 1.

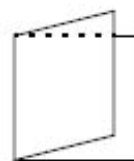
Por exemplo, para  $n = 13$ , a saída será:

$$40 \rightarrow 20 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$$
  3. Faça um programa que calcule as 4 operações aritmética entre dois números digitados pelo usuário. O usuário deverá digitar, além dos dois números, a operação que será calculada. Exemplo:  $2 + 5$ . O programa entende que o + significa adição.
  4. Suponha uma folha de papel quadrada. Imagine que, nesta folha de papel será feita a operação “dobradura” D, que consiste em dobrar a folha 2 vezes, conforme vemos na imagem 1. Essa operação poderá ser feita N vezes (desconsidere que o papel, a cada dobradura, se torna mais duro). Após as operações, o papel será cortado em verticalmente e horizontalmente, conforme a imagem 2. O seu programa deverá contar com um valor de entrada N que consistirá na quantidade de vezes que a operação será realizada. Se for digitado um valor negativo, finaliza-se a operação. Após ser digitado o valor de N, o seu programa deverá apresentar a quantidade de **pedaços** que resultarão a partir das N operações. Exemplo:  $N=1 \Rightarrow 9$  dobraduras;  $N=0 \Rightarrow 4$  dobraduras.

Imagem 1

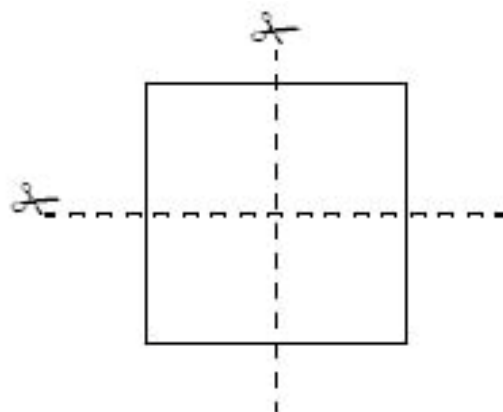


Primeira dobra



Segunda dobra

Imagem 2



5. Finalizará o programa.