



ECOP13-Lab1

Guia de Laboratório
Prof. André Bernardi
andrebernardi@unifei.edu.br



ECOP13 - Plano de curso (P1, P2, P3)

Início	Fim	Descrição
26/08/2022	26/08/2022	Apresentação da disciplina - Lab 01
02/09/2022	02/09/2022	Lab 01 - Revisão
09/09/2022	09/09/2022	Lab 02 - Classes I
16/09/2022	16/09/2022	Lab 03 - Classes II
23/09/2022	23/09/2022	Lab 04 - Sobrecarga de Operadores I
30/09/2022	30/09/2022	Lab 05 - Sobrecarga de Operadores II
07/10/2022	07/10/2022	Lab 06 - Herança
14/10/2022	14/10/2022	Lab 07 - Polimorfismo
21/10/2022	21/10/2022	Lab Avaliativo I
04/11/2022	04/11/2022	Lab 08 - Processamento de Exceções
11/11/2022	11/11/2022	Lab 09 - Templates
18/11/2022	18/11/2022	Lab 10 - STL I
25/11/2022	25/11/2022	Lab 11 - STL II
02/12/2022	02/12/2022	Lab 12 - MFC I
09/12/2022	09/12/2022	Lab 13 - MFC II
16/12/2022	16/12/2022	Avaliação Prática



Avaliações

- Primeiro Bimestre:
 - Labs de 01 a 07 – 26/08 a 14/10 – 60%
 - Lab Avaliativo I – 21 de Outubro – 40%
- Segundo Bimestre:
 - Labs de 08 a 13 – 04/11 a 09/12 – 60%
 - Lab Avaliativo II – 16 de Dezembro – 40%
- Nota final: Média aritmética



Referências

- DEITEL, Harvey M; DEITEL, Paul J. **C++:** how to program. 7 ed. Nova Jersey: Pearson Prentice Hall, 2010. xxxv, 1068. ISBN: 9780136152507, 0136117260.
- DEITEL, Harvey M; DEITEL, Paul J. **C++:** como programar. 5 ed. reimpr. São Paulo: Pearson Prentice Hall, 2008. 1163. ISBN: 9788576050568.



1º Laboratório ECOP13A

26 Ago e 02 Set de 2022



Exercícios

1ª: Faça um aplicativo em C++ para calcular o fatorial de um número, digitado pelo usuário. Lembrar que:

$$n! = 1 * 2 * \dots * (n-2) * (n-1) * n;$$

Pode ser resolvido diretamente na **main**, usando função iterativa ou função recursiva.

1ª questão – Exemplo de Solução

```
1 // Exercício 1
2 //
3 #include <iostream>
4 using namespace std;
5
6 long fat(long n){
7     return (n < 2) ? 1 : n*fat(n - 1);
8 }
9
10 long fat2(long n){
11     long result = 1;
12     for (int i = 2 ; i <= n; i++)
13         result = result * i;
14     return result;
15 }
16
17 int main ( void )
18 {
19     long n = 30;
20     cout << "Entre com um numero para o calculo do fatorial:";
21     cin >> n;
22     cout << "O fatorial de " << n << " é " << fat(n);
23 }
```



Exercícios

2ª: Crie um programa em C++ para calcular e imprimir os 20 primeiros números primos. Lembrar que um número primo é aquele que só é divisível por ele mesmo e por um.

2ª questão – Exemplo de Solução

```
1 // Exercício 2
2 #include <iostream>
3 using namespace std;
4
5 bool verificaPrimo(int x){
6     if (x == 2) return true;
7     if (x % 2 == 0) return false;
8
9     for (int i = 3; i <= sqrt(x)+1; i++){
10         if (x % i == 0) return false;
11     }
12     return true;
13 }
14
15 int main( void ){
16     cout << "INICIO" << endl;
17     int cnt = 0;
18     int i = 2;
19     while (cnt < 20 ){
20         if (verificaPrimo(i) == true){
21             cout << i << " ";
22             cnt++;
23         } i++;
24     }
25     cout << endl << "FIM" << endl;
26 }
```



Exercícios

3^a: Construir um programa em C++ que sirva de menu de opções (formato texto) para selecionar entre os programas deste laboratório. Utilize a estrutura switch/case.



3ª questão

Exemplo de Solução

```
1 // Exercício 3
2 #include <iostream>
3 using namespace std;
4
5 int menu(){
6     cout << "0 - sair" << endl;
7     cout << "1 - fat" << endl;
8     cout << "2 - 20 primeiros primos" << endl;
9     cout << "4 - mudança de base" << endl;
10    cout << "5 - retangulo" << endl;
11    cout << "6 - soma ate n" << endl;
12    cout << "7 - mdc" << endl;
13    cout << "8 - binário" << endl;
14    cout << "9 - palindromo" << endl;
15    cout << "Escolha:" << endl;
16    cin >> op;
17    return op;
18 }
```



3ª questão

Exemplo de Solução

```
19 int main( void ){
20     int op = menu();
21     while (op != 0){
22         switch (op){
23             case 1: cout << "Opção 1: " << endl; break;
24             case 2: cout << "Opção 2: " << endl; break;
25             case 3: cout << "Opção 3: " << endl; break;
26             case 4: cout << "Opção 4: " << endl; break;
27             case 5: cout << "Opção 5: " << endl; break;
28             case 6: cout << "Opção 6: " << endl; break;
29             case 7: cout << "Opção 7: " << endl; break;
30             case 8: cout << "Opção 8: " << endl; break;
31             case 9: break;
32         }
33         op = menu();
34     }
35     return 0;
36 }
```



Exercícios

4ª: Analisar o programa a seguir e verificar qual o resultado para a chamada do método correspondente para $n=100$ e $base = 16$.

Obs: A análise consiste em executar o programa a seguir como se você fosse o computador.

4ª questão – Exemplo de solução

```
1 // Exercício 4
2 // Teste de Recursão.
3 #include <iostream>
4 using namespace std;
5
6 int main ( void )
7 {
8     int n = 100;
9     int base = 16;
10    printInt(n, base);
11 }
12
13 // função recursiva - critério de saída n < base
14 void printInt( int n, int base )
15 {
16     char array[16] = {'0', '1', '2', '3', '4', '5', '6', '7',
17                      '8', '9', 'a', 'b', 'c', 'd', 'e', 'f'};
18     if(n >= base)
19         printInt(n/base, base);
20     cout << array[ n % base ] ;
21 }
22 // fim do arquivo
```



Exercícios

5ª Escreva um programa em C++ que desenhe um Retângulo Na Tela, com as seguintes personalizações possíveis:

- Possua **largura** e **altura** lidas pelo teclado;
- Leia o caractere que vai ser usado para desenhar a **borda**;
- Leia o caractere de **preenchimento**;
- Pergunte ao usuário se o retângulo vai ser **preenchido** ou não.

Como exemplo, se o programa executasse com largura = 12, altura = 5, com o caractere **x** para borda, o caractere **.** para preenchimento e o retângulo fosse preenchido, o resultado seria o mostrado abaixo:

```
xxxxxxxxxxxxxxxx
x.....x
x.....x
x.....x
x.....x
xxxxxxxxxxxxxxxx
```



5ª questão

Exemplo de Solução

```
1 // Exercício 5
2 #include <iostream>
3 using namespace std;
4
5 void imprime(int l, int a, char b, char p){
6     for (int i = 0; i < a; i++){
7         for (int j = 0; j < l; j++){
8             if (i == 0 || j == 0 || i == a - 1 || j == l - 1)
9                 cout << b;
10            else cout << p;
11        }
12        cout << endl;
13    }
14}
```




5ª questão

Exemplo de Solução

```
19 int main( void ){
20     int l, h;
21     char borda, preenchimento;
22
23     cout << "digite largura e altura" << endl;
24     cin >> l >> h;
25
26     cout << "Digite char da borda" << endl;
27     cin >> borda;
28
29     cout << "Digite preenchimento" << endl;
30     cin >> preenchimento;
31
32     imprime(l, h, borda, preenchimento);
33
34     return 0;
35 }
```



Exercícios

6ª: Escreva uma função recursiva para calcular a soma dos n primeiros números inteiros.

7ª: O máximo divisor comum (**mdc**) é definido como o maior número inteiro pelo qual dois números são divididos exatamente. Criar uma função para encontrar **mdc** dados dois inteiros. Sabendo que

$$\begin{aligned}\text{mdc}(x, y) &= x && , \text{ para } y = 0; \\ \text{mdc}(x, y) &= \text{mdc}(y, x \% y) && , \text{ caso contrario}\end{aligned}$$

6ª e 7ª questão

Exemplo de Solução

```
1 // Exercício 5
2 #include <iostream>
3 using namespace std;
4
5 int gcd(int a, int b){
6     return b ? gcd(b, a%b) : abs(a);
7 }
8
9 int soma(int x){
10     return (x == 1) ? 1 : x + soma(x - 1);
11 }
12
13 int main( void ){
14
15     cout << « Máximo divisor comum." << endl;
16     cout << "Digite a e b" << endl;
17     cin >> a >> b;
18     cout << gcd(a, b) << endl;
19
20     cout << « Soma ate n" << endl;
21     cout << "digite n" << endl;
22     cin >> n;
23     cout << soma(n) << endl;
24
25     return 0;
26 }
```



Exercícios

8. Escreva um aplicativo que insira um inteiro contendo apenas 0s e 1s (isto é, um inteiro binário) e imprima seu equivalente decimal.

Instruções: use os operadores de resto (%) e de divisão (/) para selecionar os dígitos do número binário, um de cada vez, da direita para a esquerda. No sistema numérico decimal, o dígito mais à direita tem um valor posicional de 1 e o próximo dígito à esquerda um valor posicional de 10, depois 100, depois 1000, e assim por diante. O número decimal 234 pode ser interpretado como $4 * 1 + 3 * 10 + 2 * 100$. No sistema numérico binário, o dígito mais à direita tem um valor posicional de 1, o próximo dígito à esquerda um valor posicional de 2, então 4, depois 8 e assim por diante. O equivalente decimal do binário 1101 é $1 * 1 + 0 * 2 + 1 * 4 + 1 * 8$, ou $1 + 0 + 4 + 8$ ou, **13**.

8ª questão

Exemplo de Solução

```
1 // Exercício 8
2 #include <iostream>
3 #include <cmath>
4 using namespace std;
5
6 int main( void ){
7
8     int bi;
9     cout << "Insira um valor em binário: " << endl;
10    cin >> bi;
11    int sobra, novoValor = 0;
12
13    for (int j = 0; j < 6; j++) {
14        sobra = bi%10; //bi - ((bi / 10)*10);
15        bi /= 10;
16        novoValor += sobra * pow(2, j);
17    }
18    cout << "Valor em Decimal: " << novoValor << endl;
19    return 0;
20 }
```



Exercícios

9. Um palíndromo é uma sequência de caracteres na qual é possível ler-se o mesmo conteúdo, tanto do início até o final como do final até o início. Por exemplo, cada um dos quatro números inteiros seguintes é um palíndromo: 1234321, 5555555, 4556554 e 1126211.

Escreva um aplicativo que leia um **inteiro** de sete dígitos e determine se é um palíndromo. Se o número não tiver sete dígitos, exiba uma mensagem de erro e permita que o usuário insira outro valor.



Exercícios

Questão extra: Defina uma classe que represente um retângulo, com os atributos comprimento e largura. Setar o valor padrão desses atributos para 1. Criar funções de acesso para cada um dos atributos, validando os valores como números entre 0 e 20. Definir construtores que permitam o recebimento do valor de um atributo como parâmetro. Criar métodos para o cálculo da *área* e *perímetro* do retângulo. Criar uma função capaz de imprimir esse retângulo conforme descrito na questão 5. Criar um método para verificar se o retângulo é um quadrado.



Compiladores

- Codeblocks
- Dev
- Visual Studio Code
- etc.