

**Programas , Trabalhos da faculdade ,Idéias .**

**Sorocaba  
2020**

## **Biografia**

### **Kaicon Ricardo Paes Marcino de Almeida**

Natural da cidade de Apiaí , situada no estado de São paulo . Atualmente é formando em Gestão de Tecnologia da Informação . Possui alguma experiência na área de Automação Industrial onde já atuou como operador de estação automatizada para tratamento e descarte de resíduos de tornearia . Possui cursos técnicos na linguagem de Programação C Sharp porém não trabalhou no desenvolvimento de um software de forma profissional sendo um amador em áreas como eletrônica , robótica , e mesmo na Música onde toca Violino , teclado .

**Sorocaba  
2020**

## **Resumo**

Neste livro eu venho juntando as anotações , ideias ,programas em c sharp que eu venho escrevendo , bem como técnicas de programação que funcionam . E insertos de sobras de anotações dos outros livros .

**Sorocaba  
2020**

## Sumário

1. CLASSE PESSOA	22
2. CONTROLE DE ACADEMIA	24
3. programa montador v1	30
4. prog mont wind appl	37
5. prog montador 09 11 2019	40

## Insero do livro : **Novela do pim página 10** **Planejamento e implantação da Segurança de Informação**

As definições do planejamento de segurança da informação são importante e cruciais em qualquer setor empresarial porem esta etapa de planejamento deve estar seguindo os processos da empresa em questão . Existem variados niveis dentro da Segurança da Informação . No **Nível físico** deve existir a diretriz de se impedir acesso de não - funcionários a areas restritas pois os mesmos podem ter acesso a equipamentos importantes ao funcionamento da empresa como por exemplo **Servidores** de onde vem a necessidade de se identificar funcionários exatamente ou seja onde atuam e quando atuam e mesmo ações simples como instalar câmeras de segurança em areas críticas

Dentro do **Nível Lógico** existe a possibilidade de se instalar programas que tem a função de monitorar redes e quem acessa os computadores em tempo integral.

Já no **Nível Humano** pode se projetar treinamento sobre a politica de segurança para que nao ocorra quebra de protocolo, exemplo acesso indevido a areas, software etc

Existem ainda procedimentos de planejamento como o desenho, projeção e implantação de plano PDCA contínuo de análise de fatores de riscos , design do treinamento para uso do software e da Política de Segurança de Informação

### FLUXOGRAMA DO PROCESSO

Infraestrutura tem o modelo estipular recursos necessarios atingir metas estipuladas

o fluxograma a seguir mostra os processos da infraestrutura do projeto

Análise de possíveis falhas

o fluxograma a seguir mostra os processos da análise de riscos e falhas

Conclusão e possíveis resultados

LINK MONITOR DE REDES

<https://www.br.paessler.com/prtg> "O PRTG MONITORA CADA ASPECTO DE SUA

INFRAESTRUTURA DE TI"

instalar programa supervisor banco de dados que seleciona

informacoes LINK MONITOR BANCO DE DADOS

<https://www.quest.com/br-pt/products/foglight-for-sql-server/> "Foglight Performance

Investigator for SQL Server" criticas, alteracao de dados nao comum etc

**programas escritos em c sharp 04 03 2020**

Programas escritos em C sharp

04 / 03 / 2020

```
using System;
namespace gewgwegg
{
public class Teste
{
public static string fichadesaida1 ;
public static string fichadesaida2 ;
}
class Program
{
public static void Main(string[] args)
{
Console.WriteLine("programa montador de pecas ");
Console.ReadKey(true);
Console.Clear();
inicio :
Console.WriteLine("alternativas de montagem : ");
Console.WriteLine(" 1 velocidade ");
Console.WriteLine(" 2 força ");
Console.WriteLine(" 3 imprima resultados ");
Console.WriteLine(" 4 fim do programa ");
Console.WriteLine(" 5 opção ");
Console.WriteLine(" 6 opção ");
Console.WriteLine(" 7 opção ");
```

```
Console.WriteLine(" 8 opção ");
Console.WriteLine(" 9 opção ");
Console.WriteLine(" 10 opção ");
Int16 pergunta ;
Console.WriteLine("Selecione uma opção : ");
pergunta = Int16.Parse(Console.ReadLine());
Console.ReadKey(true);
Console.Clear();
switch (pergunta )
{
case 1:
goto velocidade ;
case 2:
goto força ;
case 3 :
goto saidafinal ;
case 4 :
goto fimdoprograma ;
default:
Console.WriteLine("alternativa invalida ");
break;
}

Console.ReadKey(true);

velocidade :
```

```
Console.WriteLine("alternativas de montagem : ");
Console.WriteLine(" 1 motor ");
Console.WriteLine(" 2 motor ");
Console.WriteLine(" 3 x ");
Console.WriteLine(" 4 x ");
Console.WriteLine(" 5 x ");
Console.WriteLine(" 6 x ");
Console.WriteLine(" 7 x ");
Console.WriteLine(" 8 x ");
Console.WriteLine(" 9 x ");
Console.WriteLine(" 10 x ");
Int16 pergunta1 ;
string velocidade;
```

```
Console.WriteLine("Selecione uma opção : ");
pergunta1 = Int16.Parse(Console.ReadLine());
Console.ReadKey(true);
Console.Clear();
switch (pergunta1 )
{
case 1:
```

```
Console.WriteLine("voce escolheu motor 1 " );
velocidade = " motor1 ";
Teste.fichadesaida1 = velocidade;
```

```
Console.ReadKey(true);
```

```
Console.Clear();
```

```
break;
```

```
case 2:
```

```
Console.WriteLine("voce escolheu motor 2" );
```

```
velocidade = " motor1 ";
```

```
Teste.fichadesaida1 = velocidade;
```

```
Console.ReadKey(true);
```

```
Console.Clear();
```

```
break;
```

```
default:
```

```
Console.WriteLine("alternativa invalida ");
```

```
break;
```

```
}
```

```
Console.ReadKey(true);
```

```
força :
```

```
Console.WriteLine("alternativas de montagem : ");
Console.WriteLine(" 1 estrutura ");
Console.WriteLine(" 2 estrutura ");
Console.WriteLine(" 3 x ");
Console.WriteLine(" 4 x ");
Console.WriteLine(" 5 x ");
Console.WriteLine(" 6 x ");
Console.WriteLine(" 7 x ");
Console.WriteLine(" 8 x ");
Console.WriteLine(" 9 x ");
Console.WriteLine(" 10 x ");
Int16 pergunta2 ;
string força;
Console.WriteLine("Selecione uma opção : ");
pergunta2 = Int16.Parse(Console.ReadLine());
Console.ReadKey(true);
Console.Clear();
switch (pergunta2 )
{
case 1:

Console.WriteLine("voce escolheu estrutura 1 " );
força = " estrutura 1 ";
Teste.fichadesaida2 = força;

Console.ReadKey(true);
Console.Clear();
```

```
break;
```

```
case 2:
```

```
Console.WriteLine("voce escolheu estrutura 2" );
```

```
força = " estrutura 2 ";
```

```
Teste.fichadesaida2 = força;
```

```
Console.ReadKey(true);
```

```
Console.Clear();
```

```
break;
```

```
default:
```

```
Console.WriteLine("alternativa invalida ");
```

```
break;
```

```
}
```

```
Console.ReadKey(true);
```

```
goto inicio;
```

saidafinal :

```
Console.WriteLine( Teste.fichadesaida1 + "E" + Teste.fichadesaida2 );
```

```
Console.ReadKey(true);
```

```
Console.Clear();
```

goto inicio;

findoprograma :

```
Console.WriteLine(" FIM DO PROGRAMA");
```

```
Console.ReadKey(true);
```

```
Console.Clear();
```

novo progrma EFWFESDGVVDVG

```
/*
```

```
* Created by SharpDevelop.
```

```
* User: Kaicon
```

```
* Date: 12/11/2019
```

```
* Time: 11:07
```

```
*
```

```
* To change this template use Tools | Options | Coding | Edit Standard Headers.
```

```
*/
```

```
using System;
```

```
namespace gewgwegg
{
public class Teste
{
public static string fichadesaida1 ;
public static string fichadesaida2 ;
public static string fichadesaida3 ;
public static Int16 somavelocidade ;
public static Int16 somapesagem ;
public static Int16 somaprocessamento;
public static Int16 somavalortotal;
public static Int16 somavalortotal1;
public static Int16 somavalortotal2;
public static Int16 estoque;
public static Int16 estoque1;
public static Int16 estoque2;
public static Int16 precounitario1;
}
class Program
{
public static void Main(string[] args)
{
Console.WriteLine("===== ");
Console.WriteLine("===== programa montador de CPU ===== ");
Console.WriteLine("===== ");
Console.ReadKey(true);
Console.Clear();
Console.WriteLine("===== ");
```

```
Console.WriteLine("==== login de usuario ==== ");
Console.WriteLine("===== ");
Console.ReadKey(true);
Console.Clear();
inicio :
Console.WriteLine("alternativas de montagem : ");
Console.WriteLine(" 1 gabinete ");
Console.WriteLine(" 2 placa mae ");
Console.WriteLine(" 3 Processador ");
Console.WriteLine(" 4 imprima resultados ");
Console.WriteLine(" 5 fim do programa ");
Console.WriteLine(" 6 valores do sistema ");
Console.WriteLine(" 7 controle de estoque ");
Console.WriteLine(" 8 fluxo de caixa ");
```

```
Int16 pergunta ;
Console.WriteLine("Selecione uma opção : ");
pergunta = Int16.Parse(Console.ReadLine());
Console.ReadKey(true);
Console.Clear();
switch (pergunta )
{
case 1:
goto gabinete ;
```

case 2:

goto placamae ;

case 3 :

goto centraleletronica;

case 4 :

goto saidafinal ;

case 5 :

goto fimdoprograma ;

case 6 :

goto somadosresultados;

case 7 :

goto unidadesrequisitadas;

case 8 :

```
goto fluxodecaixa ;
```

```
default:
```

```
Console.WriteLine("alternativa invalida ");
```

```
break;
```

```
}
```

```
Console.ReadKey(true);
```

```
gabinete :
```

```
Console.WriteLine("alternativas de montagem : ");
```

```
Console.WriteLine(" 1 gabinete aerocool cylon rgb R$ 221,00 ");
```

```
Console.WriteLine(" 2 gabinete warrior R$ 188,00 ");
```

```
Console.WriteLine(" 3 xxxxxxxxxxxxxxxx R$ 15,00 ");
```

```
Console.WriteLine(" 4 xxxxxxxxxxxxxxxx R$ 20,00 ");
```

```
Console.WriteLine(" 5 xxxxxxxxxxxxxxxx R$ 25,00 ");
```

```
Console.WriteLine(" 6 xxxxxxxxxxxxxxxx R$ 30,00 ");
```

```
Console.WriteLine(" 7 xxxxxxxxxxxxxxxx R$ 35,00 ");
```

```
Console.WriteLine(" 8 xxxxxxxxxxxxxxxx R$ 40,00 ");
```

```
Console.WriteLine(" 9 xxxxxxxxxxxxxxxx R$ 45,00 ");
```

```
Console.WriteLine(" 10 xxxxxxxxxxxxxxxx R$ 50,00 ");
```

```
Int16 pergunta1 ;
```

```
Console.WriteLine("Selecione uma opção : ");
```

```
pergunta1 = Int16.Parse(Console.ReadLine());
```

```
Console.ReadKey(true);
```

```
Console.Clear();
```

```
switch (pergunta1 )
```

```
{
```

```
case 1:
```

```
Console.WriteLine("=====
= ");
```

```
Console.WriteLine("===== voce escolheu gabinete aerocool cylon rgb ===== ");
```

```
Console.WriteLine("=====
= ");
```

```
string velocidade;
```

```
velocidade = " gabinete aerocool cylon rgb ";
```

```
Teste.fichadesaida1 = velocidade;
```

```
Int16 somavel;
```

```
somavel = 5 ;
```

```
Teste.somavelocidade = somavel ;
```