

## PROGRAMAÇÃO

### Arte ou Profissão?

Como você sabe sou desenvolvedor (*grosso modo* uma mistura de Analista de Sistema e Programador 2 em 1) e isso há muito tempo. A alguns dias um conhecido me perguntou o que era ser programador e como é ser programador no Brasil. Essa curiosidade fez-me pensar que seria oportuno contar o que é programação, como é ser um programador e como é o mercado de trabalho de programação no Brasil e quais suas perspectivas, caso você, leitor se interesse em trabalhar nessa área ou esteja também curioso sobre essa "misteriosa" e muito incompreendida profissão.

Não serei, porém, subjetivo por que há muitos profissionais nesta área que podem apontar diversas facetas ou terem experiências não aplicáveis em todos os casos. Deste modo vou procurar expor esse universo dos que constroem o mundo digital, Byte por Byte, de forma "generalizada".

#### **O que é PROGRAMAR?**

Quando ligamos o computador quase nem notamos, mas diversos programas começam a ser lidos e suas instruções são executadas pela máquina, caso contrário, não surgiriam letras no monitor, seu mouse não serviria de nada, você não ouviria sons pelas caixas acústicas do computador, nem a tela de inicialização poderia aparecer.

Quando o "computador" trava, normalmente alguma parte de um programa apresentou falha, entrando em ciclos infinitos ou utilizando partes de memória inválidas entre outros problemas, e pensamos que é a culpa é da máquina.

Quase todos os problemas que temos ao usar um computador são gerados por programas (erros ou falhas de programação, programas mau feitos, programas criados para fazerem estragos como vírus, worms, etc.), o Sistema Operacional também é um conjunto de programas responsável pela administração do hardware (máquina), outros softwares e a interação com o usuário.

#### **Como esses programas surgem?**

Ainda não existe programa que crie programas do nada, assim, todos os programas que existem foram imaginados, elaborados e escritos por pessoas de elevada qualificação técnica. Centenas de milhares de cérebros, mundo a fora, trabalharam para escrever seus jogos favoritos, o sistema operacional de seu computador, os programas que você utiliza em casa, no caixa do Banco, do supermercado e até nos sistemas digitais do seu carro ou forno microondas.

#### **Mas o que são esses programas?**

Um programa de computador é um conjunto de instruções que diz ao computador o que deve ser feito e quando, como se fosse um roteiro de filme ou uma receita culinária. O computador (máquina) sem esses programas é como um bloco de pedra! É incapaz de fazer qualquer coisa; nem mesmo funciona!

Para que o computador consiga funcionar é necessário um conjunto básico de instruções que diga como acessar a memória, o processador, o HD, como enviar e receber dados entre os diversos componentes, essas rotinas básicas são conhecidas por BIOS (Basic Input/Output System - Sistema básico de entrada e saída), sem esses programas essenciais seu computador é apenas um caro enfeite de mesa.

A BIOS normalmente é escrita em Assembler, uma antiga e muito poderosa linguagem de programação, as instruções compiladas em assembler são interpretadas pelo processador.

Perdeu-se? Se acalme. Vamos entender isso.

Todos os programas tem que ser escritos, ou seja, a idéia que está na cabeça do autor do programa tem que ser colocada em uma seqüência de instruções lógicas tipo: faça isso aqui, depois faça aquilo lá e assim por diante.

Porém o programador não escreve de qualquer jeito, senão como o computador vai "entender" o que ele quer?

Mas como o computador "entende" algo?

Esse "entendimento" primário é montado diretamente no microprocessador, ou seja, se, por exemplo, uma informação que está no disco deve ir para a memória RAM, no processador está "gravado" qual instrução deve ser enviada para que ele faça isso, assim, quando ele recebe o comando certo, com base na lista de instruções que pode executar, o executa conforme foi **construído** para fazer.

Essa linguagem embutida no processador é chamada de "código de máquina" e vem diretamente do "Assembler" que é a linguagem mais "crua" possível, ou seja, aquela mais próxima das instruções diretas da máquina. Por causa disso o Assembler é uma linguagem muito difícil de usar, é pouco intuitiva.

Pode-se programar escrevendo diretamente o código de máquina, usando um editor hexadecimal, mas isso é coisa que só é feita em casos extremos ou por entusiastas muito fanáticos em programação, o código de máquina, que é a única linguagem entendida pelo processador, é obtido das outras linguagens através de um programa chamado "Compilador".

O Compilador traduz um texto escrito em uma linguagem de programação, como o Assembler, no código de máquina, linguagem entendida pelo processador.

Há muitas linguagens de programação, qualquer programador com conhecimento técnico suficiente pode escrever uma linguagem própria. Uma linguagem de programação é resumidamente, um "universo" composto por tipos de variáveis (objetos que podem armazenar informações de diferentes características), comandos, objetos visuais (em caso de programação para ambientes gráficos como o Windows) tudo reunido dentro de normas e padrões conhecidos por "sintaxe". Uma vez estabelecido esse "universo" escreve-se o compilador que vai ser capaz de traduzir a linguagem em código de máquina.

Cada linguagem existente tem características próprias, vantagens e desvantagens. A que eu recomendaria para um principiante seria o Delphi, ainda que eu prefira o Visual Basic por ser muito mais fácil de usar, mas o Delphi apresenta similaridades interessantes com a mais importante de todas as linguagens de programação atuais, a linguagem "C", e, se quiser realmente trabalhar na área de programação, a "C", "C++" é imprescindível.

Então: um programa de computador é um conjunto de instruções, escritas de forma lógica em uma linguagem de programação.

O texto do programa é chamado de código-fonte.

Algumas linguagens não possuem compiladores (como o Visual Basic), eles possuem, ao invés, um "interpretador", que é um outro programa que se encarrega de entender as instruções e mandar ao processador ou ao sistema operacional os comandos apropriados. A vantagem da linguagem interpretada é que ela é menos sujeita a falhas não compensadas, porém um programa feito numa linguagem assim é mais lento para rodar que um que está compilado, já que não é executado "diretamente" no processador.

Sei que essa parte mais teórica é meio maçante para a maioria das pessoas, mas se quer ser um programador é importante saber como as coisas funcionam ;-)

### **Como é escrito um programa de computador?**

A maioria das linguagens de programação possuem um ambiente integrado de desenvolvimento, IDE (Integrated Development Environment), onde é possível escrever os códigos, depurar erros, simular funcionamento do programa, desenhar as interfaces, efetuar testes e ajustes. Mas também se pode escrever a maioria dos programas num simples editor de texto ASCII (texto sem formatação) como o Bloco de Notas do Windows.

### **Como é um programa de computador?**

Um programa de computador, como vimos, é um conjunto de instruções que informam ao processador o que ele deve fazer.

Se quiser podemos escrever um programinha exemplo, que irá "rodar" no seu navegador da internet. Por ser executado no navegador ele não precisará ser compilado (são um conjunto de instruções entendidas pelo navegador). Usaremos a linguagem "Java Script" para fazê-lo.

Vamos agora pensar como um programador:

"Eu quero um programa no qual eu possa escrever um texto pequeno e que, quando eu clicar num botão apareça uma caixa de mensagem com aquele texto e indicando a hora em que cliquei no botão"

Esse objetivo procurado é chamado de "Sistema" (um Analista de Sistemas é, em essência, o profissional encarregado de estudar as necessidades de um projeto, definindo que informações entram no programa, quais saem, de que forma o fazem e qual o processamento envolvido). Em nosso sistema temos o seguinte: um local (campo) para escrever o texto, a hora em que o botão for clicado - informações que entram. O processamento consiste em concatenar (juntar) as duas informações (texto e hora) de modo a exibi-las numa janelinha - informação que sai.

Mas um programador precisa de mais informações por que o computador funciona de forma exata. Um texto é algo muito vago. Texto de que tamanho?

Em nosso caso queremos uma frase pequena apenas, assim, 30 letras são suficientes. Definiremos as 30 letras como tamanho limite. E a hora? Só hora ou a data também? Hora em que formato? 24h? Mostraremos os segundos?

Viu?

Ao programar você precisa pensar em absolutamente tudo que possa estar envolvido ao programa, à informação e a maneira como o usuário interagirá.

Vamos definir que basta a hora no formato 12h, manhã e noite.

Para escrever esse programinha abrimos o editor de texto simples, no Windows o "Bloco de Notas", e, como vamos escrever algo que rodará no navegador da internet precisamos escrever alguma coisa básica para que possa ser rodado sem erros.

Copie o seguinte texto azul para um novo documento no bloco de notas (a cor do texto é só para indicar o que deve ser copiado, ok?):

```
<html>
<head>
<title>Programa TESTE</title>
</head>
<script language="JavaScript">

</script>
<body>
<br><br><div align="center">
<form name="f1">

</form>
</div></body></html>
```

O que escrevemos é apenas o arcabouço onde colocaremos nosso programinha. Toda a programação possui uma estrutura, como essa programação html, que permite compor a página de internet definindo blocos onde as informações pertinentes devem estar. A parte onde escreveremos o programa fica entre os textos: `<script language="JavaScript">` e `</script>` e a parte relacionada à interação com o usuário será escrita entre `<form name="f1">` e `</form>`

Vamos a parte de interação primeiro. Pode parecer confuso, já que ela vem depois na estrutura, mas o processo lógico e fazê-la primeiro por que precisaremos do nome do campo de texto para poder pegar seu conteúdo.

Entender o encadeamento, a seqüência de processos é fundamental na programação e é peça chave para programas rápidos e eficientes ou mostrengos cheios de erros. Esse encadeamento é chamado "algoritmo". Um Algoritmo é a forma pela qual a informação é trabalhada ou um processo é executado passo-a-passo.

Vamos pensar no algoritmo de nosso programa:

1º o usuário escreve um texto.

Dúvida: e se o usuário não escrever nada, devemos deixar que ele proceda ao passo seguinte?

Temos que pensar em todas as possibilidades!

Caso o usuário não escreva nada então nada acontecerá. Mas onde colocaremos essa verificação?

2º o usuário clica num botão para que a mensagem apareça.

Pensemos: o usuário só deve clicar nesse botão após escrever a mensagem, então uma coisa razoável é colocar a checagem para ser feita depois que o usuário clicar no botão por que a mensagem já deverá estar escrita, não é mesmo?

Vamos escrever a parte de interação com o usuário:

```
<b>Escreva sua frase: </b>
<input type="text" name="ctexto" size="20" maxlength="30" value="">
<br><br>
<input type="button" value="Mostrar" onClick="verificar()">
```

Nessa parte definimos o limite do texto (`maxlength`), que é 30 caracteres e colocamos a ação associada ao botão que, quando clicado irá chamar uma "função" denominada "verificar()".

Preste muita atenção aos caracteres maiúsculos e minúsculos por que na maioria das linguagens de programação são coisas bem diferentes, procure escrever exatamente como está aqui (as cores não são importantes, servem apenas para identificar o texto a ser escrito).

Agora temos que escrever a parte de programação propriamente dita.

Teremos duas funções, uma para verificar se há um texto escrito, outra para compor a mensagem. Por que duas funções?

Isso é para facilitar a programação. Dividir as tarefas em funções, cada qual fazendo apenas uma coisa ajuda a entender melhor o código e também a aprimorar as funções sem interferir no resto.

Um programador tem que ser metodoso, detalhista e, principalmente, saber documentar bem seu projeto e deixá-lo muito bem organizado, isso é fundamental quando é necessário fazer correções, melhorias e ampliações, bem como para posterior reutilização de trechos de um código escrito para um programa em outro. Um programador profissional sabe que adotar boas práticas de programação é o que diferencia o céu do inferno, e gera um enorme ganho de produtividade e qualidade.

A primeira função é a que verifica:

```
function verificar(){
  if (document.f1.ctexto.value.length > 0){
    processar();
  }
}
```

Esse jeito "esquisito" de escrever, com parênteses e chaves fazem parte das linguagens de programação, no caso, para esclarecer, os parênteses após o nome da função são o local onde se colocam parâmetros caso sejam passados, como não há parâmetro ou valor sendo passado, está vazio.

As chaves iniciam e terminam a função. O comando if (se), verifica o tamanho do valor contido no "ctexto", nome que demos ao campo de texto, que está no formulário cujo nome é "f1" que consta da página da internet que escrevemos (identificada como document). Se o tamanho do texto for maior que zero então ele vai para outra função chamada "processar". O ponto e vírgula no final é necessário para indicar que estamos "pulando linha". Algumas linguagens de programação derivadas da linguagem C usam esse "argumento" para informar que a linha é outra - não basta "teclar enter".

A segunda função:

```
function processar(){

  var texto = document.f1.ctexto.value;

  var agora = new Date();
  var horas = agora.getHours();
  var minutos = agora.getMinutes();

  var mensagem = texto + "\n" + ((horas >12) ? horas -12 :horas)
  mensagem += ((minutos < 10) ? ":0" : ":") + minutos
  mensagem += (horas >= 12) ? " P.M." : " A.M."

  Alert(mensagem);
}
```

Continuando nosso algoritmo: precisamos pegar a frase que a pessoa escreveu, pegar a hora atual, vamos colocar essa hora no padrão 12+12h e mesclar tudo para mostrar a mensagem.

Para ficar bem claro, nós vamos definir 5 variáveis (variável é um objeto cujo valor pode ser alterado, ou seja, pode variar). Na primeira colocaremos o texto que o usuário escreveu (texto). A segunda variável através de um comando específico pega a data completa do sistema (inclui dia, mês, ano, hora, minuto e segundo), mas só queremos a hora e o minuto, por isso a variável horas ficará só com o valor correspondente à hora e a minutos idem utilizando funções da linguagem javascript.

A variável "mensagem" mescla o "texto", pula a linha, se as horas forem maiores que 12, subtrairemos 12 dela (por exemplo, se forem 16h, -12 ficam 4 horas, por que usaremos o padrão de 12h no relógio e não 24h). Juntamos depois os minutos. Também consideramos que se os minutos forem inferiores a 10 colocaremos, para efeito estético, um zero antes, por exemplo, 9 minutos ficaria 09.

Para finalizar adicionaremos à mensagem A.M. ou P.M. (Anterior ao Meio-dia, Posterior ao Meio-dia). Se as horas forem maiores ou iguais a 12 poremos P.M., se menores, colocaremos A.M. e depois, com o comando Alert() mandamos exibir nossa "mensagem" completa para o usuário.

O nosso programa no total fica assim:

```
<html>
<head>
<title>Programa TESTE</title>
</head>
<script language="JavaScript">

function verificar(){
  if (document.f1.ctexto.value.length > 0){
    processar();
  }
}

function processar(){

  var texto = document.f1.ctexto.value;

  var agora = new Date();
  var horas = agora.getHours();
  var minutos = agora.getMinutes();

  var mensagem = texto + "\n" + ((horas >12) ? horas -12 :horas)
```

```

mensagem += ((minutos < 10) ? ":0" : ":") + minutos
mensagem += (horas >= 12) ? " P.M." : " A.M."

Alert(mensagem);
}
</script>
<body>
<br><br><div align="center">
<form name="f1">
<b>Escreva sua frase: </b>
<input type="text" name="ctexto" size="20" maxlength="30" value>
<br><br>
<input type="button" value="Mostrar" onClick="verificar()">
</form></div>
</body>
</html>

```

Complicado?

Bom, não é simples explicar os aspectos básicos da programação como linguagens, algoritmos, sistemas em poucas palavras, bem como mostrar como um programa é pensado, mas, acredito, você conseguiu, se me acompanhou até aqui, ter uma boa idéia de como "essa coisa" funciona.

Se quiser testar nosso "programinha", salve o texto como "teste.htm" (não deixe a extensão .txt caso apareça, no fim por que aí você não conseguirá abrir o documento como página de internet).

Abra o "Teste.htm" clicando duas vezes sobre ele e veja o programa em ação. Se você tiver feito tudo direitinho ele funcionará sem problemas.

Esse exemplo foi muito simplificado. Um programa de computador pode ter centenas de milhares de linhas de código, sendo que esse código deve estar disposto de forma lógica, exata, encadeada passo-a-passo senão o programa pode apresentar problemas, travar, gerar erros, enfim: "dá-pau"!

Por isso, se seu jogo de computador ou vídeo-game favorito apresenta uma falha ou se seu programa travou, lembre quão difícil, detalhista e complexo é programar e, ao invés de xingar a mãe de sei lá quem, ajude o desenvolvedor do programa/jogo enviando um email contando o que aconteceu e como.

Assim que um programa é finalizado normalmente os autores dispõem a versão "rascunho" ou "beta" para que algumas pessoas possam checar eventuais erros. Essa caça a erros imprevistos e falhas é chamada depuração e as pessoas que testam os programas "beta" são denominadas "beta-testers".

## O que é necessário para ser um programador?

Evidentemente ter conhecimento sobre uma linguagem de programação e os demais "utensílios", mas o que importa é que um programador é uma pessoa metódica, determinada, paciente (e haja paciência), objetiva, lógica, fria mas, também, criativa, apaixonada e "doida". Confuso novamente?

Parece engraçado, é verdade, mas para ser programador você precisa ser coisas bem contraditórias. Se você não for capaz de pensar com absoluta lógica não terá como escrever um programa que funcione por que as instruções devem ser passadas sempre em seqüência lógica, mas se não for "doido" será apenas um "miquinho adestrado", ou seja, será, apenas, capaz de escrever os programas que "aprendeu" a fazer, se alguém te pedir algo diferente que você não tenha "aprendido", ou seja, não tenha de onde copiar, você será incapaz de fazê-lo. Infelizmente mais de 90% dos profissionais formados por escolas, cursos técnicos e faculdades de ciências de computação são assim. Pasteuriza-se a programação e se entregam certificados para quem sabe copiar melhor o que foi passado, aliás, a própria programação de alto nível tem seguido para essa estrutura "tijolinhos lego" (.Net) na qual, ao invés de programar, que é difícil e exige um cérebro capaz de pensar, tem-se apenas que montar objetos pré-definidos com quase ou nenhuma linha de programação.

Você precisa ser "doido" para ser capaz de imaginar o processamento dos dados, os comandos em algoritmos que não existem e isso é muito doido mesmo, por que estamos falando de coisas que nem existem no mundo "real".

Você precisa ser frio, para agüentar as frustrações de partes de código que você tinha certeza que iriam funcionar mas que, por uma razão qualquer, não funcionam como você queria, te obrigando a passar horas, às vezes dias, analisando a falha até sua solução. Ao mesmo tempo tem que ser apaixonado, por que ficar concentrado por horas, dias, meses até, num único projeto, cujo código pode, se impresso, preencher centenas de páginas, trabalhando ininterruptamente por mais que surjam erros ou que o orçamento esteja apertado é só para quem realmente "ama" isso.

## Qual a diferença entre programador e desenvolvedor?

Há uma diferença técnica fundamental: o programador apenas programa.

Um projeto de software complexo normalmente envolve equipes multidisciplinares. Aos Analistas de Sistema compete fazer o projeto do programa, por assim dizer. Eles devem dizer que informações o programa requererá, quando fará isso e por que meios, o que fará com essas informações e como fornecerá o resultado de seu trabalho para o usuário. Um "Gerente de Projeto" é o sujeito que conversa com os clientes, que querem o programa, com a equipe de Analistas e, muitas vezes, também cuida do orçamento e do cronograma.

Da forma como o projeto do programa é feito depende diretamente o tempo que levará para sua execução. Um projeto bem feito, finalizado e detalhado facilita a vida dos programadores à beça, já projetos sem o devido detalhamento ou se ocorrem mudanças durante a fase de programação são desastrosos e podem arrastar o desenvolvimento do programa indefinidamente.

O programador é o "escravo", o sujeito que vai pegar o projeto e transformá-lo no produto final: o software, gastando seus "dedinhos", neurônios e vistas.

Já um desenvolvedor é uma pessoa que faz o serviço completo, ou seja, elabora o projeto, interfaces, sistema e, no fim, escreve o software.

## **É boa a profissão de PROGRAMADOR?**

Qualquer um pode, por hobby, fazer um ou outro programinha, joguinho, etc., você não precisa de nenhum talento especial para isso e é tão bom como exercício ao raciocínio quanto jogar Xadrez, por isso, se quiser um passatempo que lhe ajude a melhorar a acuidade mental, pense na possibilidade de aprender programação, sem compromissos, apenas para exercitar a mente mesmo.

Mas, se quer trabalhar na área, só o faça se "AMA" programar.

Por quê?

Primeiro lugar, programadores não são ricos. Um salário para programador em início de carreira é pouco superior a mil reais em média. Um salário médio para programador fica pouco acima de R\$ 3.000, ou seja, igual ao de um policial rodoviário em início de carreira. Um bom salário raramente ultrapassa R\$ 5.000.

Um fiscal em início de carreira, dependendo do município, ganha mais de R\$ 7.000, cargos no judiciário também ultrapassam esse valor. Em empresas privadas inúmeras profissões bem menos exigentes, desgastantes, estressantes e insalubres pagam muito mais. Enfim, pelo nível intelectual exigido para uma pessoa ser uma boa programadora, ela facilmente conseguiria empregos melhores em outras áreas.

Além do que, a maior parte do mercado que existe para "programadores" é o de suporte. Suporte por assistência, por treinamento ou por customização de programas para adaptá-los às necessidades dos clientes. Assim o campo do desenvolvimento de novas tecnologias, programação de ponta, desenvolvimento de novos produtos é restrito a muito poucos que conseguem se destacar ou que tem muita sorte.

Acho que todos ouvimos dizer que as grandes empresas de software do mundo todo correm à caça de programadores, mas não pense que basta ter um diploma. Elas não querem os miquinhos-amestrados que saem das faculdades de ciências da computação, de cursos técnicos ou de certificação que só sabem fazer o que aprenderam. Elas estão atrás de PROGRAMADORES, pessoas capazes de CRIAR soluções novas, de desenvolver tecnologias, de solucionar problemas novos quer tenham faculdade, curso, certificação ou não, o que elas querem é TALENTO, não papel carimbado.

Por isso essas empresas tem tanta dificuldade em achar o tipo de mão de obra que querem. Não estou te dizendo, com isso, que você não deva fazer um curso ou faculdade, se você tem esse talento sabe muito bem que quanto mais informação e estudo tanto melhor! Mas, se não tem esse talento, essa "vocação", não se iluda: não será uma faculdade que lhe dará isso.

Claro que você pode pensar: vou ser programador, escreverei um programa de sucesso, um jogo e ficarei rico!

É uma idéia boa, pena que o mundo não funciona desse jeito.

Grandes empresas de jogos, com jogos de sucesso, como a 3DO faliram, dezenas de milhares de programas são lançados todos os anos, quantos fazem sucesso? Será que os que não fizeram são ruins?

Não, não são. Mas há muita coisa excelente sendo lançada continuamente, a maioria não conseguirá ganhar o suficiente para pagar o custo de desenvolvimento. Empresas de software que começam cheias de sonho fecham pouco tempo depois, mesmo tendo feito ótimos produtos.

### **Essa é a verdade!**

Não é o que você vê na mídia? Não, não é. Mas a mídia só mostra o que faz sucesso.

Visite sites de download de softwares e você verá a enorme quantidade de programas que eles tem, e, mesmo o gigantesco [www.Tucows.com](http://www.Tucows.com) só tem uma parte pequena do que é lançado todo o tempo.

Veja se tudo o que tem lá aparece na mídia?

Se tiver paciência, visite alguns sites desses produtos e volte a visitá-los depois de um ano. Muitos dos sites ainda serão os mesmos (se ganharam muito dinheiro por que não mudariam o visual do site?), outros sites talvez nem mais estejam no ar.

E quanto aos jogos?

Nossa, quantos títulos novos, fantásticos, não surgem todos os anos?

Muito dinheiro ( milhões de dólares ) são gastos na produção desses jogos, dezenas ou centenas de milhões para as campanhas de marketing e, no fim, boa parte desses títulos fecha no vermelho, alguns com resultados decepcionantes, principalmente por causa da pirataria.

Deve desistir então?

Não, claro que não. Mas não tenha fantasias.

O universo dos programadores é um mundo silencioso de trabalho meticuloso, desgastante (fisicamente, mentalmente), estressante, que exige alto nível intelectual, técnico, paga muiiiiito mal e não há mágica: a pirataria destrói qualquer incentivo que você possa ter em montar sua empresa.

Qual é a recompensa?

É ver que idéias que estavam em sua mente tomam forma, "criam vida" e vão divertir, entreter, auxiliar pessoas em seus trabalhos, lares, talvez mundo afora. É saber que você faz diferença, ainda que possa não ser muita. Veja, como exemplo, o jogo Roller Coaster Tycoon: ele foi feito por um único desenvolvedor (Chris Sawyer - que "só para humilhar" escreveu quase todo o jogo em Assembler (rs)), jogo esse que fez enorme sucesso em todo o mundo divertindo e entretendo dezenas de milhares de pessoas e arrebanhando uma legião de fãs.

Por isso, se quer ser um profissional de programação lembre que ela é uma Arte, encare-a assim. Se é sua vocação, se é algo que "ama", estude, dedique-se, não importa se o mercado de trabalho não é bom, por que você vai conseguir se destacar e, no fim, ou fará algum software de muito sucesso, ou será cobiçado pelas grandes desenvolvedoras mundiais.

Claro que se você quer trabalhar na área de suporte, então seu mercado de trabalho será fantástico. Um profissional bem qualificado em informática tem emprego garantido. A chance de trabalhar em boas empresas, nacionais e estrangeiras é elevada, principalmente se você tiver muita disposição para estar sempre aprendendo e estiver "antenado" às novidades tecnológicas. Mas não dê preferência aos jogos não, o que realmente "paga" nesse mercado são produtos para empresas, bases de dados, segurança e comunicação.

### **Como é o dia-a-dia do Programador?**

Esqueça os estereótipos. Um programador é um profissional como qualquer outro. A idéia que somos uns "nerds" com óculos fundo de garrafa, anti-sociais e esquisitos, que falam rapidamente e normalmente coisas que ninguém consegue entender é bobagem de filmes. Alguns adolescentes, às vezes, que gostam de informática tendem a se deixar levar por essa imagem ridícula, mas, na verdade, um programador comum é uma pessoa que casa, tem família, filhos, vida social, amigos, contas para pagar, passeia, exatamente como todas as demais pessoas. É comum sermos mais quietos que os demais e falarmos de forma mais resumida, isso são vícios do trabalho. Dos programadores que conheci poucos usavam óculos (a maioria usa lentes) mas isso é quase inevitável por causa do trabalho intenso com o computador. Poucos programadores tem inteligência "privilegiada", ou seja, a maioria é como qualquer outra pessoa. Programar exige conhecimento, atenção e concentração, não exige QIs altíssimos não.

Diferente de muitas outras profissões, porém, o trabalho no desenvolvimento de softwares ocorre quase 100% no cérebro do programador. Tudo é pensado, imaginado, analisado mentalmente. Acho que a programação é uma das poucas profissões nas quais você não tem como reagir de forma automática, você precisa de sua "massa cinzenta", de toda a capacidade dela.

Quando um programador "de verdade" está escrevendo um software ele não tem como parar por que deu a hora de ir embora, por que, se ele fizer isso, muita coisa pendente à fazer, que está anotado apenas em sua memória mental, poderá ser esquecida. Ele tem que terminar o trabalho ou deixá-lo num estágio que permita ser retomado no dia seguinte. Alguns programadores continuam seus trabalhos em casa, outros varam a noite trabalhando. Normalmente um programador aprende a ser "frio" por que não se pode deixar que problemas pessoais afetem o raciocínio, sob pena de prejudicar substancialmente o trabalho. Um programador não pode usar drogas e, em geral, não gostam muito de bebidas alcoólicas menos ainda de cigarro. Ou seja, procuram manter sua ferramenta de trabalho - o cérebro - na melhor forma possível.

O único vício que se nota em um programador é o próprio trabalho.

A concentração no trabalho também é fundamental, às vezes eu digo que o melhor lugar para um programador trabalhar seria um mosteiro. São necessários o silêncio e ausência de interrupções para que o profissional consiga concentrar-se adequadamente. Também é importante dispor de locais "contemplativos" para que o programador possa caminhar ou "descansar" sem ser incomodado por que ele trabalha em sua mente primeiro, é depois que ele "coloca as coisas" no computador. Bill Gates, que é programador, sabe muito bem disso e é por isso que emprega horas flexíveis e locais para "meditação" para os programadores da Microsoft.

Mas você, se conhecer alguém que trabalhe com programação, deverá notar que parece ser uma pessoa distraída, às vezes "desmemoriada", mas não se engane: a distração se dá por que estamos pensando em muitas outras coisas ao mesmo tempo e a memória geralmente está ocupada por pendências e detalhes de projetos nos quais estamos trabalhando e, por isso, informações menos "importantes" acabam não sendo assimiladas.

E, para finalizar: programador que se preze "não vive" sem café! =)

*O computador é a ferramenta que transforma a imaginação em algo coletivo.*