

JESSICA LIVINGSTON

# Startup

FUNDADORES DA APPLE, DO YAHOO!, HOTMAIL,  
FIREFOX E LYCOS CONTAM COMO NASCERAM  
SUAS EMPRESAS MILIONÁRIAS

**A**

---

NEGÓCIOS

Startup

# Startup

FUNDADORES DA APPLE, DO YAHOO!,  
HOTMAIL, FIREFOX E LYCOS  
CONTAM COMO NASCERAM SUAS  
EMPRESAS MILIONÁRIAS

Jessica Livingston

REVISÃO TÉCNICA  
Carlos Irineu da Costa



---

NEGÓCIOS

Título original  
*Founders at Work*

Edição em língua inglesa publicada por Apress Inc., 2560 Ninth Street,  
Suite 219, Berkeley, CA 94710 USA.

Copyright © 2007, Apress, Inc..

Copyright da edição brasileira © 2009, Agir Editora Ltda. Todos os direitos reservados.

*Capa*  
Rita da Costa Aguiar

*Revisão*  
Patrícia Reis

*Produção editorial*  
Juliana Romeiro

CIP-Brasil. Catalogação-na-fonte. Sindicato Nacional dos Editores de Livros, RJ.

---

L762s Livingston, Jessica

Startup: fundadores da Apple, do Yahoo, Hotmail, Firefox e Lycos  
contam como nasceram suas empresas milionárias / Jessica Livingston;  
[tradução Marilena Reginato de Moraes Souza]. — Rio de Janeiro: Agir,  
2009.

Tradução de: Founders at work: stories of startups' early days  
ISBN 978-85-220-0994-7

1. Empresas novas — Estados Unidos — Estudo de casos. 2. Indústria  
eletrônica — Estados Unidos — Estudo de casos. I. Título.

09-0021

CDD: 338.040973

CDU: 330.342.146(73)

---

Todos os direitos reservados à Agir Editora Ltda. — uma empresa Ediouro Publicações S.A.

Rua Nova Jerusalém, 345 — CEP 21042-235 — Bonsucesso — Rio de Janeiro — RJ

Tel.: (21) 3882-8200 fax: (21) 3882-8212/8313

# Sumário

7	PREFÁCIO, MARCOS WETTREICH
9	AGRADECIMENTOS
11	INTRODUÇÃO
15	CAPÍTULO 1. Steve Wozniak, co-fundador, Apple Computer
47	CAPÍTULO 2. Mike Lazaridis, co-fundador, Research In Motion (criadora do BlackBerry)
61	CAPÍTULO 3. Sabeer Bhatia, co-fundador, Hotmail
77	CAPÍTULO 4. Evan Williams, co-fundador, Pyra Labs (Blogger.com)
97	CAPÍTULO 5. Tim Brady, primeiro funcionário contratado, Yahoo!
115	CAPÍTULO 6. Craig Newmark, fundador, Craigslist
125	CAPÍTULO 7. Caterina Fake, co-fundadora, Flickr
135	CAPÍTULO 8. Charles Geschke, co-fundador, Adobe Systems
157	CAPÍTULO 9. Blake Ross, criador, Firefox
171	CAPÍTULO 10. Bob Davis, fundador, Lycos
183	CAPÍTULO 11. Max Levchin, co-fundador, PayPal
205	CAPÍTULO 12. Mitchell Kapor, co-fundador, Lotus Development

- 225 CAPÍTULO 13. Steve Perlman, co-fundador, WebTV
- 245 CAPÍTULO 14. Mark Fletcher, fundador, ONElist, Bloglines
- 263 CAPÍTULO 15. David Heinemeier Hansson, sócio, 37signals
- 273 CAPÍTULO 16. Paul Buchheit, criador, Gmail

## Prefácio

Antes de mais nada, empreender é muito mais difícil do que se supõe. O nível de dedicação necessário é bem maior do que imaginam aqueles que nunca tiveram a oportunidade de iniciar um negócio. Além disso, para ser bem-sucedido e criar uma empresa realmente inovadora, não existe leitura nem manual que realmente ensine o que fazer. Por isto mesmo Jessica Livingston foi especialmente bem-sucedida por escrever este livro. Ao trazer para estas páginas as experiências de empresários inovadores que conseguiram criar negócios reais e diferenciados, especialmente em uma área em que tudo está em constante inovação (tecnologia), ela conseguiu transmitir algo que normalmente custaria muitos e muitos anos e enorme dispêndio de energia, e mesmo capital, para ser adquirido: a experiência.

A escritora acertou também na decisão sobre a escolha dos empreendedores, evitando entrevistar os Gates e Jobs fartamente encontrados nas revistas de negócios. Aqui temos inovadores que aprenderam a importância da persistência, que descobriram que planos de negócio (*business plans*) são necessários mas estão sujeitos a mudanças a todo momento, e que é preciso ser apaixonado ou até paranóico para fazer seu negócio prosperar. Pessoas que seguem em frente mesmo contra o bom senso vigente (até mesmo em confronto com parentes e amigos mais próximos), confiando quase cegamente em sua convicção, e que constatam que muitas vezes ter uma idéia original não é o que faz com que uma empresa seja bem-sucedida.

Apesar de a maneira mais efetiva de aprender ser a prática, nas páginas deste livro o leitor ganhará indiretamente parte desta experiência que tanto custou ser adquirida por outros empreendedores. E direto da fonte! Como ter a idéia

que me deixará rico? Como convencer investidores a investir dinheiro em uma nova idéia? O que fazer quando algo dá errado?

É claro que não existem receitas únicas ou mágicas para responder a estas perguntas, e o que funcionou para um inovador aqui entrevistado não necessariamente funcionaria para um negócio que você, leitor, possa ter em mente. Mas, sem dúvida, elementos contidos nestes relatos terão grande valia quando aplicados a novas iniciativas. Até mesmo as contradições que podem ser encontradas nas várias visões contidas neste livro são, por si só, lições. Elas demonstram que os caminhos são os mais diversos possíveis.

Leia, reflita, aprenda e aja. Não existe realização sem execução, e quanto mais aprender antes de seus primeiros passos, maiores serão suas chances. Boa sorte.

Marcos Wettreich



## Agradecimentos

Em primeiro lugar quero agradecer à minha tia, Ann Gregg, por seu apoio incontestado e seu incentivo. Ela é uma leitora de incrível percepção e seus conselhos ajudaram a tornar este livro melhor.

Agradeço aos entrevistados por partilhar suas histórias, seu tempo. Há um detalhe em relação aos entrevistados que não mencionei na introdução: como *gostei* deles. São verdadeiros e inteligentes, foi uma honra ter essa oportunidade. Sei que a natureza ingênua de suas histórias e de seus conselhos vai inspirar outros fundadores em potencial.

Agradeço a Gary Cornell por estar disposto a fazer uma espécie diferente de livro e à equipe da Apress por trabalhar numa publicação diferente.

Quero agradecer a muitas pessoas pela disposição em me apresentar a tantas outras: Jim Baum, Patrick Chung, Mark Coker, Jay Corscadden, Rael Dornfest, Jed Dorsheimer, Randy Farmer, Steve Frankel, Anand Gohel, Laurie Glass, James Hong, Mitch Kapor, Morgan Ley, Mike Palmer, Tom Palmer, Bryan Pearce, Andrew Pojani, Will Price, Ryan Singel, Langley Steinert, Chris Sacca e Zak Stone.

Agradeço a Kate Courteau pelos escritórios aconchegantes que criou para eu trabalhar; a Lesley Hathaway pelo apoio e pelos conselhos; a Alaina e David Sloo por me apresentarem a muitas pessoas; e a Sam Altman, Paul Buchheit, Lynn Harris, Marc Hedlund e Aaron Swartz, leitores dos primeiros capítulos. Meu agradecimento a Lisa Abdalla, Michele Baer, Jen Barron, Ingrid Bassett, Jamie Cahill, Jessica Catino, Alicia Collins, Caitlin Crowe, Julie Ellenbogen, John Gregg, Chrissy Hathaway, Katie Helmer, Susan Livingston, Nadine Miller, Sara Morrison, Bridget O'Brien, Becky Osborne, Allison Pellegrino, Jennifer Stevens e a Suzanne Woodard pelo incentivo.

Agradeço aos que compartilharam seus insights sobre empresas startups em jantares na Y Combinator ou pessoalmente: Rich Bacon, Greg Benning, Tom Chur Chill, Michael Ellenbogen, Jonathan Gertler, Hutch Fishman, Sara Harrington, Bill Herp, Bradley Horowitz, Joel Lehrer,Carolynn Levy, Simon London, Page Mailliard, Udi Manber, Fredrick Marckini, Greg McAdoo, Mark Macenka, Mike Mandel, Jerry Michael, Rich Miner, Mark Nitzberg, Peter Norvig, Steve Papa, Tom Pinckney, Stan Reiss, Olin Shivers, Hugues Steinier, Jeff Taylor, Rob Tosti e Stephen Wolfram.

Agradeço aos fundadores das startups que financiamos na Y Combinator. São inspiradores e sei que têm histórias valiosas para compartilhar.

Agradeço em especial a Trevor Blackwell e a Robert Morris pelo grande apoio. Tenho a sorte de trabalhar com eles.

Agradeço a meus avós, Baba e Bob, a quem admiro e cujos conselhos como escritores muito me ajudaram. Agradecimentos especiais a papai e a Michele, que me apoiaram mesmo quando tive idéias loucas como deixar o emprego para fundar uma empresa e escrever um livro. Ao longo dos anos, meu pai jamais pareceu duvidar de que eu poderia fazer alguma coisa de que ele realmente se orgulhasse, e aprecio isso.

## Introdução

Uma espécie de mágica acontece nas startups, especialmente bem no começo, mas apenas os fundadores estão lá para ver. A melhor maneira de compreender o que acontece é perguntar a eles; e foi o que fiz.

Neste livro, você vai conhecer as histórias dos fundadores das startups por suas próprias palavras. Quero compartilhar alguns dos padrões que percebi. Quando se entrevista uma série de famosos fundadores de startups, não se pode evitar a tentação de buscar alguma qualidade especial em comum que os tenha tornado bem-sucedidos.

O que mais me surpreendeu foi como eles não tinham realmente certeza de estarem iniciando alguma coisa de grandes proporções. Algumas dessas empresas começaram quase por acidente. O mundo pensa que esses fundadores têm algum tipo de confiança super-humana, mas muitos estavam inseguros quanto a iniciar o negócio. Tinham a certeza, porém, de estar fazendo algo de bom — ou de estar tentando consertar algo.

Todos estavam determinados a construir coisas que dessem certo. Na verdade, eu diria que a determinação é a qualidade mais importante para um fundador de startup. Se as pessoas que entrevistei tinham algo de super-humano era a perseverança. Isso apareceu repetidamente nas conversas.

Perseverança é importante porque, numa startup, as coisas não seguem um plano. Os fundadores vivem o dia-a-dia com uma dose de incerteza, isoladamente e, às vezes, falta de progresso. Além disso, as startups, pela própria natureza, fazem coisas novas — e quando se faz coisas novas, a rejeição é frequente.

Este foi o segundo aspecto surpreendente que aprendi com as entrevistas: como os fundadores foram de início rejeitados; por investidores, jornalistas,

empresas tradicionais. As pessoas gostam da idéia de inovação no abstrato, mas tendem a rejeitar uma inovação concreta que lhes é apresentada porque não está de acordo com o que já conhecem.

Se fizermos uma retrospectiva, veremos que inovações foram inevitáveis, mas, na época em que surgiram, travou-se uma dura batalha. É curioso pensar que a tecnologia que hoje nos parece natural, como a troca de mensagens por e-mail, já foi dispensada por não ser promissora. Como disse Howard Aiken: “Não tema que suas idéias geniais sejam roubadas. Se elas forem realmente boas, será tão difícil as pessoas aceitarem que você terá de enfiá-las pelas goelas delas.”

Além de ter perseverança, os fundadores de startups precisam ser adaptáveis. Não apenas porque é necessário um certo nível de flexibilidade mental para entender o que os usuários desejam, mas porque o plano provavelmente vai mudar. As pessoas pensam que as startups surgem de uma brilhante idéia inicial, como se fosse uma semente que resulta numa planta. Mas quase todos os fundadores que entrevistei foram mudando suas idéias à medida que as desenvolviam. A PayPal começou como um software de criptografia, a Flickr veio de um jogo on-line. Iniciar uma startup é um processo de tentativa e erro. O que guiou os fundadores nesse processo foi sua empatia pelos usuários. Sempre visaram a fazer o que as pessoas queriam. Os fundadores de startups de sucesso, em geral, ficaram ricos, mas os que entrevistei não estavam no negócio apenas pelo dinheiro. Tinham muito orgulho do trabalho braçal. Queriam mudar o mundo e, por isso, muitos iniciaram novos projetos igualmente ambiciosos. É claro que têm prazer em ter mais liberdade financeira, mas escolhem usar o dinheiro construindo mais novidades.

As startups bem-sucedidas são diferentes de empresas estabelecidas — é mesmo surpreendente como começaram. Seria bom se as pessoas prestassem mais atenção nesse nicho tão importante do mundo dos negócios, embora quase sempre incompreendido. É nele que está a essência da produtividade. Na sua forma mais simples, a produtividade parece tão estranha que a impressão, para muitos, é que nada tem a ver com negócios. Mas se o início

das startups foi assim, o mundo empresarial poderia ser mais produtivo se funcionasse da mesma forma.

Meu objetivo com essas entrevistas foi registrar um conjunto de experiência para ajudar a todos. Você perceberá certos problemas que são recorrentes. Todos os fundadores desejavam ter conhecido certas coisas quando estavam começando. São coisas que agora estão à disposição para futuros fundadores em potencial.

Espero sinceramente que este livro inspire pessoas que desejam iniciar seus negócios. A fama que vem com o sucesso faz os fundadores de startups parecerem uma espécie diferente. Se as pessoas puderem saber como essas empresas realmente começaram, talvez tenham mais coragem de iniciar um negócio próprio. Espero que muitos que leiam essas histórias pensem: “Esses caras já foram como eu. Talvez eu também consiga.”

## Steve Wozniak

CO-FUNDADOR DA APPLE COMPUTER

Steve Wozniak pode ser considerado o homem que deu início à revolução dos computadores pessoais. Foi ele quem projetou a máquina que reuniu todos os elementos do computador desktop ao criar o Apple II.

Wozniak e Steve Jobs fundaram a Apple Computer em 1976. Com a habilidade técnica de Wozniak e a energia magnética de Jobs, eram uma dupla e tanto. Naquele ano, Woz mostrou pela primeira vez o Apple I, no Homebrew Computer Club no Vale do Silício — era uma reunião periódica de amadores que projetavam e construíam computadores pessoais numa época em que eles ainda não vinham prontos dentro de caixas.

A Apple deslanchou rapidamente depois que Jobs conseguiu um contrato com uma loja local de computadores para fornecer cem máquinas montadas e prontas para serem usadas.

No ano seguinte, na West Coast Computer Faire — considerada a primeira feira de computadores pessoais, na época um hobby para fãs de eletrônica —, Woz apresentou a máquina que deu início à empresa e colocou os computadores pessoais, pela primeira vez, ao alcance de usuários leigos: o Apple II. Tanto o hardware quanto o software do Apple II foram projetados por Wozniak, um feito extraordinário e talvez ainda mais impressionante se levarmos em conta que, enquanto fazia tudo isso, Wozniak trabalhava em tempo integral na Hewlett-Packard.

Em 1980, menos de cinco anos após sua fundação, a Apple Computer abriu seu capital, na maior oferta pública inicial desde a Ford Motor Company, em 1956, criando mais milionários instantâneos do que qualquer outra empresa até então.

*Jessica: Como você se envolveu com computadores?*

Wozniak: No ensino médio, eu já sabia projetar computadores usando apenas metade dos chips que outras empresas da época utilizavam. Aprendi tudo sozinho, de uma forma que me fez conhecer todos os macetes dos componentes então disponíveis. Na vida, temos que valorizar aquilo em que somos bons, e meu grande talento era criar projetos com poucos componentes usando todos os recursos possíveis. Eu valorizava qualquer produto feito com esta filosofia.

Isso me ajudou de duas formas. Quando se está em uma startup ou se trabalha sozinho, há pouco dinheiro e quanto menos componentes for preciso comprar, melhor. Além disso, quando você usa poucos componentes, tudo se torna tão claro e ordenado que você consegue compreender as coisas mais a fundo em sua mente. Você também comete menos erros. É possível passar seus dias e noites com todos os detalhes do produto em mente.

Poucos anos antes de criar a Apple, estava trabalhando na Hewlett-Packard projetando calculadoras científicas. Era uma grande oportunidade trabalhar com um produto importante naquele momento. O que me levou a fundar uma empresa, contudo, era o que eu fazia fora da HP. Quando voltava para casa, continuava projetando equipamentos eletrônicos, porém não eram mais as calculadoras que criávamos na HP. Através de conhecidos, estava envolvido com os primeiros jogos domésticos. Descobri os videogames: o primeiro que realmente deu certo foi o *Pong*, então resolvi desenvolver a minha própria versão. A Atari queria usar meu projeto para criar o primeiro *Pong* para usuários domésticos. Queriam que eu colocasse o jogo em um único chip, o que seria mais econômico, considerando-se o volume de vendas que teriam. Foi uma idéia de Steve Meyer. Eu tinha uma boa relação com a Atari e eles reconheciam meu talento como projetista, então queriam me contratar.

*Jessica: Como a Atari chegou até você?*

Wozniak: Steve Jobs trabalhava lá em meio expediente, finalizando os jogos desenvolvidos em Grass Valley. Ele me chamou e me mostrou o que estavam fazendo. Na mesma hora, Nolan Bushnell me ofereceu um emprego. Recusei, dizendo que jamais deixaria a Hewlett-Packard, onde achava que iria trabalhar toda a minha vida. Na época, era a melhor companhia para os engenheiros, pois nos tratavam como se fôssemos uma comunidade, uma família, e todos cuidavam uns dos outros. Os engenheiros — os últimos no organograma — tinham a liberdade de apresentar idéias que depois poderiam se transformar no próximo produto importante da empresa. Estavam sempre abertos à reflexão, às conversas, à inovação.

Em seguida desenvolvi um jogo para a Atari chamado *Breakout*, um produto incrível. Era ótimo ter meu nome associado a um videogame que de fato foi lançado comercialmente. Estávamos vivendo o começo de uma indústria da qual eu não fazia parte, mas queria ser um desenvolvedor e ter algum vínculo com aquela nova área.

Em meio a todos esses projetos, acabei me envolvendo em outro. Naquele momento, a Arpanet — embrião da internet atual — tinha no máximo uma dúzia de computadores conectados em rede. Era possível escolher um computador para visitar e você teria uma conta com acesso limitado, aquilo que chamamos de “guest account”, uma conta temporária para usuários externos. Aqueles que tivessem uma senha de usuário podiam ir mais longe e fazer outras coisas. Para interagir com esses computadores de grande porte, raramente havia um terminal de vídeo, então usávamos teletipos como dispositivo de entrada e saída de dados. Uma vez vi alguém digitando em um terminal de teletipo e falando sobre jogar xadrez com um computador em Boston. Foi quando pensei que realmente queria fazer aquilo, que precisava ter aquilo, de alguma forma. É assim que muitos empreendedores começam: vêem uma coisa e pensam que precisam daquilo. Então projetam algo para si mesmos e fundam uma nova empresa.

Eu não tinha como comprar as peças de que precisava — não podia comprar um teletipo, por exemplo, porque era muito caro — por isso tive que



projetar meu próprio terminal. A única coisa que eu podia obter de graça era uma televisão comum para ver os caracteres que estavam sendo digitados ou aquilo que o computador gerava em resposta. Comprei um teclado por sessenta dólares, o que, na época, era incrivelmente barato. Foi o item mais caro desse terminal.

Em seguida, precisei desenvolver circuitos e códigos de computador para colocar numa tela de TV pontos que formassem as letras do alfabeto, a fim de escrever o que vinha do computador remoto, conectado a uma rede. Usando o teclado, era possível digitar de volta os dados para alimentar esse mesmo computador. Para estabelecer a comunicação entre meu terminal e o computador remoto, construí um modem — um aparelho capaz de transferir dados usando uma linha telefônica normal — e foi assim que transformei uma tela de TV num monitor. É preciso lembrar, contudo, que eu ainda estava trabalhando na HP e fazia essas coisas só por diversão.

*Jessica: Foi quando você freqüentava o Homebrew Computer Club, certo?*

Wozniak: Exato. Era um clube composto essencialmente por jovens, pessoas que adoravam montar coisas e colocá-las para funcionar. Todos que estavam lá poderiam ter se tornado empresários, mas poucos eram desenvolvedores, realmente capazes de criar coisas novas. A maioria pensava nos circuitos com a mente de um técnico: soldando coisas, examinando-as, verificando os circuitos para saber se estavam recebendo as voltagens certas. Havia também muita gente ligada a software, mas sem experiência em hardware, o que era essencial para construir as primeiras máquinas.

Bem no início, eu ainda pensava que precisava construir tudo de graça. Então, descobri que os microprocessadores tinham surgido. De certa forma, eu estava meio à parte do mundo da eletrônica, dos computadores, por causa do meu trabalho com calculadoras. Quando descobri os microprocessadores, fiquei intrigado, pensando sobre o que era aquilo. Não conseguia entendê-los totalmente, por isso comecei a ler a respeito. Eu me sentia desconfortável porque o mundo, de certa forma, havia se antecipado a mim — subitamente havia microcomputadores pequenos e baratos baseados em microprocessadores, mas

eu sequer tinha ouvido falar no assunto, não fizera parte daquilo. Numa noite, analisei um microcomputador e descobri que era igual aos que eu projetava no ensino médio e que, aliás, eram muito bons.

Um dia pude examinar o computador Altair, que dera início a tudo. Foi o primeiro microcomputador, mas não era realmente um computador. Não para mim. Eu só precisava de uma coisa. No segundo grau, disse a meu pai que teria um Data General Nova com memória de 4k. Essa é a quantidade mínima de memória necessária para executar uma linguagem de programação. Você tem que ser capaz de codificar em Fortran ou Basic ou em alguma outra linguagem para criar novos programas, fazer coisas novas.

O Altair, que estava sendo vendido a um preço ridiculamente baixo, não era mais que uma versão melhorada de um microprocessador da Intel. Tudo o que fizeram foi lançá-lo no mercado dizendo que, a partir daquilo, era possível conectar todas as coisas que um microprocessador normalmente já aceitaria.

Era possível adicionar RAM (Random Access Memory), placas que se comunicavam com teletipos, ou até comprar um teletipo por alguns milhares de dólares... Para ter um computador que realmente executasse uma linguagem de programação, contudo, era necessário gastar uns cinco mil dólares, algo fora do alcance de quase todos nós, que estávamos apenas começando e queríamos explorar aquele mundo. A turma do clube não tinha tanto dinheiro assim, mas todos queriam ter seus próprios computadores.

Eu me lembro também que, cinco anos antes, em 1970, eu tinha projetado e construído um computador que era exatamente igual ao Altair. Chamava-o de "Cream Soda", por conta de uma história que contarei mais adiante. Como eu não tinha um microprocessador naquela época, tive que construí-lo usando chips. Construí um processador que ficava em uma placa muito pequena, cerca de 9 x 12 cm. Tinha interruptores, luzes, parecia uma cabine de avião, exatamente como o Altair. E tinha tanta memória quanto o Altair (a memória inicial era de apenas 256 bytes). Eu podia configurar os interruptores, apertar botões, colocar seqüências binárias na memória e executá-las como um programa, e também podia ver que o código realmente estava lá, sendo executado.

Novamente, já tinha feito aquilo cinco anos antes. Na época eu estava vendo o Altair e os microprocessadores e sabia que aquilo não bastava. Era necessário algo mais para rodar uma linguagem de computador completa. Contudo, a eletrônica havia avançado, estávamos mais próximos.

Então fui pesquisar. Quando projetava qualquer coisa, queria que fosse viável e que não fosse algo caro, mas precisava no mínimo de 4k de memória RAM. As primeiras RAMs dinâmicas de 4k foram introduzidas em 1975. Era a primeira vez que as RAMs custavam menos do que as memórias com núcleos magnéticos que os computadores usavam até então. De repente, o mundo inteiro estava passando a usar RAMs. O silício se tornaria a nossa memória.

Todo o resto — o Altair, os computadores da Sphere, da Polymorphic e da Insight — eram máquinas projetadas por engenheiros essencialmente incapazes, e não engenheiros de alto nível. Eram técnicos que sabiam como comparar as folhas de dados dos chips de RAM com as de um microprocessador, achar o ponto onde cada diagrama especificava “endereçamento” e conectar as duas coisas com um fio. É um trabalho muito simples enquanto as RAMs são estáticas.

As RAMs dinâmicas custariam metade ou mesmo um quarto do preço. Com elas, o computador teria memória suficiente para rodar uma linguagem de programação com apenas oito chips de RAM, em vez de 32. Mas a RAM dinâmica, que continuamos usando em nossos computadores atuais, precisa de um circuito de apoio para acessar cada endereço individual da RAM a cada dois milésimos de segundo, ler o que está lá e escrever de volta, do contrário o valor existente naquele endereço se perde.

Resolver este problema me custou um bom tempo de reflexão e alguns circuitos extras, mas, quando montei meu computador, já tinha os contadores usados para criar seqüências regulares para a tela de TV do meu terminal, então decidi que usaria os mesmos contadores para fornecer os valores e dizer ao microprocessador, constantemente, que endereços deveriam ser atualizados. Isto exigiu um projeto mais sofisticado, mas, no fim, não só havia menos chips como também eram menores. Impressionava qualquer um que o visse. Era

mais barato e mais rápido. Você sempre consegue coisas baratas e rápidas se usar a abordagem certa.

No final dos anos 1960, surgiram diversos minicomputadores, todos usando os mesmos chips 7400, com quatro portas cada. E todos seguiam o mesmo padrão: estabelecer um conjunto básico de instruções que torna o computador utilizável. A partir daí é possível criar um sistema operacional e desenvolver linguagens de programação.

Então a Data General lançou o minicomputador Nova e, em vez de precisar de cinquenta instruções para realizar uma seqüência básica de operações matemáticas, tinha uma só: uma instrução de 16 bits. Todos esses bits tinham um sentido e executavam ações. Pude observar como o Nova funcionava quando estudei seu projeto. Era uma forma mais simples de lidar com as operações lógicas. Refiz o projeto, como sempre, e quando terminei minha versão, o “meu” Nova tinha quase metade dos chips de todos os outros minicomputadores da Varian, da Digital Equipment Corp., da Hewlett-Packard — todos os minicomputadores da época (eu estudava o projeto de todos eles). E com metade dos chips, o Nova era um computador tão bom quanto os outros. O que mudou? A arquitetura se ajustava ao menor número possível de chips.

*Jessica: Você estudava e refazia o projeto de todos esses computadores no ensino médio, em casa, só por diversão?*

Wozniak: Sim, porque nunca pude construir um de fato. Eu ia além, contudo, e refazia os projetos de cada um dos computadores sempre que novos chips apareciam. Pegava um dos novos chips e refazia o projeto de um daqueles computadores porque assim surgiam idéias sobre como economizar mais alguns chips. Só fazia tudo isso porque não tinha dinheiro, não podia construir um computador de fato. Naquela época os chips eram caríssimos: como eu disse, um computador funcional valia tanto quanto a entrada de uma boa casa. Então a única coisa que eu podia fazer era projetar as máquinas no papel e tentar otimizá-las o tempo todo. Competia comigo mesmo para ter idéias que ninguém mais teria. Foi assim que refinei minha habilidade.

Já sabia que eu usava muitas abordagens que nenhum ser humano utilizaria na prática. Nada daquilo sequer poderia ser ensinado formalmente na universidade. Fiz muitas coisas de cabeça, aprendi sozinho. Sequer havia computadores na escola onde eu estava cursando o ensino fundamental, naquela época. E mesmo assim eu os projetava. Dei sorte por ter conseguido alguns periódicos da área e descobri como obter manuais dos computadores, enquanto meu pai conseguia os manuais de chips para mim. Foi quando comecei a pensar em como se construía um computador partindo daqueles chips.

Eu tinha uma habilidade específica e um objetivo definido: queria um computador. Minha habilidade era a de combinar chips com muita eficiência para atingir o meu objetivo, mesmo sem ter projetado nada parecido antes. Não sabia como fazer todas as coisas, mas, quando chegava o momento de elaborar um projeto — um controlador de floppy disk, uma interface para modem ou impressora, o que fosse —, eu projetava as coisas do meu jeito, de forma nova, sem saber como as outras pessoas faziam. As melhores coisas que fiz na Apple vieram do fato de eu não ter dinheiro e de nunca tê-las feito antes.

*Jessica: Você chegou a entrar na faculdade mas depois abandonou o curso, certo?*

Wozniak: Não exatamente. Não saí nem abandonei a faculdade. Depois do primeiro ano, trabalhei por um ano em programação para pagar o terceiro ano. Depois do terceiro ano, fiquei fora um ano para trabalhar e ganhar dinheiro para fazer o quarto ano. Foi quando consegui o emprego na Hewlett-Packard. Que emprego incrível! Minha carreira começou a deslanchar. Tinha aqueles projetos paralelos que eu fazia e depois veio o Apple. Não tive oportunidade de voltar a estudar naquele momento, apesar de ter esse desejo.

Em 1981, sofri um acidente de avião que me deixou com amnésia. Cinco minutos após sair da amnésia, compreendi que era hora de voltar à faculdade. Não haveria outra oportunidade. Voltei e me formei. Sempre gostei da universidade e era um bom aluno, um estudante nota dez. Meus pais tinham formação universitária e eu achava que os meus filhos precisavam saber que o pai deles também tinha formação universitária. Ainda assim, não aprendi nada

sobre computadores na faculdade. Nunca tive uma aula sobre linguagens de programação, por exemplo.

*Jessica: Quando se deu conta de que podia construir o Apple I?*

Quando construí o Apple I, usei o terminal de TV que eu já tinha desenvolvido. Mas o Apple I era quase um protótipo: não havia sido projetado para ser eficiente, este seria o Apple II. O que fiz na época foi pegar o terminal que funcionava com meu aparelho de TV, um teclado e juntar alguns outros componentes.

Pensei: “Esses computadores que estão surgindo têm interruptores e luzes e parecem cabines de avião. E são como o Cream Soda que construí há cinco anos: lentos e sem graça!” Era legal ter um computador, mas ele ainda não fazia o que eu queria. Tinha vontade de desenvolver programas, fazer um jogo e jogá-lo, ou desenvolver um programa que resolvesse as minhas simulações para o trabalho na HP. (Eu usava o grande computador deles. O minicomputador era compartilhado por quarenta engenheiros. Para usá-lo, era necessário agendar um horário.)

Naquele ponto, já sabia que, para construir o computador que queria, precisava de um microprocessador. Já tinha descoberto que os microprocessadores eram muito similares àqueles minicomputadores que eu costumava projetar. Usar RAM dinâmica era a escolha ideal para economizar dinheiro e peças e, como disse, já possuía o terminal. Faltava, então, uma linguagem de programação, o que era possível agora, com 4k de memória. O problema era que ainda não havia linguagens de alto nível para aquele microprocessador da Motorola. Por um lado, fiquei um pouco desapontado, mas por outro me empolguei, porque eu seria o primeiro a escrever uma linguagem de programação para aquele processador. Iria ganhar um certo prestígio com isso. Como era muito tímido, para mim a única maneira de ser notado era projetando coisas importantes. Mas nunca havia implementado uma linguagem. Então, abri o manual da Hewlett-Packard no trabalho e estudei BASIC. Estudei todos os comandos e a partir deles desenvolvi minha própria implementação da linguagem. Além de criar uma versão dessa linguagem e de fazer o projeto do computador, eu também montava e soldava as peças. Eu também era o técnico: fiz de tudo nos meus projetos da Apple.

Voltando ao Apple I, eu soube que teria o computador com que havia sonhado a vida inteira no primeiro encontro do Homebrew Computer Club. Entendi isso naquela noite, quando descobri o que era um microprocessador. Fui para casa e estudei o assunto, pensando que teria que conseguir dinheiro para comprar um deles. Foi difícil porque o processador da Intel custava 400 dólares, e eu não tinha como conseguir tanto dinheiro rápido. Seria como conseguir 2 mil dólares hoje, não é fácil. Então descobri que havia um processador da Motorola que eu poderia comprar por 40 dólares na Hewlett-Packard, mas logo em seguida a Motorola lançou o 6502 por 20 dólares. Foi o que comprei porque era muito barato e também o melhor naquele momento.

A próxima etapa era construir o hardware. Examinei todos os computadores à minha volta, e percebi que eles sempre seguiam o padrão daqueles computadores antigos — interruptores e luzes, além de slots para inserir placas e conectá-los a teletipos. E pensei que não queria nada daquilo, queria algo completo porque, naquele momento, o preço era viável. Tinha meu terminal com um teclado para digitar. O problema era carregar o programa na memória. Estava começando com um microprocessador que nem mesmo tinha uma linguagem de programação, e aí não tem jeito, você tem que colocar seqüências de números binários diretamente na memória. Resolvi escrever um programa simples, 256 bytes armazenados em apenas dois chips. Meu programa lia o que era digitado no teclado e cumpria as funções do painel frontal, porém cem vezes mais rápido. Também podia mostrar na TV o que estava na memória. Era possível inserir dados e rodar um programa. Isso me possibilitou ir mais longe ao digitar meus uns e zeros. Enquanto desenvolvia minha versão de BASIC, precisava digitar os uns e os zeros manualmente, até chegar a um ponto em que levava quarenta minutos para carregar todo o meu programa na memória. Não digitava em binário; usava o sistema hexadecimal (números em base 16). Então carregava o programa na memória e testava diferentes partes a cada vez para ver o que acontecia. Este não era um projeto normal, em que você tem ferramentas. Não havia ferramentas; minha abordagem da vida era justamente usar meu próprio conhecimento. Tenho mais controle sobre o que está acontecendo se não usar uma ferramenta.

*Jessica: Além de sua TV, você tinha um gravador cassete para armazenar dados, não é?*

Wozniak: Tinha. Depois que o Basic chegou a certo ponto, tinha que armazenar um programa grande de forma eficiente em algum tipo de mídia. Usei um gravador para não ter que ficar digitando durante quarenta minutos. Mas isso foi bem no final, quando meu BASIC já estava quase pronto.

*Jessica: Você apresentou esse trabalho no Homebrew Computer Club?*

Wozniak: A cada duas semanas eu apresentava meu projeto, que eventualmente virou o Apple I. Eu o levava ao clube e mostrava às pessoas, além de levar os diagramas. No trabalho, tirava cópias de todos os meus diagramas e os distribuía no clube para ser conhecido por fazer coisas legais. Fazia questão de que meu nome estivesse em todos eles, pois eu era tímido e queria que me conhecessem por estar fazendo algo legal. Era também uma forma de dizer aos outros que eles podiam construir os próprios computadores, por um preço bem barato, e não do jeito que os fabricantes do Altair queriam. Estava tentando dizer que havia outro tipo de computadores, outra forma de pensar. Algumas pessoas sacaram, outras não.

*Jessica: As pessoas que entenderam tentaram construir seus próprios computadores?*

Wozniak: Ainda era um processo muito trabalhoso. Muitos ali eram ligados à programação e não ao hardware. Não eram técnicos, muitos participantes do clube sequer sabiam como soldar. Então Steve Jobs começou a dizer que devíamos abrir uma empresa, pois muitas pessoas queriam construir um computador mas, mesmo que comprassem os chips em um kit, não iam querer soldar as peças. Ele sugeriu que criássemos uma placa na qual as pessoas poderiam colocar seus chips e ter um computador pronto. Soldar peças em um circuito impresso é fácil, não há fios.

A idéia era fundar uma empresa, construir placas a 20 dólares e vendê-las a 40. O único lugar em que eu poderia vender era o clube. Pensei se haveria cinquenta pessoas ali dispostas a comprar meu computador em vez do Altair com um Intel. Achava que não, mas Steve disse que, mesmo que não conseguíssemos recuperar o investimento, ao menos teríamos uma empresa.