

Jornal do HP Club do Brasil

<http://hello.to/hpclub>
Edição no.3 - 13/05/1999

Linguagens de Programação da HP48

Muitas pessoas sempre perguntam: "o que é USER? O que é sysrpl? O que é ML?". Todas essas nomenclaturas são apenas linguagens de programação usadas na HP48.

O USER (ou USER-RPL) é a linguagem do usuário, ou seja, que qualquer usuário pode facilmente programar na HP48. Essa é a linguagem mais simples que não necessita de nenhuma compilação para se executar seus programas, podendo digita-los diretamente na HP sem a necessidade de nenhum programa específico.

O SYS (ou sysrpl) é uma linguagem onde muitos termos são parecidos com o USER, a diferença é que existem muito mais opções de comandos e funções. Essa linguagem, apesar de ser muito parecida com USER é mais complexa, já que para uma mesma função em USER, você possui dezenas de funções em sys, uma para cada tipo de objeto. Para se programar nessa linguagem é necessário um compilador que pode ser usado tanto no PC (HPIde, HPDev) ou na própria HP (Jazz). Todos comandos em USER são resultado de compilações de comandos em sysrpl. Nesse tipo de programação é necessário também um maior cuidado na programação já que caso utilize um objeto indevido na execução do programa pode ser causada uma perda de memória da HP.

O ML (machine language) é o Assembler da HP. A programação em assembler é a mais complexa já que em vez de utilizar funções você usa posições de memória para manipular seus objetos. Para programação em assembler é desejável (para não dizer necessário) que se conheça também o hardware da calculadora. Também é necessário que se utilize um compilador como os citados acima e o cuidado necessário ao se programar nessa linguagem é maior ainda.

Agora você pergunta: "o que significa o RPL?". RPL é apenas uma abreviação para Reverse Polish Language, ou linguagem polonesa reversa, que é apenas o modo da HP tratar dados, sendo necessário por exemplo, para um cálculo a entrada dos valores e depois a execução da função desejada.

Para facilitar a visualização e identificação dos tipos de linguagens mais comuns da HP48, a tabela abaixo mostra alguns exemplos de programas e funções.

	<i>USER-RPL</i>	<i>Sysrpl</i>	<i>ML</i>
Na HP48	« VARS SORT ORDER »	External	Code
Código Fonte	« VARS SORT ORDER »	:: GARBAGE MEM UNCOERCE %2 %/	CODE GOSBVL =SAVPTR LC(5) =CONTRAST DO=C C=DAT0 W C=C+1 W DAT0=C W GOSBVL =GETPTRLOOP ENDCODE
Função	Ordena as variáveis do DIR corrente alfabeticamente	Verifica memória disponível (comando MEM)	Aumenta o contraste do visor da HP48

Tacio

Você sabia...

- A HP já possui como funções inclusas o cálculo de combinações ou permutações. Para calcular é só entrar com o número de dados e o tamanho dos arranjos e ela retorna o número de combinações ou permutações possíveis. Essas funções estão em [MTH] [NXT] |PROB|.
- Para arredondar um valor na pilha operacional é só digitar o número de casas decimais desejadas e executar RND ([MTH] |REAL| [NXT][NXT])
- Para cancelar todos os alarmes pendentes é só clicar em [ON] e [4] simultaneamente.

COISAS QUE PODEMOS FAZER PARA ECONOMIZAR BATERIAS

Em época de crise mundial e escassez de recursos aqui vão algumas coisas que ajudam na drenagem mais rápida das baterias para ajudar você a ver onde pode economizá-las :

- Pressionar teclas : É claro que se você não pressionar umas teclas de vez em quando, você não vai conseguir fazer nada com sua calculadora, mas usando-as mais do que o mínimo necessário é um excelente sugador de baterias. Particularmente, a técnica de pressionar o ALPHA enquanto você digita um texto drena as baterias mais do que quando se pressiona duas vezes as mesmas teclas, travando no modo de texto. Se você digita três ou mais caracteres alfabéticos com certa periodicidade, você pode salvar a tecla de ser usada fazendo -60 SF, que causa uma única pressionada na tecla ALPHA para travar, ao invés das duas pressionadas consecutivas. Se você digita a maioria em letras minúsculas, pressionando <= ALPHA enquanto já em modo ALPHA produz letras minúsculas como padrão e maiúsculas somente quando se pressiona <=, que é o inverso do modo ALPHA inicial (isto só dura durante a sessão de edição corrente ou até <= é repetido).
- Deixando a porta serial aberta depois de transferir dados : Qualquer transferência de dados abre a porta serial, mas nada o fecha de volta exceto o comando CLOSEIO ou desligando a calculadora. Portanto, poderia prolongar a vida útil da bateria se tomássemos o hábito de desligar a HP depois de qualquer transferência manualmente inicializada, ou de incluir o comando CLOSEIO como parte de todo programa que usa a porta serial (e não esquecer de usar IFERR transferir dados THEN ... END CLOSEIO para que a porta serial seja fechada mesmo depois de um erro ou tecla CANCEL. Modo SERVER também abre a porta serial.
- "Beeping" : Se você nota mensagens de erro mesmo sem lembrete auditivo você pode desligar o beep padrão (-56 SF). Seus colegas de sala podem até lhe agradecer. Inclusive, um alarme comum daqueles de rádio-relógio é até melhor para lhe acordar de manhã antes de ir para aula !
- Ligando o relógio por muito tempo : Aquele relógio no menu de status faz um cálculo completo de data e rotina de display todo mísero segundo enquanto o relógio tá ligado; isto 3600 vezes a hora, ou vários minutos extra de programa rodando por hora !
- Deixando a calculadora ligada : Eu sei que ela desliga sozinha depois de 10 minutos, e ela só usa +/- 50% de força de quando está rodando um programa, mas isto significa que 10 minutos de "breve hibernação" é mais ou menos o mesmo que 5 minutos de um programa rodando. Se você sabe que parou de usar por agora, seria econômico desligar a HP, huh !?
- Jogando Doom e MK em todas as aulas : overloaD ... overLoaD ... oVerLoaD ... OverLoaD ... OverLoaD ... OverLoaD ... OverLoaD !! hehe

Parece coisa de turco ? Bem ...

Daniel Lopes - Dookie - danlop@brhs.com.br - ICQ#9259919

Para compactar programas, em qualquer linguagem, é muito simples.

Você deve pegar o pacote BZ series, na pagina do Tacio. Nesse pacote, você deve escolher um BZ de 1 ate 15. Quanto maior o numero mais rápida será a compactação, mas também será necessária muito mais memória, numa progressão geométrica. Eu aconselho até o BZ5.

Transfira ele para sua HP e mude o nome para BZ simplesmente. Este programa é o compactador/descompactador, e possui cerca de 1,5kb. Se você quer simplesmente compactar um arquivo, deixe o conteúdo do arquivo (não o nome) no nível 1 da pilha e acione este programa. Verifique se a compactação foi satisfatória através das informações no alto da tela, e então, se for o caso, salve o resultado por cima do arquivo antigo. Para descompactar: deixe o conteúdo do arquivo no nível 1 da pilha e acione o BZ novamente, e salve o arquivo ou execute o programa direto apertando EVAL.

Para criar programas que rodam diretamente sem precisar descompactar

Transfira para sua HP os programas BZM e UBZ. O UBZ tem 321 bytes, e serve apenas para descompactar. O BZM é o responsável por criar programas que rodam diretamente sem precisar de descompactação. Para funcionar ele precisa do BZ e do UBZ, e estes dois programas precisam estar com estes nomes, exatamente e em maiúsculas. Isso feito, deixe o conteúdo de um arquivo na pilha, e acione o BZM. Irá aparecer um menu em branco (invertido), escrito BZ, UBZ, duas teclas em branco, STR, PRG.

STR é para compactar qualquer arquivo, de modo que esse arquivo se descompacte sozinho e deixe seu conteúdo aberto na pilha, mas sem executar.

PRG faz isso e também roda o programa. E roda de forma tão rápida que você nem percebe o programa sendo descompactado.

Compensa conhecer os compactadores da linha BZ. Em uma GX, certa vez eu consegui carregar 240Kb de dados, usando os compactadores.

Qualquer duvida, escrevam.

Gustavo Domit
UIN 21354999
hp48club@hotmail.com

Cursos de manuseio e programação da HP48G/G+/GX

Domine todo o potencial de sua calculadora!
Aulas particulares ou turmas fechadas de no máximo 15 alunos!
Aproveitamento total!

Maiores informações <http://hello.to/hpclub> ou
(011) 55893540/91428837 com Tacio Philip

1. Prefácio

Desde que comprei a minha HP 48 G há cerca de 2 anos atrás, já sabia de algumas de suas incríveis possibilidades de uso. E, a cada dia que passava, eu ia descobrindo mais e mais coisas sobre ela. Em três semanas eu já fazia alguns programinhas em USER-RPL, e já tinha lido o manual dela praticamente inteiro. Um ano depois comecei a programar em SYS-RPL e praticamente desisti, por ser relativamente mais difícil que USER-RPL (na verdade, a maneira como programar é a mesma, mas em SYS-RPL você tem que ser um programador muito bom para fazer um programa totalmente seguro).

Mas, mesmo conhecendo sobre programação e desenvolvendo alguns programinhas para satisfazer minhas necessidades na escola, parece que faltava alguma coisa, a HP parecia estar isolada do mundo, se limitando a transferir arquivos para o computador, para outra HP e mais nada que isso. Já tinha visto alguns programinhas que imitavam controles remotos, outros que verificavam o "eco" do infravermelho... Mas a porta serial estava ali, seguia um padrão mundial (o RS-232C) e podia ser utilizada para a HP se comunicar com o mundo lá fora. Em um belo dia, peguei no site do hpcalc.org um arquivo onde Daniel Roggen explicava como ligar HPs em rede. Foi o passo inicial que culminou no desenvolvimento deste projeto. Durante um mês, passei a estudar as especificações da porta serial da HP, li e reli várias vezes o artigo do Daniel. Vocês, ao lerem o documento, vão notar uma forte ligação entre ele e este que vocês estão lendo. Vocês até poderão dizer: - Mas ele simplesmente traduziu o documento do Daniel!!!. Não vou negar, utilizei o documento como uma fonte de informações e, a maioria dos conceitos aqui utilizados foram retirados do documento, mas esse projeto não tem o objetivo de simplesmente ser uma tradução e, sim dar uma pequena contribuição sobre o assunto, explicando como realizar uma ligação em rede e utilizando um software relativamente simples para que pessoas possam trocar mensagens nesta pequena rede.

Espero que vocês gostem do resultado do trabalho.

2. Agradecimentos. Dedicatória.

Antes de tudo, gostaria de agradecer a compreensão dos meus pais, que ao me verem durante tanto tempo em frente ao computador, tiveram a paciência necessária para, como diz o Tacio do HP Club do Brasil, não me internarem em uma clínica para viciados em computadores.

Dedico esse trabalho à Patrícia, minha amiga; Alexandre, companheiro de projetos loucos de eletrônica; Michel, que muito me incentivou e até hoje me incentiva a aprender novas linguagens de programação e coisas novas sobre HP 48 G; Alberto; Gustavo Adolpho; e todos os meus outros amigos.

3. Introdução

Como discutido anteriormente, esse trabalho tem o objetivo de oferecer uma explicação de como fazer com que várias HPs sejam ligadas em rede para trocarem mensagens. Vou procurar explicar os seguintes assuntos: a porta serial da HP, o modelo de rede Token Ring e um software para fazer esse trabalho.

Devido ao tamanho desse artigo escrito pelo Francis tornou-se inviável coloca-lo na íntegra no Jornal do HP Club, estando este disponível para download na íntegra na página do HP Club do Brasil, menu Dicas.

Muitas vezes em programação é necessário a utilização de uma rotina cíclica para se obter o resultado desejado. Essa rotina pode ser executada até que ocorra um determinado evento ou ser executada um número pré definido de vezes.

Para facilitar a visualização vamos imaginar a seguinte hipótese:

- Você quer fazer um programa que calcule a somatória de uma função de n_0 até n , sendo colocado para isso na pilha operacional o valor de n_0 , n e a função.

Como você faria esse programa? Uma das respostas para essa pergunta seria:

SOMA1 (77.5 bytes checksum #8F48h)

```

«
→ f n0 n                               @salva como variáveis temporárias f, n0 e n
  « 0 n0 n                               @define n0 e n como início e fim do LOOP
    FOR i f X i 2 →LIST |                @coloca na pilha operacional a função e a
    +                                       @calcula para i somando a seguir
    NEXT                                    @termina o LOOP
  »
»

```

no lugar de ... X i 2 →LIST ... você poderia substituir por { X i } mas desse modo causaria erro quando a função a ser calculada fosse apenas 'X'.

Imagine agora que você quer executar uma função n vezes até retornar um valor maior ou igual a t . Como dado seria a função e o valor de t . Você poderia fazer para isso:

* r é definido como o resultado da função calculada e z um contador

SOMA2 (112.0 bytes checksum #2A13h)

```

« 0 0 → f t z r                         @define f t z e r como variáveis temporárias
  « DO f X z 2 →LIST |                    @calcula f(z)
    'r' STO 'z' INCR                       @salva resultado em r e incrementa z
    UNTIL 'r>t'                             @clausula de teste
    END DROP                                 @apaga último valor já que LOOP é executado
  »                                          @n+1 vezes
»

```

| é encontrado em [<-] SYMBOLIC [NXT]

Um outro modo seria:

SOMA3 (114.5 bytes checksum #93D9h)

```

« 0 0 → f t z r                         @define f t z e r como variáveis temporárias
  « WHILE 'r<t'
    REPEAT
      f X z 2 →LIST |                      @calcula f(z)
      'r' STO 'z' INCR                     @salva resultado em r e incrementa z
    END
  »
»

```

O funcionamento das funções exemplificadas pode ser descrito como:

```

« ... inicio fim FOR contador ... NEXT ... »
« ... DO ... UNTIL clausula_de_teste END ... »
« ... WHILE clausula_de_teste REPEAT ... END ... »

```

Mais exemplos de programas com clausulas cíclicas na próxima edição do
Jornal do HP Club

Algumas diferenças entre as calculadoras HP48

Sempre que alguém vai comprar uma calculadora HP e chega em uma loja e vê os muitos modelos existentes se pergunta: "Qual a diferença entre elas?".

Na tabela abaixo você vê algumas diferenças de hardware e operacionais entre as calculadoras da série 48.

	48S	48SX	48G	48G+	48GX
Memória RAM	32kb	32kb	32kb	128kb	128kb
Memória ROM	256kb	256kb	512kb	512kb	512kb
Expansão (entrada de cartões)	Não disponível	Disponível (2 portas até 288kb)	Não disponível	Não disponível	Disponível (2 portas até 4Mb)
Input forms (campos de entrada)	Não disponível	Não disponível	Disponível	Disponível	Disponível
Equation Library (biblioteca de equações)	Não disponível	Disponível com cartão aplicativo	Disponível	Disponível	Disponível
Cálculos com listas	Não disponível	Não disponível	Disponível	Disponível	Disponível
Transferência XMODEM	Não disponível	Não disponível	Disponível	Disponível	Disponível
Gráficos 3D	Não disponível	Não disponível	Disponível	Disponível	Disponível

Agora você pergunta: "O que significam essas diferenças?"

Memória RAM (Random access memory) - memória do usuário utilizada para armazenar programas, variáveis, arquivos etc.

Memória ROM (read only memory) - memória apenas de leitura, a partir disso percebe-se que para a série G deve existir muito mais funções que para série S.

Expansão - indica a possibilidade de expansão de memória a partir de cartões de memória. Note que a SX só aceita cartões de 128kb enquanto que uma das portas da GX aceita cartões de até 4mb!

Input Forms (campos de entrada) - são aqueles campos de entrada que aparecem quando você entra em alguma função como [->] PLOT, esse tipo de entrada só está disponível na série G.

Equation Library (biblioteca de equações) - uma biblioteca com dezenas de fórmulas já inclusas possibilitando a solução por qualquer variável.

Cálculos com listas - apenas na série G é possível executar uma função à vários objetos de uma vez utilizando listas. Na série S isso não é possível.

Transferência XMODEM - é um modo de transferência entre HP e PC (HP-HP possível com a utilização de um cabo de comunicação HP-HP) bem mais rápido que o Kermit (disponível nas séries S e G). Esse modo de transferência só foi disponibilizado na série G.

Gráficos 3D - a série G plota gráficos 3D - $f(x,y)$, opção não disponível na série S.

Além dessas existem ainda outras diferenças menos importantes que não foram listadas aqui.

Cabos Pag's - Cabos de comunicação para HP48

Sua interface com o PC

Conecte sua HP ao PC e aproveite todo o seu potencial!

Maiores informações **Cabos Pag's**

<http://www.abati.com.br/CabosPags/>

Quando você possui uma matriz e executa a função 1/x na 48 ela te retorna o inverso da matriz. Imagine agora que você esta apenas utilizando uma matriz para armazenar dados e quer calcular o inverso dos elementos da matriz, como você faria?

Para executar essa função um dos métodos é criar um programa que pegue os elementos da matriz, calcule seus inversos e os coloque de volta.

Abaixo segue um programa em sysrpl que executa essa função.

MATRIZ2SYS (52.5 bytes checksum #5282h)

```
::
CK1NOLASTWD      @checa se há um argumento na pilha operacional
CK&DISPATCH1    @checa o tipo de objeto
FOUR              @verifica se é uma matriz ou vetor (ARRAY)
::
DUP
ARSIZE           @tamanho da matriz (vetor) = número de elementos
#1+
ONE
DO              @início do LOOP
INDEX@          @contador i
DUP
ROT
SWAP
PULLREALEL      @retira da matriz o elemento i
%1/             @calcula o inverso
ROT
PUTREALEL       @coloca o inverso do elemento de volta na posição i
LOOP            @fim do LOOP
;
;
```

Um programa similar que executa a mesma função em USER seria:

MATRIZ2USER (89 bytes checksum #A171h)

```
<<
DUP SIZE OBJ→    @tamanho da matriz
DROP2 → n
  << 1 n
    FOR i 1 n      @inicia o LOOP
      FOR j i j 2 →LIST @ lista com as coordenadas dos elementos { i j }
        DUP2 GET    @retira o elemento da matriz
        INV PUT     @calcula o inverso e retorna o valor
      NEXT
    NEXT
  >>
>>
```

Para o calculo do inverso dos elementos de uma matriz [[1 2][3 4]] o programa em sysrpl levou 0,027s enquanto que o programa escrito em USER levou 0.39s, ou seja, o programa em sysrpl foi executado 14 vezes mais rápido!

Isso se deve ao fato de cada comando USER ser uma compilação de vários comandos sys, entre eles a checagem do tipo de argumento. Por esse motivo torna-se mais difícil a programação em sysrpl, onde o programador é responsável pela checagem de argumentos para evitar erros em seu programa.

Trena Eletrônica

Caros amigos do HP Club,

Tenho lido algumas mensagens nesta lista sobre a possibilidade de fazer da HP uma espécie de trena eletrônica. A idéia não é má mas acho existem uma série de fatores que contribuem para que isto não aconteça na HP48:

1° - O alcance do Infra Vermelho da HP é muito pequeno. Sei que existe maneiras de se fazer com que o alcance dele aumente, mas acho que isso iria consumir muita bateria e poderia até mesmo danificar os circuitos internos da HP, que foram projetados para um valor pré-definido. Trocar um resistor poderia fazer com que a vida útil de todo o circuito de transmissão de Infra Vermelho diminuísse consideravelmente.

2° - O Luís Daniel postou uma mensagem dizendo que esse tipo de medição é feito por Ultra Som. E ele está certo, é mais fácil fazer uma trena eletrônica usando Ultra Som, devido ao seguinte fator: a velocidade do som é cerca de 881 mil vezes menor que a da luz (o infravermelho, no caso da trena de infravermelho). A medição utilizando ultra som é feita da seguinte forma: manda-se um sinal de som em uma frequência inaudível ao ouvido humano, e espera que esse sinal retorne. Medindo-se o tempo entre transmissão e recepção e conhecendo a velocidade que o som se propaga no ar, pode-se saber a distância de ida e vinda que o sinal percorreu. Por exemplo, se quiséssemos medir um objeto a 1 metro de distância da trena eletrônica utilizando ultra som, o tempo seria de ida e volta seria de aproximadamente 5.88 milisegundos. Mas se utilizássemos um raio de infravermelho, esse tempo seria de aproximadamente 6.66 nanosegundos, que é um tempo mais difícil de se medir. Principalmente na HP 48, que funciona a 4MHz pois, mesmo que se uma instrução do processador fosse executada em 1 pulso de clock (a maioria dos processadores gasta bem mais do que isso), essa seria executada em 250 nanosegundos, ou seja, o infravermelho seria enviado e a HP não teria tempo de saber que o raio voltou.

3° - Está certo, existem outras técnicas de se medir por infravermelho, diferentes da técnica explicada acima. Mas isso iria requerer hardware extra, e isso é inviável.

Obrigado a todos pela atenção.

Francis M. de P. Mendes

Técnico em Eletrônica

URL: <http://www.geocities.com/SiliconValley/Peaks/9628>

UIN: 28654769

Alguns motivos para utilizar o emulador da HP48

1. Testar rapidamente programas pegos na Internet sem necessidade de transferência para uma HP real;
2. Fazer programas sem o perigo de perder a memória da sua HP;
3. Ter a possibilidade de possuir dezenas de calculadoras com dezenas de cartões de memória no seu PC;
4. Fazer cálculos no computador sem a necessidade de se pegar uma HP real, utilizando uma calculadora com a qual você já esta bem familiarizado no próprio PC;
5. Ficar jogando MINEHUNT sem se preocupar com o consumo de baterias;
6. Fazer com que os outros tenham certeza de que você é um Hpmaníaco.

Tacio