

PalmOS Pascal Compiler

Date de publication : 05/05/2008 , Date de mise à jour : 05/05/2008

Par **Thireus**

PalmOS Pascal Compiler (PP Compiler) : présentation et comment débiter

Généralités

Editeur et exemples

Astuces

Avis personnel et conclusion

Généralités

PP est un compilateur gratuit pour PalmOS, il permet de générer des exécutables .PRC (en mode 68000 et ARM) ou des applications en mode console, à partir du langage Pascal standard.

Vous pourrez aussi créer avec ce compilateur des shared libraries, DA et autres armlets... directement sur votre Palm ! Idéal pour programmer lorsque vous n'avez pas d'ordinateur sous la main !

PP a été créé par Philippe Guillot, vous trouverez la dernière version sur le site suivant : <http://www.ppcompiler.org/>.

Editeur et exemples

Je conseille à tous ceux qui souhaitent programmer sur Palm d'utiliser le freeware **PIAF** qui est un éditeur pour PP mais aussi pour iziBasic. Outre la coloration syntaxique, il possède de nombreuses autres fonctionnalités, notamment celle de compiler directement son code source, pour permettre sur une seule interface de coder et compiler !

Vous trouverez cet éditeur sur le site de PP à l'adresse : <http://www.ppcompiler.org/download.php>.

Vous trouverez aussi sur cette page tous les fichiers pour débiter dans le monde de PP ! A savoir API, exemples, tutoriaux, librairies etc.

Voici un exemple de ma création qui fut mon premier programme avec PP compiler :

```
{ $appl Thir }
Program Thireus (cmd); // Nom du programme

{ $i PalmAPI2.pas } // Déclaration du fichier API
{ $i PPlib.pas } // Une petite librairie bien utile

Procedure Write (S: String; X, Y: Integer);
Var
  N: Integer;
Begin
  N := Length(S);
  WinDrawChars (S, N, X, Y);
End;

Procedure DrawCell (u, v, X, Y: Integer);
Var
  Rec: RectangleType;
Begin
  With Rec do begin
    topLeft.x := X;
    topLeft.y := Y;
    extent.x := u;
    extent.y := v;
    WinDrawRectangle (Rec, 0);
  end;
End;

Var
  Cmd: UInt16;
  Wrt, oldScale: Integer;

Procedure Go;
Label
  1;
Var
  i, x, y: integer;
  x_n, x_nn, y_n, a_0, b_0: real;
  a: Boolean;
  w, wt, depl, dep2: UInt32;
  mode: WinScreenModeOperation;
Begin
  a := true;
  w := 320;
  wt := 480;
```

```

dep1 := 8;
dep2 := 16;
mode := winScreenModeSet;
WinScreenMode(mode,w,wt,dep1,a); // Petite astuce pour passer en Hires+
WinScreenMode(mode,w,wt,dep2,a); // Vous avez une autre idée vous ?

WinSetBackColor(255);
WinSetForeColor(255);
WinSetTextColor(0);

WinPushDrawState;
oldScale := WinSetCoordinateSystem(0);

for x := 0 to 320 do begin
  b_0 := (x-160)/150;
  for y:=0 to 480 do begin
    a_0 := -(y-240+100)/150;
    i:=0;

    x_n := 0;
    y_n := 0;

    while i<100 do begin
      x_nn := x_n;
      x_n := x_n*x_n - y_n*y_n + a_0;
      y_n := 2*x_nn*y_n + b_0;
      if (x_n*x_n + y_n*y_n) > 4 then begin
        WinSetForeColor(100-i);
        DrawCell(1,1,x,y);
        i := 100;
      end;
      i:=i+1;
    end;
  end;
end;

write('Currently running on column '+intToString(x)+' ',0,0);

if KeyCurrentState<>0 then begin
  goto 1;
end;

end;

// WinDrawLine(0,0,160,225);
1:
WinPopDrawState;
End;

Begin // Programme principal
Go;

while KeyCurrentState = 0 do;

// if cmd = sysAppLaunchCmdNormalLaunch then begin
// Wrt := SysTaskDelay(10*SysTicksPerSecond);
// end;

End.

```

Il s'agit de la fractale Mandelbrot (pour ceux qui font des Maths parmi vous...).

Vous verrez de vous-même que les API fournies sont très limitées et qu'il faudra créer les vôtres si vous voulez faire de jolis programmes.

On passe donc à la partie astuces qui vous sera bien utile, vous verrez.

Astuces

Pour étendre les possibilités de l'API fournie par défaut à savoir "PalmAPI2.pas", il existe un site traitant de "Palm OS® Programmer's API Reference" : <http://www.access-company.com/developers/documents/docs/palmos/PalmOSReference/ReferenceIX.html>.

Allez voir dans le fichier "PalmAPI2.pas" et vous aurez une idée de la syntaxe à utiliser pour créer vos nouvelles procédures ou fonctions.

Par exemple :

"PalmAPI2.pas" contient la procédure suivante :

```

procedure WinDrawLine (x1,y1,x2,y2:Coord); inline (SYSTRAP,$A213);

```

"Palm OS® Programmer's API Reference" vous renseigne :

WinDrawLine Function
<p>Purpose Draw a line in the draw window using the current foreground color.</p>
<p>Declared In Window.h</p>
<p>Prototype void WinDrawLine (Coord x1, Coord y1, Coord x2, Coord y2)</p>
<p>Parameters x1 : x coordinate of line start point. y1 : y coordinate of line start point. x2 : x coordinate of line endpoint. y2 : y coordinate of line endpoint.</p>

Returns

Returns nothing.

Comments

This function differs from WinPaintLine() in that it always uses winPaint mode (see WinDrawOperation). WinPaintLine uses the current drawing state transfer mode instead of winPaint.

See Also

WinDrawGrayLine(), WinEraseLine(), WinFillLine(), WinInvertLine(), WinPaintLine(), WinPaintLines()

Il vous suffit donc juste de prendre connaissance de la partie *Prototype* pour les paramètres d'entrée. Puis il vous faudra le systrap ! (**inline(SYSTRAP,\$A213);**)

Pour prendre connaissance des systrap à utiliser pour que les fonctions marchent, il vous suffit d'aller sur ce site : <http://nsbasic.com/palm/info/technotes/TN06.htm>.

Vous trouverez par exemple la ligne suivante :

WinDrawLine 531

Il vous suffira alors de convertir le nombre décimal en hexadécimal : 531 (décimal) -> 213 (hexadécimal) et de rajouter la lettre A au début, car l'hexadécimal fonctionne par couple, ce qui nous donne : **A213**. Ce qui nous donne bien le **inline(SYSTRAP,\$A213);** (Un autre exemple : 12 (décimal) -> C (hexadécimal) -> A00C (systrap)).

On vient de recréer la procédure :

```
procedureWinDrawLine(x1,y1,x2,y2:Coord); inline(SYSTRAP,$A213);
```

Avis personnel et conclusion

Tout d'abord, le fait d'avoir un compilateur de poche c'est vraiment très pratique !

Je tiens aussi à dire que personne ne m'a appris à me servir de PP, j'ai appris le Pascal directement avec PP en lisant le peu d'exemples qui sont fournis avec le compilateur. Pour ce qui est de la partie astuces, je l'ai moi-même trouvée à cause d'une programmation intensive avec PP, et dû au manque de fonctions.

Je trouve qu'il est dommage qu'il n'y ait pas plus de développeurs PP.

Je compte donc sur vous pour commencer dès maintenant à programmer sur Palm avec PP ! Maintenant que vous avez un mini tutoriel sous la main, foncez ! Vous ne serez pas déçu.