The Tutorial Series

This is the fourth issue of a series of tutorials for the HP Prime, written by Edward Shore. This session will cover CHOOSE and CASE. If you have programmed with the HP 39g, 39g or 39gII, you will recognize the programming as the HP Prime programming language (HPPP) is similar. We are using the latest firmware in this series, available on the website.

How to start?

1. Press Shift + 1 (Program).
2. Press New. It is the second touch key.
3. Enter the name of the program. Pressing the ALPHA key twice will turn on UPPERCASE Α. Pressing ALPHA, Shift, ALPHA will turn on lowercase alpha-lock. To exit any lock, press the ALPHA key one more time. When you’re happy with the name, press Enter.

Rules for Program Names:

1. Letters, numbers, and the underscore character (_) only.
2. The program name must start with a letter.

Structure of a HP Prime Program

A HPPP program is encased of an EXPORT - BEGIN - END structure. The layout is generally like this:

```
EXPORT program_name(arguments)
BEGIN
commands and comments go here
END;
```

Each line containing a command generally must end with a semicolon (;). A semicolon can by type by pressing ALPHA then the Plus key (+).

Comments can be typed. They are designated by two forward slashes. The slashes are typed by pressing the Divide key ( ÷ ). Anything in the line following the two slashes is ignored in running the program.
**CHOOSE and CASE**

**CHOOSE:** Creates a pop up choose box, similar to what you see when you click on a soft menu. There are two syntaxes for CHOOSE:

Simple Syntax (up to 14 options):
```
CHOOSE(var, "title string", "item 1", "item 2", ..., "item n");
```

List syntax (infinite amount of items):
```
CHOOSE(var, "title string", ["item 1", "item 2"]);
```

Choosing item 1 assigns the value of 1 to var, choosing item 2 assigns the value of 2 to var.

Access: Cmds, 6. I/O, 1. CHOOSE

**CASE:** Allows for different test cases for one variable. Also includes a default scenario (optional).

```
CASE IF test 1 THEN do if true END;
IF test 2 THEN do if true END;
...
DEFAULT commands END;
```

Access: Cmds, 2. Branch, 3. CASE

Let's look at two programs to demonstrate both CHOOSE and CASE.

**TERMVEL - Terminal Velocity of an Object**

```
EXPORT TERMVEL()
BEGIN
LOCAL L0:={9.80665,32.174},
L1:={1.225,.0765},
L2:={.47,1.05,1.15,.04},C,K,M,A,T;
CHOOSE(C,"Units","SI","English");
CHOOSE(K,"Type of Object","Sphere","Cube","Cylinder","Tear-Shaped");
INPUT({M,A},"Object",
{"M=","A="},{"Mass","Surface Area"});
T:=√((2*M*L0(C))/(L1(C)*A*L2(K)));
MSGBOX("Terminal Velocity="+T);
RETURN T;
END;
```

Examples:

- Sphere, SI Units, M = 0.05 kg, A = 0.0028 m\(^2\)
  Terminal Velocity: T = 24.66 m/s

- Cube, US Units, M = 1.2 lb, A = 0.3403 ft\(^2\)
  Terminal Velocity: T = 53.14 ft/s

**TIP:** Use the IF THEN ELSE structure with INPUT to execute a set of default instructions if the user presses cancel. INPUT returns a value of 0 if ESC or cancel is pressed, and 1 if a value is entered.

```
IF INPUT(...) THEN
commands if values are entered
ELSE
commands if Cancel is pressed
END;
```

Default values can be assigned to values as an optional fifth argument for INPUT.

```
INPUT(var, "Title", "Prompt", "Help", default value)
```

The type of variable maybe set to other than real numbers. Just remember to store such type before the INPUT command. For example, if you want var to be a string, store an empty string:

```
var:=" ";
```
AREAC - Area of Circles, Rings, and Sectors

EXPORT AREAC()
BEGIN
LOCAL C,R,S,θ,A;

INPUT(R, "Input Radius", "R =");

CASE
IF C==1 THEN A:=π*R^2; END;

IF C==2 THEN
INPUT(S,"Small Radius","r=");
A:=π*(R^2-S^2);
END;

IF C==3
INPUT(θ,"Angle","θ=");
\ Assume you are in the correct angle mode
IF HAngle==1 THEN
\ Test Angle Mode
θ:=θ*π/180;
END;
A:=θ*R^2/2;
END;

MSGBOX("Area is "+A);
RETURN A;
END;

Examples:
R = 2.5, r = 1.5, θ = π/4 radians or 45°

Circle: 19.6349540849
Ring: 12.5663706144
Sector: 2.45436926062