



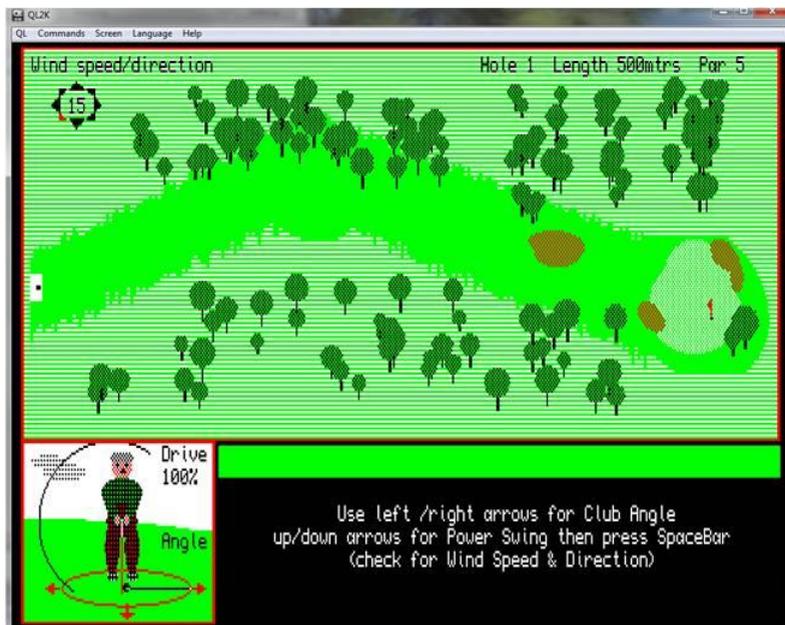
Sinclair QL retro-gaming



Sinclair QL retro-gaming



QBITS Golf 2016



QBITS Golf 2016

Returning to The Sinclair QL

As an owner of a QL I guess I'm somewhat biased. Due to my own background involvement to my mind the QL makes an excellent choice for those wishing to engage in a bit of computer programming and dabbling in early gaming software. This hankering for such I'm reliably informed by some is called retro-computing.

A first in what was to follow in Personal Computing the QL computer produced in 1983/4 came with a multi-tasking operating system called QDOS and a SuperBASIC interpreter for users to write their own programs. Bundled with the QL were four business applications; Quill a word processor, Abacus a spreadsheet, Archive an intelligent database and Easel a graphic package for charts etc.

Sadly my own QL microdrives and add-ons, floppy drives, extended memory etc. reside in the loft, not used since the TV/Monitor bought in the late eighties gave up the ghost. After a long gap, the last time I had the QL working was over ten years ago, the floppy drives were on their last legs and most of the floppies I had were unreadable.

However, searching online I came across the QL emulators and today I use copies of **QL2K emulator** loaded on my Win 7 desktop and Win 10 Tablet.

The QL user's manual

This original documentation as an introduction to the Sinclair QL, its operating system, programming in the SuperBASIC language, concepts etc. it is still to my mind a worthwhile read.

QL Emulators

There are several available for the original QL as well as its later spin offs. You can download these and run them on PC's, Desktops Laptops and Tablets under the Windows, Mac or Linux operating systems. Then there are the additional ROM's and toolkit extensions and an extensive number of useful programs all with plenty of helpful documentation available.

Check out Dilwyn's web site below for downloads, helpful information and links to other suppliers of QL software and documentation.

<http://www.dilwyn.me.uk/>

Updated: 07.09.2015

QBITS Golf 2016

QBITS

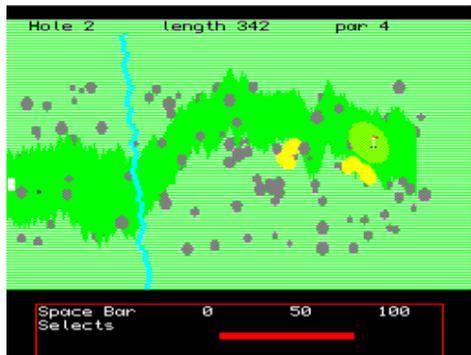
As a QL tinkerer with limited programming skills, back in the nineteen eighties I produced a few games for the family written in SuperBASIC. Some of which were purchased by other QL users at the time under the banner of QBITS software. In the eighties I was introduced to Forth and then Basic on a BBC micro and later tried a short course in C++ but never pursued my curiosity in programming. However, online visits to the QL scene over the past few years has rekindled the interest in exploring the QL and its SuperBASIC

Stored in the loft with my QL hardware and its documentation I came across notes on my games programs. With time on my hands in reading my notes I now find myself on something of a mission to completing the aspiration of those early attempts.

QL Golf Game beginnings

In the QL User magazine - May 1985 - R Shergold & D Tose provided a 9 hole Golf simulation game written in SuperBASIC. It packed in a lot of detail, creating different Fairways with all the usual hazards of trees, bunkers, streams, lakes. The spacebar was used to toggle the power for each club stroke and a clock style entry used for direction. The original Golf Game had a big appeal in my family.

Here's the main screen layout showing the hole number, length and par at the top.



The power strength of the club swing selected using the spacebar.

The angle of direction using clock hours and minutes. "3.0" the ball goes to the right.



The putting phase upon reaching the Green.



Owing much to the inspiration that came from this earlier golf game, at the time it seems I had several idea's for changes to its graphical display...

QBITS Golf 2016

QBITS Golf

This began with the decision to create an 18 Hole Golf course with Fairways of varying lengths and difficulty and an enlargement of the Green for a putting phase. A Golfer swinging his club up and down an Arc to depict the Power Drive. The Angle of direction to be a line moving around a circle/ellipse at the Golfer's feet. The normal hazards, trees, bunkers stream, lake and rough ground. Yet another was to include cross winds and add some kind of indicator showing the wind speed and direction blowing across the Fairway.

At the end of each hole played, a scorecard to show the player results to include other opponent players. The decision was made to run this in QDOS Mode 4 yet still maintain a good graphic representation. In addition the possibility of saving the game after so many holes, returning later to complete the course.

The Welcome Screen

Here a choice was made to add some scenic graphics, create a background for the introduction to the QBITS Golf game. Then options for starting a **NEW** game, **LOAD** a previously saved one or simply **Quit!**

The Scorecard

Extending mostly printed lines with a few variables tacked on, was a display with columns for hole 1-9 on the left and hole 10-18 on the right both with headings of Hole number, Length in metres and Par, then the number of Shots taken by the player and their opponents. The shots for each hole tallied up and compared with the course par to give an overall score rating as you play through the 18 holes. Option to **Continue** with the present Game, **SAVE** or **Quit!**

Score ratings

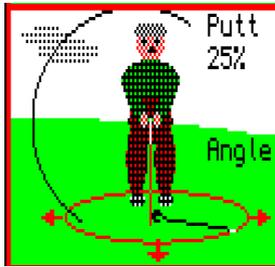
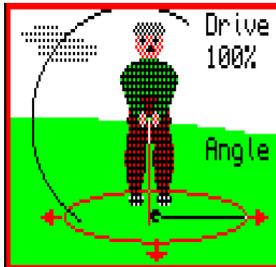
Shots =1	“Hole in One! - Awesome!”
Three under par	“An Albatross... First Class!”
Two under par -	“An Eagle... Excellent!”
One under par	“A Birdie... Well played!”
Shots equal to par	“A Par - not bad!”
Shots greater than par	“Over par on this hole.”
Score	“Out of shots” limit number of shots

If **LOADing** a game that was previously **SAVEd** then the scorecard to display the number of shots taken against previously played holes and the overall par position for the player and opponent players.

QBITS Golf 2016

Club Power, Angle

Another QBITS Game had a character seated on forklift truck so in creating the Golfer wasn't too much of a leap into the unknown. The Golfer and Angle graphics are shown below generated using the QL Graphics coordinates system. The height of swing determines the power, for the Fairway a full swing and for the Green a half swing to represent a putting stroke.



Power % of club reduced due to a hazard
50% in the Rough
25% in a Bunker



Wind Speed/Direction

Added to the Fairway was a Wind speed/direction indicator. This was based on a simple eight point compass design. The direction of wind selected drawn in a red triangle the others in black with the speed printed in the centre.

Club variables

These define the range of Club Swing for Power and the Angle of direction either in Drive (Fairway) or Putt (Green) mode and are dependent on Fairway length, par, wind and any hazards encountered. Plus a tally of shots taken.

Fairway or Green	putt	Fairway=0, Green=1, Continue Putting=2
Club type	club\$	club\$='Drive' (putt=0) club\$='Putt' (putt>0)
Club Max Power	club	Fairway=200 Green=80
Club Power	pow	0 to club
Power reduction	rmax	/2 normal, /4 rough (50%), /8 Bunker (25%)
Angle	ang	0-360 (in 10 degree steps)
Fairway length	flgth	between 180 and 525 mtrs
Fairway par	par	par 3 <270 par 4 >270<430 par 5>430
Number of shots	shots	0-9 shots taken at each hole (set to a limit)

Thoughts on Calculating Golf Ball movement

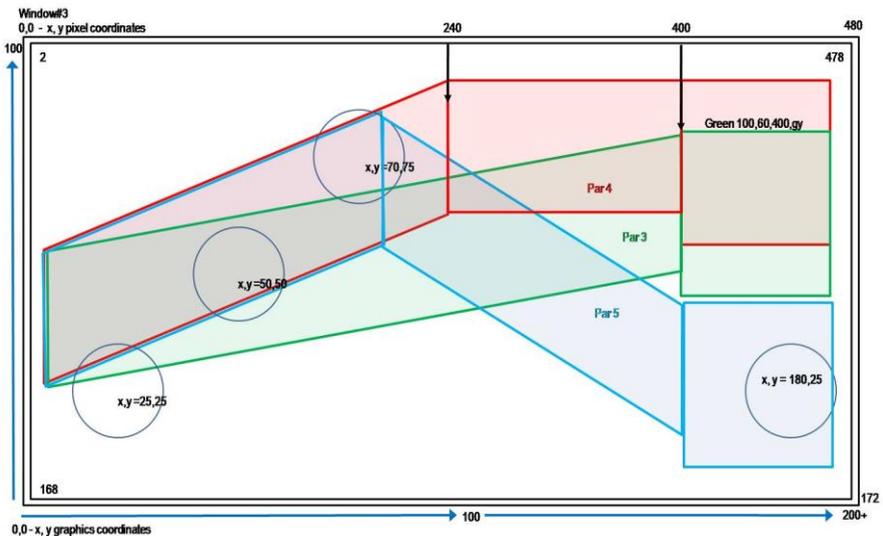
This would involve Power of the Club Stroke and Angle of direction in line with the Fairway length, Par, and values for the Wind strength/direction and any other limitations caused by hazards encountered.

QBITS Golf 2016

The Fairway & Green

First thing the Fairway and Green presented a bit of a challenge. It was necessary for each Fairway to be represented with differing difficulties, taking into account course par and distance. This turned out to be a little more problematic than originally expected and required playing around with a number of randomised variables.

The QDOS SuperBASIC command BLOCK was used to draw the fairway as a series of narrow width green coloured rectangles progressing across the screen.



Block #channel, width, height, x, y, colour

As the x coordinate increased, the y increases or decreases to shape the Fairway dependant on its par and length:- **Par=3** acts as gradual ascend to the Green, **Par=4** ascends then levels off to the Green, **Par=5** ascends then descends to Green. The aim to represent the varying difficulties of a typical golf course.

For the Green at the end of the Fairway, the Window is resized see right hand smaller rectangle in drawing above. The Green, any bunkers and trees are all drawn using the Graphics coordinate system.

As to the Green enlargement with the graphics coordinates system this was just a change of window size and redrawing from the initialised set of randomly generated and stored data.

QBITS Golf 2016

The hazards

Following the original Golf code hitting a Tree resets the ball address $bx=bx-2$, $by=by+RND(-1 \text{ TO } 1)$. The new position is tested and continues to be tested until clear or encounters a different hazard. Landing in a Bunker, Club Power is reduced to 25% on the next shot, likewise landing in the Rough, Club power is reduced to 50%. Landing in water (stream or lake) the ball is returned to the Tee. Landing on or beyond the outer boundaries (Fairway or Green) will incur a penalty shot.

Branches	“You have hit a Tree!”	
Bunker	“BUNKERED!”	max=8 (25%)
Rough	“You are in the Rough!”	max=4 (50%)
Water	“WATER!!! Back to Tee.!”	
Lost	“Out of Bounds - penalty shot.”	shots = shots +1

Colour codes and patterns:

Fairway	4	green
Green	4,7,3	green/white/stipple 3
Branches (Trees)	4,0,3	green/black/stipple 3
Bunker	2,4,3	red/green/ stipple 3
Rough	4,7,0	green/white/stipple 0
Stream/Lake (Water)	7	white

QBITS Golf screen's

QL Screens mode 4 (x axis 0 to 511 - y axis 0 to 255) 511

The screenshot displays the QBITS Golf game interface with the following windows and elements:

- Window#0 (command window):** Located at the bottom, it contains instructions: "For club use left /right arrows for Angle up/dpwn arrows for power then press SpaceBar (check for Wind Speed & Direction)".
- Window#4:** Located at the bottom left, it shows a golfer character with a "Drive 100%" indicator and an "Angle" label.
- Window#5 (provides information on state of play):** A large green area at the bottom right.
- Wind speed/direction:** A circular gauge in the top left showing a wind speed of 15.
- Window#3 Fairway and Green:** The main playing area, showing a golf course with a stream, lake, and various hazards. It includes a "Fairway" section with a "15" in a circle and a "Green" section with a "100" in a circle.
- Window#3 Green Enlarged:** A detailed view of the green area, showing "Ball position: bx=128+INT((bx-200)/2.4), by=16-INT((by-97)/2.6)".
- Window#3 Green:** A detailed view of the green area, showing "Screen RAM: Hole no: 180, hy=106" and "Graphics Scale 100.0.0".
- Colour codes:** A list of codes for different elements: Fairway (4), Green (4,7,3), Rough (4,7,0), Branches (4,0,3), Bunker (2,4,3), and Water (7).

QBITS Golf 2016

Initialising the Course

For each hole the Window#3 background colour is used to represent the rough. The Fairway a solid green strip progressing across the screen from the Tee to the Green. Along the Fairway hazards in the form of a bunker, a stream or a lake are randomly positioned before the Green. The course is then overlaid with a random number of trees. Finally the Green itself is drawn with its own bunkers and trees.

For the Fairway the SuperBASIC keyword BLOCK using the QL windows x, y coordinate system is used. For the Trees, Bunkers, Stream or Lake, and the Green, the CIRCLE and LINE of the Graphics x, y coordinate system are used. A third system of coordination is used for positioning the golf ball, this is carried out using the PEEK & POKE commands to directly access the QL screen RAM.

Identifying Location

Boundaries:	zx, mx, zy, smy	(z) zero position – (m) max position
Green:	gx, gy	SCALE 100,0,0 Graphic coordinates
Hole:	hx, hy	Different for Fairway Green or Enlarged Green
Ball:	bx, by	Anywhere within Fairway or Green boundaries

The location coordinates are worked out within a 256 by 256 grid. They provide the link to a specific point within the QL screen that identifies the corresponding bite within the QL screen RAM. The Fairway and Green boundary coordinates are positioned within the area of their overlaying window as is the hole position and any movements shown of the golf ball in play. The ball locations in terms of screen RAM are calculated both in relation to the size and positioning of window#3 for the Fairway and then when resized for the enlarged Green.

Interpreting Screen RAM Word/Byte/bit

The screen RAM is organised in a series of 16bit words starting at address Hex 20000 (131072 Dec) progressing across the screen in rows of 128 Bytes then down the screen in 256 lines. For mode 4 the high Byte is set 1 for green and the low bite set 1 for Red. Both bites set with 1's produce White when both are set as 0, Black. A 2x2 pixel colour fill can be interpreted with primary colour, contrast colour and stipple allocation (see over the page).

The Golf Ball position and relevant bytes are copied and held by variables colour and colour1. The ball position is written to the Screen RAM as newcol and newcol1. The pixels from colour and colour1 are then checked against colours and stipple patterns to determine if any hazards have been encountered. They are used to restore the original pixel colours when the ball is moved on.

QBITS Golf 2016

Working the QL Screen RAM

Calculations for bx, by locations of screen RAM address

$$bx = \text{INT}((bx + wdx + \text{COS}(\text{ang1}) * 525 * (\text{pow}/\text{dist})))$$

$$by = \text{INT}((by + wdy - \text{SIN}(\text{ang1}) * 525 * (\text{pow}/\text{dist})))$$

131072 base address of screen RAM

$$\text{add} = \text{INT}(131072 + 128 * by + \text{INT}(bx/4) * 2)$$

To Obtain and Restore original colours

$$\text{colour} = \text{PEEK_W}(\text{add}); \text{colour1} = \text{PEEK_W}(\text{add} + 128)$$

$$\text{POKE_W}(\text{add}, \text{colour}); \text{POKE_W}(\text{add} + 128, \text{colour1})$$

Workings for reading Byte/bits

$$\text{bit} = (bx \text{ MOD } 4) * 2$$

$$(bx \text{ MOD } 4) = (0 \text{ or } 1 \text{ or } 2 \text{ or } 3) * 2 = 0, 2, 4, \text{ or } 6$$

Ball bit location

$$\text{bitt} = \text{INT}(255 - 2^{(7-\text{bit})} * 1.87)$$

$$\text{INT}(255 - 2^{(7-0)} * 1.87) = 15$$

$$2) = 195$$

$$4) = 240$$

$$6) = 251.25$$

Binarycode

0000 1111

1100 0011

1111 0000

1111 1011

$$\text{IF } \text{bitt} - \text{INT}(\text{bitt}) \text{ THEN } \text{bitt} = \text{INT}(\text{bitt} * 1.018)$$

$$6) = 251.25 * 1.018 = 255$$

1111 1111

$$= 251.25 * 1.0189 = 256 \quad 1 \text{ 0000 } 0000$$

Ball colour black (block width=3, height=2)

$$\text{newcol} = (\text{bitt} \&\& \text{PEEK}(\text{add})) * 256 + (\text{bitt} \&\& \text{PEEK}(\text{add} + 1))$$

$$\text{newcoll} = (\text{bitt} \&\& \text{PEEK}(\text{add} + 128)) * 256 + (\text{bitt} \&\& \text{PEEK}(\text{add} + 129))$$

Colour Test (still not sure why these calculations appear to work)

Screen Ram 16bit word High Byte Low Byte

$$\text{bitt} = 2^{(7-\text{bit})}$$

$$2^{(7-0)} = 128$$

$$2) = 32$$

$$4) = 8$$

$$6) = 2$$

Green bit

1000 0000

0010 0000

0000 1000

0000 0010

$$\text{gb} = 4 * ((\text{bitt} \&\& \text{PEEK}(\text{add})) <> 0)$$

128*4 10 0000 0000

32*4 1000 0000

8*4 0010 0000

2*4 0000 1000

$$\text{bitt} = 2^{(7-\text{bit})} * 1.5$$

$$2^{(7-0)} * 1.5 = 192$$

$$2) = 48$$

$$4) = 12$$

$$6) = 3$$

Red bit

1100 0000

0011 0000

0000 1100

0000 0011

$$\text{rb} = (\text{bitt} \&\& \text{PEEK}(\text{add} + 1)) / \text{bitt} * 3$$

192/192*3 0000 0011

48/48*3 0000 0011

12/12*3 0000 0011

3/3*3 0000 0011

gb + rb colours (0-7)

Hazards check

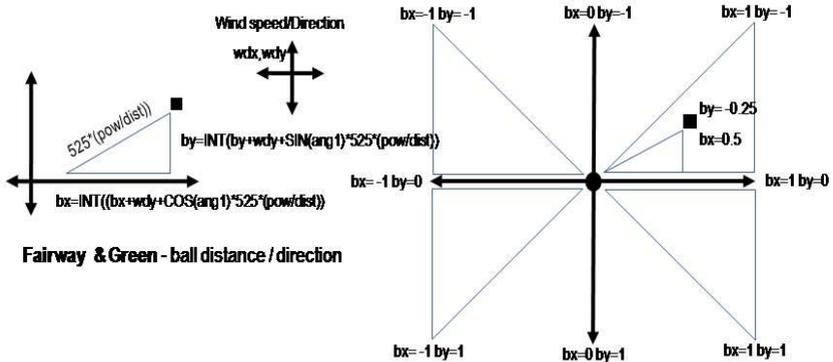
$$\text{IF } c1=4 \text{ AND } c3=0 \text{ OR } c1=0 \text{ AND } c3=4 \text{ :branches}$$

$$\text{IF } c1=2 \text{ AND } c3=5 \text{ OR } c1=5 \text{ AND } c4=2 \text{ :bunker}$$

$$\text{IF } c1=7 \text{ AND } c3=4 \text{ OR } c1=4 \text{ AND } c4=7 \text{ :rough}$$

$$\text{IF } c1=7 \text{ AND } c3=7 \text{ OR } c1=7 \text{ AND } c4=7 \text{ :water}$$

QBITS Golf 2016



Calculating screen position of ball: bx, by using triangulating coordinates.

Interpreting colour codes and patterns checking for hazards.

Function	GetCol(bxby,)	Gb Rb bit	Ball 3x2 pixels	Stipple	Contrast XOR main(mix)	Colour																	
C3	<table border="1"><tr><td>gn</td><td>rb</td></tr></table>	gn	rb	00 Black	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■	Bit 7 6	5 4 3 2 1 0													
gn	rb																						
■	■	■																					
C1	<table border="1"><tr><td>gn</td><td>rb</td><td>gn</td><td>rb</td></tr></table>	gn	rb	gn	rb	01 Red	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■	<table border="1"><tr><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■	■	■	■	■	■	■	■		
gn	rb	gn	rb																				
■	■	■																					
■	■	■	■	■	■	■	■	■	■														
C2	<table border="1"><tr><td>gn</td><td>rb</td></tr></table>	gn	rb	10 Green	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■	<table border="1"><tr><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■	■	■	■	■	■	■	■				
gn	rb																						
■	■	■																					
■	■	■	■	■	■	■	■	■	■														
C4	<table border="1"><tr><td>gn</td><td>rb</td></tr></table>	gn	rb	11 white	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■	<table border="1"><tr><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■	■	■	■	■	■	■	■				
gn	rb																						
■	■	■																					
■	■	■	■	■	■	■	■	■	■														

Procedure Test

Function	GetCol(bxby,)	Gb Rb bit	Ball 3x2 pixels	Stipple	Colour								
Fairway G	<table border="1"><tr><td>1</td><td>0</td><td>c3-4</td></tr></table>	1	0	c3-4	00 Black	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	0	c3-4											
■	■	■											
	<table border="1"><tr><td>1</td><td>0</td><td>1</td><td>0</td><td>c1-4 c2-4</td></tr></table>	1	0	1	0	c1-4 c2-4	01 Red	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■		
1	0	1	0	c1-4 c2-4									
■	■	■											
	<table border="1"><tr><td>1</td><td>0</td><td>c4-4</td></tr></table>	1	0	c4-4	10 Green	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	0	c4-4											
■	■	■											
Green G/W/3	<table border="1"><tr><td>1</td><td>1</td><td>c3-6/7</td></tr></table>	1	1	c3-6/7	10 Green	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	1	c3-6/7											
■	■	■											
	<table border="1"><tr><td>1</td><td>0</td><td>1</td><td>1</td><td>c1-4/5 c2-6/7</td></tr></table>	1	0	1	1	c1-4/5 c2-6/7	11 white	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■		
1	0	1	1	c1-4/5 c2-6/7									
■	■	■											
	<table border="1"><tr><td>1</td><td>1</td><td>c4-6/7</td></tr></table>	1	1	c4-6/7	00 Black	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	1	c4-6/7											
■	■	■											
Trees G/Bk/3	<table border="1"><tr><td>0</td><td>0</td><td>c3-0</td></tr></table>	0	0	c3-0	00 Black	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
0	0	c3-0											
■	■	■											
	<table border="1"><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>c1-4 c2-0</td></tr></table>	1	0	0	0	c1-4 c2-0	01 Red	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■		
1	0	0	0	c1-4 c2-0									
■	■	■											
	<table border="1"><tr><td>0</td><td>0</td><td>c4-0</td></tr></table>	0	0	c4-0	10 Green	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
0	0	c4-0											
■	■	■											
Bunker R/G/3	<table border="1"><tr><td>1</td><td>0</td><td>c3-5</td></tr></table>	1	0	c3-5	00 Black	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	0	c3-5											
■	■	■											
	<table border="1"><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>c1-2 c2-2</td></tr></table>	0	1	1	0	c1-2 c2-2	01 Red	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■		
0	1	1	0	c1-2 c2-2									
■	■	■											
	<table border="1"><tr><td>1</td><td>0</td><td>c4-5</td></tr></table>	1	0	c4-5	10 Green	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	0	c4-5											
■	■	■											
Rough G/W/0	<table border="1"><tr><td>1</td><td>0</td><td>c3-4</td></tr></table>	1	0	c3-4	00 Black	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	0	c3-4											
■	■	■											
	<table border="1"><tr><td>1</td><td>0</td><td>1</td><td>1</td><td>c1-4 c2-6/7</td></tr></table>	1	0	1	1	c1-4 c2-6/7	10 Green	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■		
1	0	1	1	c1-4 c2-6/7									
■	■	■											
	<table border="1"><tr><td>1</td><td>0</td><td>c4-4</td></tr></table>	1	0	c4-4	11 white	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	0	c4-4											
■	■	■											
Water W	<table border="1"><tr><td>1</td><td>1</td><td>c3-7</td></tr></table>	1	1	c3-7	00 Black	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	1	c3-7											
■	■	■											
	<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>c1-7 c2-7</td></tr></table>	1	1	1	1	c1-7 c2-7	01 Red	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■		
1	1	1	1	c1-7 c2-7									
■	■	■											
	<table border="1"><tr><td>1</td><td>1</td><td>c4-7</td></tr></table>	1	1	c4-7	10 Green	<table border="1"><tr><td>■</td><td>■</td><td>■</td></tr></table>	■	■	■				
1	1	c4-7											
■	■	■											

Stipple	Colour	Code	Bit Pattern
Stipple 0	Black	0	000
Stipple 1	Blue	1	001
Stipple 2	Red	2	010
Stipple 3	magenta	3	011
Stipple 4	Green	4	100
Stipple 5	Cyan	5	101
Stipple 6	Yellow	6	110
Stipple 7	White	7	111

The QL program interprets the screen RAM bit pattern producing the relative colours and stipples on the screen.

Stipple	Colour	Code	Bit Pattern
Stipple 3	Black	192	1100 0000
Stipple 3	Red	194	1100 0010
Stipple 3	Green	196	1100 0100
Stipple 3	White	199	1100 0111
Trees	0,4,3	224	1110 0000
Bunker	2,4,3	242	1111 0010
Rough	4,7,0	28	0001 1100
Water	7	199	1100 0111
Fairway	4	196	1100 0100
Green	4,7,3	223	1101 1111

QBITS Golf 2016

QBGolf_Game

Dim arrays Welcome Io(8,6), Fairway Gf(18,9), Green Gn(18,10,6), Shots Gs(18,4), Golf player Gp(10,6), Wind wn(8,8)
Load/Save Device_FileName\$="fp1_QBGolf_data" Note: Can be change to meet user requirements.

PROCEDURE & FuNtion

QBGolf_Welcome Welcome and basic info about Golf Game + scenic graphics Io(8,6)
Select **NEW** - calls **Init_course** or **LOAD** calls **QBGolf_data** Calls **Init_screens**
Init_course Fills course arrays with data Gf(18,9) Gn(18,10,6)
Init_screens Opens screens and Prints#0 Basic Play Instructions Calls: **Init_golfer** **Init_wind**
Init_golfer Draws the Golfer Gp(10,6) Arc of swing and Angle using Graphic Coordinates
Init_wind Loads the data for wind Wd((8,8) to be later drawn by PROCEDURE wind

Golf_Game

Hole_ip Prints#6 hole - length - par
Calls and Initialises the **Fairway**, **Green** and Hazards [Gf(18,8) and Gn(18,10,6) data]
Club_ip Repeats Club play
Power - Angle - shoot (look : ball) Check for out of bounds or to enlarge Green
Test : Check for Hazards (hole in one : **branches : bunker : water : rough**)
End Club_ip
Score/Scorecard / Continue / Save Game / Quit
End Hole_ip
Game Finished - New game / Quit

Fairway Draws the Fairway using BLOCK#3,fw,fd,x,fy,4
trees(n) Draws trees on fairway using Graphic Coordinates
bunker Draws sandpit (Bunkers) using Graphic Coordinates
stream Draws stream using Graphic Coordinates
lake Draws lake using Graphic Coordinates
wind Draws the wind direction indicator and Prints the speed
Green Draws Green Gn(h,10,6) hole hx, hy and flag using Graphic Coordinates

Power Selects club power (draws club Swing indicator) returns pow=p/max
Angle Selects direction (draws revolving indicator) returns angle ang1
Shoot Calculates bx & by new position checks on boundaries calls look and ball
Look Calls getcol FuNtion
getcol(bx,by) FuNtion to return ball bitt colours c1,c2,c3,c4 (0-7)
ball Screen RAM Peek and save old colours Pokes the ball to new position
test Checks for hole in one and hazards
lost Win#3 Prints "Out of Bounds / penalty shot" shots=shots +1 bx,by rest to boundary
branches Win#3 Prints "You have hit a Tree" bx=bx-2:by=by+RND(-1 TO 1)
water Win#3 Prints "WATER!!! Back to Tee" restore colours to add then bx=08 by=106
bunkered Win#6 Prints "BUNKERED!" max=8
rough Win#6 Prints "You are in the rough" max=4
Score Win#6 Prints - results of present hole
Scorecard Win#3 Prints - table of results + opponents results
QBGolf_save Save present results from Gf(18,9), Gn(18,10,6),Gs(18,4)
QBGolf_load Loads a previously saved set of results into Gf(18,9), Gn(18,10,6), Gs(18,4)
Golf_close Closes windows #3 to 6 clears screen: bye... resets with NEW and WTV commands.

QBITS Golf 2016

```
100 REMark QBGolf_201701 (Working Version)
101 :
102 REMark Os-Open screen Gf-Fairway Gn-Green Gs-shots Gp-Golfer Wd-Wind
103 DIM Os(8,6),Gf(18,8),Gn(18,10,6),Gs(18,4),Gp(18,6),Wd(8,8)
104 :
105 device_filename$="flp1_QBGolf_data":REMark user device_filename
106 :
107 MODE 4:WINDOW#1,512,256,0,0:PAPER#1,0:WINDOW#0,372,62,134,192
108 Init_QBits_Welcome:Golf_Game
109 :
110 REMark Welcome screen Os(8,6)
111 DATA 62,30,85,20,,3,PI/2 :REMark cloud
112 DATA 62,40,90,16,,3,PI/2 :REMark cloud
113 DATA 62,180,75,20,,3,PI/2 :REMark cloud
114 DATA 62,195,70,16,,3,PI/2 :REMark cloud
115 DATA 4,40,30,25,,3,PI/2 :REMark landscape
116 DATA 215,170,27,20,,3,PI/2 :REMark Green
117 DATA 215,170,30,25,,3,PI/2 :REMark Green
118 DATA 0,170,29,,7,1,0 :REMark hole
119 :
120 DEFine PROCedure Init_QBits_Welcome
121 ch=3:OPEN#ch,scr_500x172a6x2:PAPER#ch,7:CLS#ch:BORDER#ch,1,2
122 RESTORE 111:BLOCK#ch,496,58,0,110,4
123 FOR a=1 TO 8
124 FOR b=1 TO 6
125 READ dat:Os(a,b)=dat
126 END FOR b
127 FILL#ch,1:INK#ch,Os(a,1)
128 CIRCLE#ch,Os(a,2),Os(a,3),Os(a,4),Os(a,5),Os(a,6)
129 FILL#ch,0
130 END FOR a
131 FILL#ch,1:INK#ch,2:LINE#ch,170,30 TO 170,42 TO 167,40 TO 170,38 TO 169,30 :FILL#ch,0
:INK#ch,0:REMark flag
132 SCALE#ch,100,0,0:CSIZE#ch,2,1:OVER#ch,1
133 INK#ch,0:FOR i=0 TO 4:CORSOR#ch,118+i,20+i:PRINT#ch,'Welcome to QBits Golf'
134 INK#ch,4:FOR i=1 TO 2:CORSOR#ch,120+i,20+i:PRINT#ch,'Welcome to QBits Golf'
135 OVER#ch,0:CSIZE#ch,0,0:INK#ch,0
136 AT#ch,6,18:PRINT#ch,"Each course of 18 holes is randomly generated."
137 AT#ch,7, 8:PRINT#ch,"You play against three other generated players to compare scores."
138 AT#ch,8,16:PRINT#ch,"Pit your skills at Power Driving off the tee and"
139 AT#ch,9,20:PRINT#ch,"Putting the ball once you reach the green."
140 PAPER#ch,4: CORSOR#ch,72,154:PRINT#ch,"press Spacebar for NEW Game - Enter for
LOAD - Esc to Quit"
141 REPEAT lp
142 k=CODE(INKEY$(-1))
143 IF k=10:QBGolf_load:EXIT
144 IF k=27:Golf_close
145 IF k=32:Init_course:EXIT
146 END REPEAT lp
147 Init_screens
148 END DEFine
```

QBITS Golf 2016

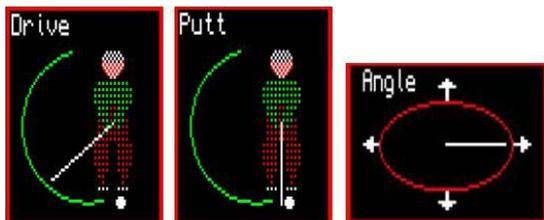
150 DEFine PROCedure Init_course

```
151 ch=3:CURSOR#ch,72,154:CLS#ch,4:PRINT#ch,"Please wait - Initialising.."  
152 FOR h=1 TO 18  
153 Gf(h,1)=RND(3 TO 5):par=Gf(h,1) :REMark par  
154 Gf(h,2)=140*(par-2)+20*(par+RND(-3 TO 3)) :REMark length  
155 Gf(h,3)=RND(40 TO 60) :REMark Trees(n)  
156 Gf(h,4)=RND(120 TO 160) :REMark Bunker  
157 Gf(h,5)=RND(1 TO 8) :REMark Wind direction  
158 Gf(h,6)=RND(1 TO 15) :REMark Wind strength  
159 Gf(h,7)=RND(1 TO 5) :REMark Stream/Lake  
160 Gf(h,8)=RND(60 TO 150) :REMark Location  
161 IF Gf(h,2)<180:Gf(h,2)=180 :REMark length checks  
162 IF Gf(h,2)>525:Gf(h,2)=525 :REMark length checks  
163 x1=RND(-5 TO 5):y1=RND(-5 TO 5)  
164 FOR n=1 TO 10  
165 IF n=1 :i=4:x=48:y=52:r=70-par:e=.8:a=RAD(45)  
166 IF n=2 OR n=3:i=215:x=50+x1:y=55+y1:r=38:e=.7:a=RND(0 TO PI/2)  
167 IF n=4 OR n=5:i=215:x=50+x1:y=40+y1:r=38:e=.8:a=PI/2  
168 IF n=6 :i=243:x=75+x1:y=80+y1:r=12+RND(-2 TO 4):e=.6:a=PI/4  
169 IF n=7 :i=243:x=85+x1:y=65+y1:r=8+RND(2 TO 5) :e=.5:a=0  
170 IF n=8 :i=243:x=15+x1:y=35+y1:r=8+RND(2 TO 5) :e=.6:a=PI/4  
171 IF n=9 :i=224:x=92+x1:y=30+y1:r=6+RND(2 TO 6) :e=.9:a=0  
172 IF n=10 :i=224:x=82+x1:y=20+y1:r=7+RND(1 TO 4) :e=.7:a=0  
173 Gn(h,n,1)=i  
174 Gn(h,n,2)=x  
175 Gn(h,n,3)=y  
176 Gn(h,n,4)=r  
177 Gn(h,n,5)=e  
178 Gn(h,n,6)=a  
179 END FOR n  
180 Gs(h,1)=0  
181 FOR p=2 TO 4:Gs(h,p)=Gf(h,1)+RND(-2 TO 3):END FOR p  
182 ch=3:CURSOR#ch,228+h*6,154:PRINT#ch,"."  
183 END FOR h  
184 END DEFine  
185 :  
186 DEFine PROCedure Init_screens  
187 ch=4:OPEN#ch,scr_128x80a6x174:BORDER#ch,1,2:PAPER#ch,7:CLS#ch  
188 ch=5:OPEN#ch,scr_370x14a136x176:PAPER#ch,4:INK#ch,0:CLS#ch  
189 AT#0,1,13:PRINT#0,"Use left /right arrows for Club Angle"  
190 AT#0,2, 6:PRINT#0,"up/down arrows for Power Swing then press SpaceBar"  
191 AT#0,3,14:PRINT#0,"(check for Wind Speed & Direction)"  
192 Init_golfer:Init_wind  
193 END DEFine  
194 :
```

Note: Course design was achieved mostly by experimentation with the use of randomised variables.

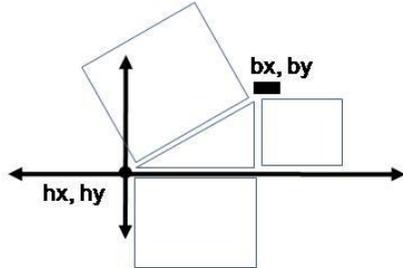
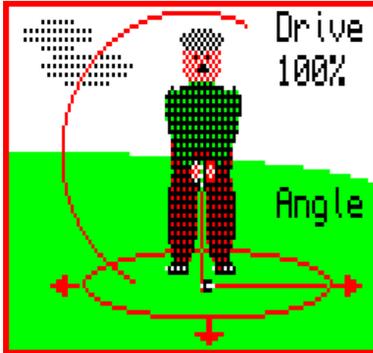
QBITS Golf 2016

```
195 REMark Golf player Gp(10,6)
196 DATA 62,12,74,10,,3,90 :REMark cloud
197 DATA 62,20,66,12,,3,90 :REMark cloud
198 DATA 4,0,36,98,,1,90 :REMark background slope
199 DATA 48,44,19,3,,4,90 :REMark left shoe
200 DATA 48,56,19,3,,4,90 :REMark right shoe
201 DATA 16,45,31,12,,3,0 :REMark left leg
202 DATA 16,55,31,12,,3,0 :REMark right leg
203 DATA 16,50,45,8,1,0 :REMark body
204 DATA 226,50,68,6,,8,0 :REMark head
205 DATA 0,49,69,,7,1,90 :REMark left eye
206 DATA 0,52,69,,7,1,90 :REMark right eye
207 DATA 0,50,66,3,,2,90 :REMark mouth/nose
208 DATA 248,50,72,6,,4,90 :REMark cap
209 DATA 32,50,55,9,,8,90 :REMark chest
210 DATA 32,44,52,9,,3,15 :REMark left arm
211 DATA 32,56,52,9,,3,165 :REMark right arm
212 DATA 226,48,42,2,6,,3,0 :REMark left hand
213 DATA 226,52,42,2,6,,3,0 :REMark right hand
214 :
215 DEFine PROCedure Init_golfer
216 LOCAL i,x,y,r,e,a
217 RESTORE 196::ch=4:ang=0:BLOCK#ch,124,36,0,42,4
218 SCALE#ch,80,0,0:CSIZE#ch,2,0:INK#ch,2:ang=0:STRIP#ch,4
219 CURSOR#ch,61,68:PRINT#ch," ↓"
220 CURSOR#ch,12,58:PRINT#ch," ← ⇒":CSIZE#ch,0,0
221 CIRCLE#ch,51,15,32,,25,RAD(90)
222 FOR a=1 TO 18
223 FOR b=1 TO 6
224 READ dat:Gp(a,b)=dat
225 END FOR b
226 END FOR a
227 FOR l=1 TO 18
228 i=Gp(l,1):x=Gp(l,2):y=Gp(l,3):r=Gp(l,4):e=Gp(l,5):a=Gp(l,6)
229 FILL#ch,1:INK#ch,i:CIRCLE#ch,x,y,r,e,RAD(a):FILL#ch,0
230 END FOR l
231 INK#ch,2:ARC#4,32,16 TO 62,76,-PI
232 END DEFine
233 :
```



Golfer Note: The Club Power Swing and Angle were originally in separate windows. Later they were overlaid with an added scenic touch to give the present graphic display. Further development then combined them into one window working with the same graphics scale settings.

QBITS Golf 2016



Green – calculation of distance from hole
 $Length = INT(\sqrt{(hx-bx)^2 + (hy-by)^2})$

383 DEFine PROCedure Power

```

384 IF putt>0
385 ch=5:INK#ch,0:length=INT(SQRT((hy-by)^2+(hx-bx)^2)):length=length/4
386 CURSOR#ch,196,2:CLS#ch,4:PRINT#ch,"approx: ";length;"mtrs from hole"
387 END IF
388 ch=4:INK#ch,0:p=0:cmax=200/max
389 STRIP#ch,7:AT#ch,0,15:CLS#ch,4:PRINT#ch,club$
390 STRIP#ch,7:AT#ch,1,15:CLS#ch,4:PRINT#ch,cmax;'%'
391 STRIP#ch,4:AT#ch,4,15:PRINT#ch,"Angle"
392 FILL#ch,1:CIRCLE#ch,51,14,1.3:FILL#ch,0
393 OVER#ch,-1:INK#ch,7
394 REPEAT AP_lp
395 LINE#ch,50,44 TO 50-32*SIN(RAD(p)),44-32*COS(RAD(p))
396 LINE#ch,51,15 TO 51+32*COS(RAD(ang)),15-8*SIN(RAD(ang))
397 k=CODE(INKEY$(5))
398 LINE#ch,50,44 TO 50-32*SIN(RAD(p)),44-32*COS(RAD(p))
399 LINE#ch,51,15 TO 51+32*COS(RAD(ang)),15-8*SIN(RAD(ang))
400 IF k=192:ang=ang+10:IF ang>=360:ang=0
401 IF k=200:ang=ang-10:IF ang<=0:ang=360
402 IF k=208:p=p+4:IF p>club:p=0
403 IF k=216:p=p-4 :IF p<0:p=0
404 IF k=32
405 FOR a=p TO 0 STEP -2
406 LINE#ch,50,44 TO 50-32*SIN(RAD(a)),44-32*COS(RAD(a))
407 LINE#ch,50,44 TO 50-32*SIN(RAD(a)),44-32*COS(RAD(a))
408 END FOR a
409 INK#ch,4:FILL#ch,1:CIRCLE#ch,51,14,1.3:FILL#ch,0
410 POKE_W add,colour:POKE_W add+128,colour1
411 IF p<80:wdx=0:wdy=0:END IF
412 pow=p/max:ang1=(12-ang/30)*PI/6:EXIT AP_lp
413 END IF
414 IF k=27:STOP
415 END REPEAT AP_lp
416 OVER#ch,0
417 END DEFine

```

QBITS Golf 2016

234 REMark Wind direction Wd(8,8)

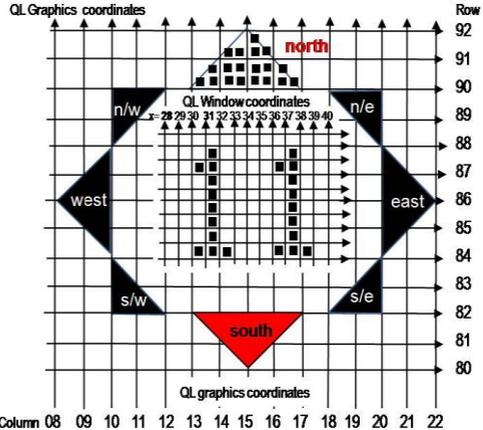
```

235 DATA 15,92,17,90,13,90,15,92      :REMark north
236 DATA 17,90,20,90,20,88,17,90      :REMark n/e
237 DATA 22,86,20,88,20,84,22,86      :REMark east
238 DATA 20,84,20,82,17,82,20,84      :REMark s/e
239 DATA 15,80,17,82,13,82,15,80      :REMark south
240 DATA 13,82,10,82,10,84,13,82      :REMark s/w
241 DATA 8,86,10,88,10,84,8,86        :REMark west
242 DATA 10,88,10,90,13,90,10,88      :REMark n/w
243 :
```

244 DEFine PROCedure Init_wind

```

245 RESTORE 235
246 FOR a=1 TO 8
247 FOR b=1 TO 8
248 READ dat:Wd(a,b)=dat
249 END FOR b
250 END FOR a
251 END DEFINE
```



Note: wind uses graphics coordinates for direction and pixel coordinates for the printed speed

362 DEFine PROCedure wind

```

363 wdx=0:wdy=0:wnd=Gf(h,5):wns=Gf(h,6)
364 CURSOR#ch,4,2:INK#ch,0:PRINT#ch,"Wind speed/direction"
365 FOR a=1 TO 8
366 IF wnd=a:INK#ch,2:ELSE INK#ch,0
367 FILL#ch,1:LINE#ch,Wd(a,1),Wd(a,2) TO Wd(a,3),Wd(a,4) TO Wd(a,5),Wd(a,6) TO
Wd(a,7),Wd(a,8):FILL#ch,0
368 IF wns<10:cp=32:ELSE cp=28
369 INK#ch,0:CURSOR#ch,26,20:PRINT#ch,' ':CURSOR#ch,cp,20:PRINT#ch,wns
370 END FOR a
371 SElect ON wnd
372 ON wnd=1:wdx=0      :wdy=-wns/4      :REMark north
373 ON wnd=2:wdx=wns/4  :wdy=-wns/8      :REMark n/e
374 ON wnd=3:wdx=wns    :wdy=0          :REMark east
375 ON wnd=4:wdx=wns/4  :wdy=wns/8      :REMark s/e
376 ON wnd=5:wdx=0      :wdy=wns/4      :REMark south
377 ON wnd=6:wdx=-wns/4 :wdy=wns/8      :REMark s/w
378 ON wnd=7:wdx=-wns   :wdy=0          :REMark west
379 ON wnd=8:wdx=-wns/4 :wdy=-wns/8      :REMark n/w
380 END SElect
381 END DEFine
```

QBITS Golf 2016

```
253 DEFine PROCedure Golf_Game
254 Sc=0:ps1=0:ps2=0:ps3=0:ps4=0:ph=0
255 FOR n=1 TO 18:IF Gs(n,1)>0:ph=n:END IF :END FOR n
256 FOR h=1 TO 18
257 IF ph>0:h=ph+1:ph=0
258 club$='Drive':club=200:shots=0:putt=0:ang=0:max=2
259 ch=3:WINDOW#ch,496,170,8,3:PAPER#ch,4,7,1:CLS#ch:Fairway
260 ch=1:WINDOW#ch,90,60,412,gy:Green
261 ch=3:WINDOW#ch,496,170,8,3:BLOCK#ch,8,12,4,98,7
262 finished=0:look:ball
263 REPEAT club_lp
264 Power:shots=shots+1:max=2:shoot
265 IF putt=1
266   ch=3:WINDOW#ch,240,164,260,6:CLS#ch:BORDER#ch,1,2
267   club$='Putt ':club=80:Green:shoot:putt=2
268 END IF
269 IF finished OR shots>9:EXIT club_lp
270 END REPEAT club_lp
271 Score:Scorecard
272 END FOR h
273 Golf_close
274 END DEFine
275 :
276 DEFine PROCedure Golf_close
277 WINDOW#1,512,256,0,0:PAPER#1,0:CLS#1:FOR ch=3 TO 5:CLOSE#ch
278 WINDOW#0,480,40,16,216:INK#0 7:PRINT#0,"Bye...":PAUSE 50:STOP
279 END DEFine
```

Golf Game Notes:

The programming has developed through a number of stages and I'm still looking to improve and refine the coding. There were a few hiccups along the way, the ball was originally four pixels by two, but at times it seems to split into three segments, occasionally it did not appear. A change to the multipliers in Procedure Ball and (as if by accident) it has reduced the ball to a three by two pixel size. The problem that occurs now is when the hazard:- branches moves the ball out of the tree, sometimes the location is not clear when amongst a group of trees or if another hazard is encountered such as in the rough or a water.

The Fairway cross wind factor in calculating bx, by was originally added to the power/distance variables, but produced some odd readings. A change to the wdx, wdy settings and adding it to the existing bx, by values first appears to have corrected this.

The variables used for boundary markings, the position of the Green on the Fairway, the hole position in both Fairway and enlarged Green phases are key to the workings and were tested for accuracy. Any future changes to window sizes and position will require further checks for accuracy.

QBITS Golf 2016

281 DEFine PROCEDURE Fairway

```
282 ch=3:par=Gf(h,1):dist=Gf(h,2):CSIZE#ch,0,0:CLS#ch,0:INK#ch,0
283 CURSOR#ch,300,2:PRINT#ch,"Hole "&h&" Length "&dist&"mtrs Par "&par
284 SCALE#ch,100,0,0:fw=2:fd=60-(5*par):fp=fd:fm=7-par
285 FOR fx=4 TO 416 STEP 2
286 fd=fp+(fx/24)+RND(-fm TO fm)
287 IF par=3:fy=80-(fx/8)+RND(-1 TO 4):fy1=fy/1.5:END IF
288 IF par=4:IF fx<=160+dist/4:fy=90-(fx/4)+RND(-1 TO 4):fx1=fx:fy1=fy:ELSE fy=fy1+RND(-
2 TO 2):END IF
289 IF par=5:IF fx<=60+dist/4 :fy=90-(fx/3)+RND(-1 TO 4):fx1=fx:fy1=fy:ELSE fy=fy1+((fx-
200)/4)+RND(-2 TO 2):END IF
290 IF fy<12:fy=12
291 IF fy>100:fy=100
292 BLOCK#ch,fw,fd,fx,fy,4
293 END FOR fx
294 IF par<>3:FILL#ch,1:INK#ch,4:CIRCLE#ch,fx1/2,3,95-fy1,fd/3,.7,1:FILL#ch,0
295 gy=INT(fy):wind:Bunker
296 IF Gf(h,7)<=3:Stream
297 Trees
298 IF Gf(h,7)=5 AND par<>3:Lake
299 bx=8:by=106:zx=8:mx=248:zy=16:my=164:hx=230:hy=gy+36
300 END DEFine
301 :
302 DEFine PROCEDURE Trees
303 FOR t=1 TO Gf(h,3)
304 tx=RND(20 TO 170):ty=RND(10 TO 40):tr=1.5+RND(1 TO 10)/5
305 INK#ch,0:LINE#ch,tx,ty TO tx,ty-tr*2 TO tx+.2,ty-tr*2 TO tx+.2,ty
306 INK#ch,4,0,3:FILL#ch,1:CIRCLE#ch,tx,ty,tr:FILL#ch,0
307 tx=RND(30 TO 120):ty=RND(70 TO 90):tr=1.5+RND(1 TO 10)/5
308 INK#ch,0:LINE#ch,tx,ty TO tx,ty-tr*2 TO tx+.2,ty-tr*2 TO tx+.2,ty
309 INK#ch,4,0,3:FILL#ch,1:CIRCLE#ch,tx,ty,tr:FILL#ch,0
310 IF par=5
311 tx=RND(140 TO 200):ty=RND(60 TO 90):tr=1.5+RND(1 TO 10)/5
312 INK#ch,0:LINE#ch,tx,ty TO tx,ty-tr*2 TO tx+.2,ty-tr*2 TO tx+.2,ty
313 INK#ch,4,0,3:FILL#ch,1:CIRCLE#ch,tx,ty,tr:FILL#ch,0
314 END IF
315 END FOR t
316 END DEFine
317 :
318 DEFine PROCEDURE Bunker
319 px=Gf(h,4):INK#ch,243
320 IF par=3:px=px:py=(130-fy)/1.5:pr=RND(7 TO 9)
321 IF par=4:px=px:py=(130-fy)/1.5:pr=RND(6 TO 8)
322 IF par=5:px=px:py=(160-fy)/1.5:pr=RND(5 TO 7)
323 FILL#ch,1:CIRCLE#ch,px,py,pr,.4,PI/2:FILL#ch,0
324 FILL#ch,1:CIRCLE#ch,px+2,py-2,pr,.6,.7+RND(-.2 TO .5):FILL#ch,0
325 END DEFine
326 :
```

QBITS Golf 2016

327 DEFine PROCEDURE Stream

```

328 sx=Gf(h,8):sy=6:INK#ch,7
329 REPEAT stream_lp
330   sx=sx+RND(-1 TO 1):sy=sy+RND(1 TO 2)
331   FILL#ch,1:CIRCLE#ch,sx,sy,RND(2 TO 4),.3,RND(0 TO .45):FILL#ch,0
332   IF sy>90:RETurn
333 END REPEAT stream_lp
334 END DEFine

```

335 :

336 DEFine PROCEDURE Lake

```

337 IF par=4:lx=Gf(h,8):ly=(120-fy)/1.5
338 IF par=5:lx=Gf(h,8):ly=(150-fy)/1.5
339 FOR l=1 TO 4
340   INK#ch,7:FILL#ch,1:CIRCLE#3,lx+l*3,ly+l*2,.3+RND(par TO 8),.65,(RND(0 TO 1.4)):FILL#ch,0
341 END FOR l
342 END DEFine

```

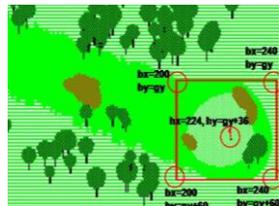
343 :

344 DEFine PROCEDURE Green

```

345 SCALE#ch,100,0,0:tt=.2
346 IF putt=1
347   POKE_W add.colour:POKE_W add+128,colour1
348   zx=132:mx=246:zy=8:my=166:hx=196:hy=104:dist=320
349   bx=130+INT((bx-206)*2.66):by=8+INT((by-gy)*2.76):tt=.5
350 END IF
351 FOR g=1 TO 10
352   i=Gn(h,g,1):x=Gn(h,g,2):y=Gn(h,g,3)           :REmark Green
353   r=Gn(h,g,4):e=Gn(h,g,5):a=Gn(h,g,6)           :REmark Bunkers
354   IF g>=9:INK#ch,0:LINE#ch,x,y TO x,y-r*1.5 TO x+tt,y-r*1.5 TO x+tt,y
355   FILL#ch,1:INK#ch,i:CIRCLE#ch,x,y,r,e,a:FILL#ch,0 :REmark Trees
356 END FOR g
357 hfx=60:hfy=40:IF putt=0:hfr=2:ELSE hfr=1
358 FILL#ch,1:INK#ch,0:CIRCLE#ch,hfx,hfy,1.25:FILL#ch,0 :REmark hole
359 FILL#ch,1:INK#ch,2:LINE#ch,hfx,hfy TO hfx,hfy+12 TO hfx-3,hfy+10 TO hfx,hfy+8:FILL#ch,0 :REmark flag
360 END DEFine
361 :

```



Green Notes:

Boundaries **zx,mx,zy,my** and hole **hx,hy** change. The ball **bx,by** recalculated from position on the Fairway Green to the enlarged Green. The hole radius resized in line with enlarged Green.

QBITS Golf 2016

```
419 DEFine PROCedure shoot
420 bx=INT(bx+wdx+COS(ang1)*525*(pow/dist))
421 by=INT(by+wdy-SIN(ang1)*525*(pow/dist))
422 IF bx<zx OR bx>mx OR by<zy OR by>my:lost
423 IF putt=0 AND by>gy AND by<gy+60 AND bx>204:putt=1:RETurn
424 look:ball:test
425 END DEFine
426 :
427 DEFine PROCedure look
428 c1=getcol(bx,by)
429 c2=getcol(bx+1,by)
430 c3=getcol(bx,by-1)
431 c4=getcol(bx+1,by+1)
432 REMark PRINT#0,c1,c2,c3,c4
433 END DEFine
434 :
435 DEFine FuNction getcol(bx,by)
436 LOCal add
437 add=INT(131072+128*by+INT(bx/4)*2)
438 bit=(bx MOD 4)*2
439 bitt=(2^(7-bit))
440 gb=4*((bitt && PEEK (add))<>0)
441 bitt=(2^(7-bit)*1.5)
442 rb=(bitt && PEEK (add+1))/bitt*3
443 RETurn gb+rb
444 END DEFine
445 :
446 DEFine PROCedure ball
447 add=INT(131072+128*by+INT(bx/4)*2)
448 colour=PEEK_W (add):colour1=PEEK_W (add+128)
449 bit=(bx MOD 4)*2
450 bitt=(255-2^(7-bit)*1.75)
451 IF bitt-INT(bitt) THEN bitt=INT(bitt*.988)
452 newcol=(bitt && PEEK (add))*256+(bitt && PEEK(add+1))
453 newcol1=(bitt && PEEK (add+128))*256+(bitt && PEEK (add+129))
454 :
455 FOR n=0 TO 4
456 POKE_W add,colour :POKE_W add+128,colour1 :PAUSE 5
457 POKE_W add,newcol:POKE_W add+128,newcol1:PAUSE 5
458 END FOR n
459 END DEFine
460 :
```

Screen RAM Note: The above Procedures and Function is where the screen RAM calculation on page 8 come into play.

QBITS Golf 2016

```
461 DEFine PROCedure test
462 ch=5:CURSOR#ch,1,2:CLS#ch,4
463 IF ABS(bx-hx+1)<=1 AND ABS(by-hy+1)<=1
464   POKE_W add,colour:POKE_W add+128,colour1:finished=1:RETurn
465 END IF
466 IF c1=4 AND c3=0 OR c1=0 AND c4=4 :branches
467 IF c1=2 AND c3=5 OR c1=5 AND c4=2 :bunkered
468 IF c1=7 AND c3=4 OR c1=4 AND c4=7 :rough
469 IF c1=7 AND c3=7 OR c1=7 AND c4=7 :water
470 END DEFine
471 :
472 DEFine PROCedure lost
473 ch=5:CURSOR#ch,12,2:CLS#ch,4:PRINT#ch,"Out of Bounds / penalty shot"
474 IF bx<zx:bx=zx
475 IF by<zy:by=zy
476 IF bx>mx:bx=mx
477 IF by>my:by=my
478 shots=shots+1:PAUSE 50
479 END DEFine
480 :
481 DEFine PROCedure branches
482 ch=5:CURSOR#ch,12,2:CLS#ch,4:PRINT#ch,"You have hit a tree!"
483 bx=bx-2:by=by+RND(-1 TO 1)
484 POKE_W add+128,colour1
485 look:ball:test
486 END DEFine
487 :
488 DEFine PROCedure water
489 ch=5:CURSOR#ch,12,2:CLS#ch,4:PRINT#ch,"WATER!!! Back to Tee"
490 bx=8:by=106
491 POKE_W add,colour:POKE_W add+128,colour1
492 PAUSE 20:ball
493 END DEFine
494 :
495 DEFine PROCedure bunkered
496 ch=5:CURSOR#ch,12,2:CLS#ch,4:PRINT#ch,"BUNKERED!"
497 PAUSE 20:max=8
498 END DEFine
499 :
500 DEFine PROCedure rough
501 ch=5:CURSOR#ch,12,2:CLS#ch,4:PRINT#ch,"You Are In The Rough"
502 PAUSE 20:max=4
503 END DEFine
504 :
```

Hazards & Boundaries Note: The above Procedures deal with the boundaries (**lost**) and hazards (**branches** [trees], **water**, **bunkered** and **rough**).

QBITS Golf 2016

505 DEFine PROCedure Score

```
506 ch=5:CURSOR#ch,12,2:CLS#ch,4
507 IF shots=1 :PRINT#ch,"Hole in One! - Superb Shot":GO TO 514
508 IF par<shots :PRINT#ch,shots-par;" over par on this hole"
509 IF par=shots :PRINT#ch," A Par - Not Bad"
510 IF par-1=shots:PRINT#ch," Well played... A birdie"
511 IF par-2=shots:PRINT#ch," Fantasic shot... An Eagle"
512 IF par-3=shots:PRINT#ch," An Albatross... Incredible"
513 IF shots>9 :PRINT#ch," Your out of shots"
514 PAUSE 20:Gs(h,1)=shots
515 END DEFine
```

517 DEFine PROCedure Scorecard

```
518 ch=3:WINDOW#ch,496,170,8,3:PAPER#ch,7:CLS#ch:INK#ch,0
519 CURSOR#ch,40,4:PRINT#ch,"Scorecard"
520 Sc=0:ps1=0:ps2=0:ps3=0:ps4=0:cpar=0:FOR hp=1 TO 18:cpar=cpar+Gf(hp,1)
521 CURSOR#ch,352,4:PRINT#ch,"Par for Course ";cpar
522 CURSOR#ch,40,16:PRINT#ch,"Hole Length Par Player shots"
523 CURSOR#ch,274,16:PRINT#ch,"Hole Length Par Player shots"
524 FOR r=1 TO 9
525 AT#ch,2+r,8:PRINT#ch,r ";" ";Gf(r,2) ";" ";Gf(r,1) ";" "
526 AT#ch,2+r,47:PRINT#ch,r+9;" ";Gf(r+9,2);" ";Gf(r+9,1);" "
527 END FOR r
528 AT#ch,12,18:PRINT#ch," Player 1 2 3 4"
529 FOR hs=1 TO h
530 IF hs>=10:r=hs-9:sp=63:ELSE r=hs:sp=26
531 AT#ch,2+r,sp:PRINT#ch,Gs(hs,1);';Gs(hs,2);';Gs(hs,3);';Gs(hs,4)
532 Sc=Sc+Gs(hs,1)-Gf(hs,1) :REMark Course par evaluation
533 ps1=ps1+Gs(hs,1)::AT#ch,12,64:PRINT #ch,"Player 1 ";ps1
534 ps2=ps2+Gs(hs,2)::AT#ch,13,64:PRINT #ch,"Player 2 ";ps2
535 ps3=ps3+Gs(hs,3)::AT#ch,14,64:PRINT #ch,"Player 3 ";ps3
536 ps4=ps4+Gs(hs,4)::AT#ch,15,64:PRINT #ch,"Player 4 ";ps4
537 END FOR hs
538 AT#ch,14,6
539 IF Sc=0:PRINT#ch,"Overall you were on Par"
540 IF Sc<0:PRINT#ch,"Overall you were under Par"
541 IF Sc>0:PRINT#ch,"Overall you were over Par"
542 CURSOR#ch,36,154:PRINT#ch,"press Spacebar to continue - Enter to SAVE - Ecs to Quit!"
543 REPEAT lp
544 k=CODE(INKEY$(-1))
545 IF k=32:ch=5:CLS#ch:EXIT lp
546 IF k=10:QB Golf_save
547 IF k=27:Golf_close
548 END REPEAT lp
549 END DEFine
```

QBITS Golf 2016

Scorecard				Par for Course 65			
Hole	Length	Par	Player shots	Hole	Length	Par	Player shots
1	180	3	3 6 4 4	10	340	4	
2	525	5	5 6 8 6	11	420	4	
3	300	4	5 2 2 4	12	180	3	
4	180	3	4 3 6 6	13	525	5	
5	180	3	4 4 5 3	14	220	3	
6	220	3	5 3 2 4	15	360	4	
7	180	3	4 1 4 3	16	180	3	
8	500	5	5 6 7 4	17	200	3	
9	200	3		18	380	4	
Player 1 2 3 4				Player 1 35			
Overall you were over Par				Player 2 31			
press Spacebar to continue - Enter to SAVE - Ecs to Quit!				Player 3 38			
				Player 4 34			

551 DEFine PROCedure QBGolf_save

```

552 ch=3:CURSOR#ch,36,154:PRINT#ch,"Please wait - Saving..."
553 DELETE device_filename$
554 OPEN_NEW#9,device_filename$
555 FOR a=1 TO 18
556 FOR bf=1 TO 8:PRINT#9,Gf(a,bf):END FOR bf
557 FOR bs=1 TO 4:PRINT#9,Gs(a,bs):END FOR bs
558 FOR b=1 TO 10
559 FOR c=1 TO 6:PRINT#9,Gn(a,b,c):END FOR c
560 END FOR b
561 CURSOR#ch,168+a*6,154:PRINT#ch,','
562 END FOR a
563 CLOSE#9
564 CURSOR#ch,36,154:PRINT#ch,"press Spacebar to continue - Enter to SAVE - ESC to quit"
565 END DEFine

```

567 DEFine PROCedure QBGolf_load

```

568 ch=3:CURSOR#ch,36,154:CLS#ch,4:PRINT#ch,"Please wait - Loading..."
569 OPEN_IN#9,device_filename$
570 FOR a=1 TO 18
571 FOR bf=1 TO 8 :INPUT#9,Gf(a,bf) :END FOR bf
572 FOR bs=1 TO 4 :INPUT#9,Gs(a,bs) :END FOR bs
573 FOR b=1 TO 10
574 FOR c=1 TO 6:INPUT#9,Gn(a,b,c):END FOR c
575 END FOR b
576 CURSOR#ch,168+a*6,154:PRINT#ch,','
577 END FOR a
578 CLOSE#9
579 END DEFine

```

QBITS Golf 2016

QBGolf

Having obtained a copy of **QBGolf** SuperBASIC code and loaded it into a recognised QL device. Use the QDOS command LRUN, as shown:-

LRUN WIN1_QBGolf.

All being well you can spend hours of fun playing my armchair version of Golf.

Notes on QL2K emulator

Both the **QLAY & QL2K emulators** use an application tool to create a QDOS directory file and append or delete files in it. Creating a new qlay.dir file first open a Windows **Command Prompt** (Win 7 Press Start button in *search programs and files* box type **command prompt**: Win 10 in *ask me anything* box type **command prompt**.)

Activate the command prompt window then navigate with DOS commands to the drive and Windows File Directory folder that holds your QL Files.

```
i.e C:\>chdir H:\QL\FDIR\WIN1_ H:\QL\FDIR\WIN1_>dir
```

This will list the files as a DOS directory. This needs to also contain a copy of **QLAYT-86.EXE** or **QLAY-X64.EXE** downloaded with **QLAY** or **QL2K**

At the DOS prompt now enter this command: -

```
H:\QL\FDIR\WIN1_>qlayt-x64.exe -c qlay.dir
```

This should create a directory file qlay.dir to which you can now append files. For example:-

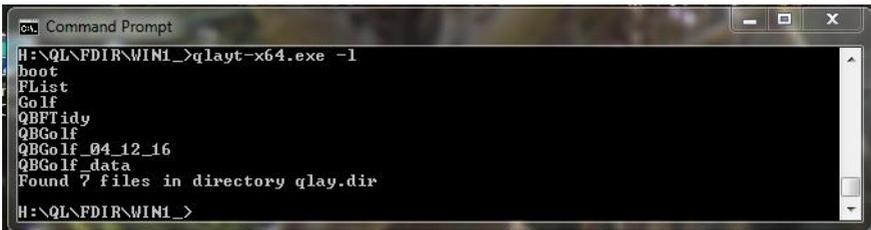
```
H:\QL\FDIR\WIN1_>qlayt-x64.exe -i Boot
```

This will append the File named '**Boot**' to the qlay.dir .

Once you have appended your files you can use the following command to list them:-

```
H:\QL\FDIR\WIN1_> qlayt-x64.exe -l
```

A list of files should now be shown contained within the qlay.dir



```
Command Prompt
H:\QL\FDIR\WIN1_>qlayt-x64.exe -l
boot
FList
Golf
QBFTidy
QBGoIf
QBGoIf_04_12_16
QBGoIf_data
Found 7 files in directory qlay.dir
H:\QL\FDIR\WIN1_>
```

To remove a file at the DOS prompt type:-

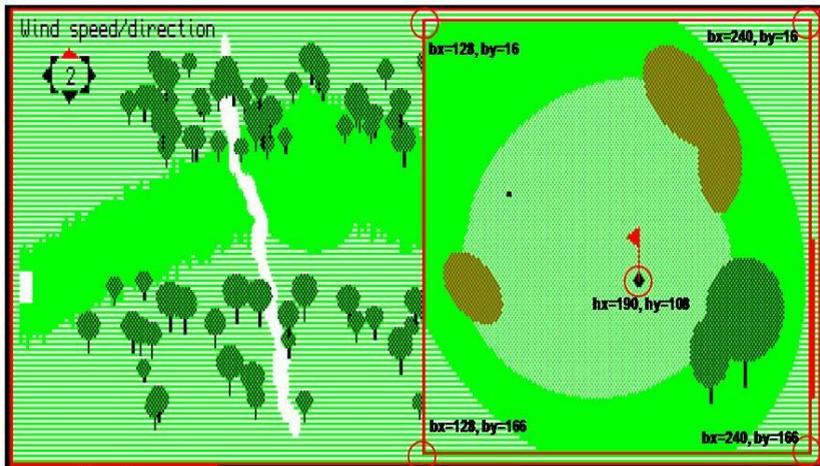
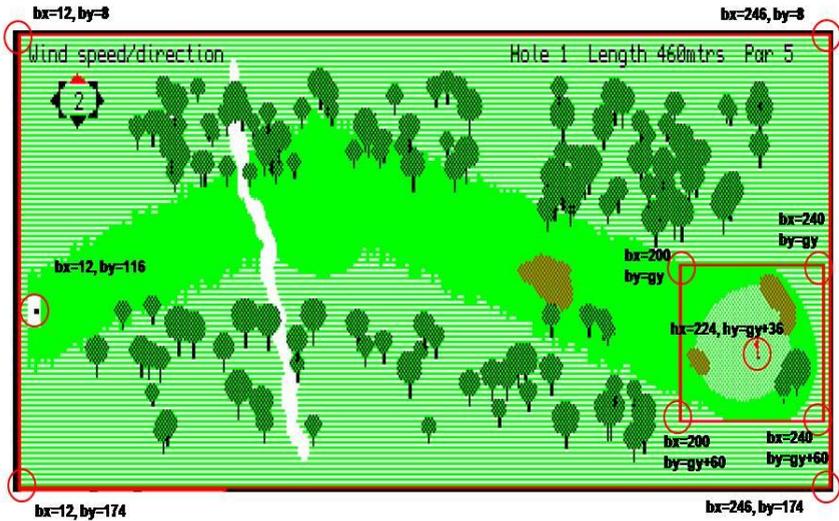
```
H:\QL\FDIR\WIN1_>qlayt-x64.exe -r Golf
```

This will remove the File named '**Golf**' from the qlay.dir .

Running the **QL2K emulator** the files listed in WIN1_ should now be readable by the QDOS DIR command; however these files will not be loadable or run if not compatible with the QDOS operating system you have. This even applies to QL software that does not work with or only works with certain versions of QL ROM's or with added Toolkit extensions.

QBITS Golf 2016

Fairway: Boundaries, Ball, Green & Hole location



Green: Boundaries & Hole location

QBITS Golf 2016

QLZK

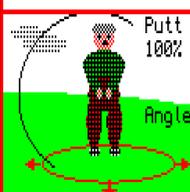
QL Commands Screen Language Help

Scorecard				Par for Course 65		
Hole	Length	Par	Player shots	Hole Length	Par	Player shots
1	180	3	3 6 4 4	10	340	4
2	525	5	5 6 8 6	11	420	4
3	300	4	5 2 2 4	12	180	3
4	180	3	4 3 6 6	13	525	5
5	180	3	4 4 5 3	14	220	3
6	220	3	5 3 2 4	15	360	4
7	180	3	4 1 4 3	16	180	3
8	500	5	5 6 7 4	17	200	3
9	200	3	5 6 5 6	18	380	4

Player 1 2 3 4

Player 1 40
Player 2 37
Player 3 43
Player 4 40

Overall you were over Par
press Spacebar to continue - Enter to SAVE - Esc to Quit!



Putt 100%

Angle

2 over par on this hole

Use left /right arrows for Club Angle
up/down arrows for Power Swing then press SpaceBar
(check for Wind Speed & Direction)

