

Mind the Gasp

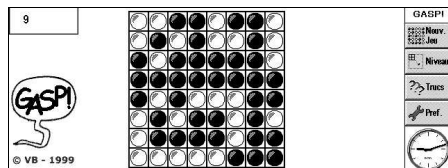
Vincent Benner's Gasp puzzle game is reviewed here by Damian Walker.

The computer has created a whole new world of solitaire games. These are games that, unlike peg solitaire, would be impractical as a board game. One of these is Vincent Benner's Gasp for the Series 5, a variation on the popular Lights Out puzzle. The game starts with a square array of two-sided Reversi pieces, all white. The player can touch any piece, which will have the effect of flipping all the adjacent pieces, so that all that were white become black, and vice versa. The object of the puzzle is to turn all the pieces black side up.

Vincent Benner's puzzle differs from the other versions that I have played. Usually, touching a piece usually affects that piece, along with the maximum of four adjacent pieces: those above, below, and to the left and right of it. In Gasp, the piece itself is not affected, but up to eight adjacent pieces are, including those diagonally adjacent. Mr Benner cites his source for the rules as the defunct magazine *Jeu & Strategie*, so I have to give him the benefit of the doubt and assume that he accurately represents the rules as given in the magazine.

You may detect some scepticism in my mind. At the initial level of four pieces, in a 2x2 grid, the solution is very easy. But the difficulty of the next level, 3x3, is greatly increased. In fact, I am beginning to wonder if it is possible. Let me explain.

The solution to each level requires that each piece is flipped an odd number of times. Given that the solution is symmetrical, though, in the 3x3 grid (or any other grid of an odd-numbered size) the central piece will always be flipped an even number of times, because tapping on it doesn't affect the piece itself—so it will never end up black side up. Try it and see! Could it be that only the grids of an even-numbered size have a solution in this game? My assumption about the solution



being necessarily symmetrical may be wrong, so please write in if you find a solution to the 3x3, 5x5 or 7x7 puzzles.

But, setting aside my reservations about the rules, let's look at the other aspects of the game. It's certainly a fun little puzzle. I spent a significant amount of time trying to work out the 3x3 conundrum. I've yet to solve the 4x4 grid either, though I've no doubt that it's possible. Gasp isn't a game I can see myself coming back to, though, once I've solved the puzzle. What's lacking is a "random" option, to scramble the puzzle into a random pattern, making every attempt different.

The graphics are quite pleasing, and of the line-drawn variety, with a bit of shading on the pieces. There is an attractively-drawn title screen, too. The user interface is simple to use, though there are few options. Sound is completely absent.

But I think that the game has been released unfinished. There is a "Prefs" button on the toolbar, which appears to do nothing. Rules are given in a couple of text files (one French, one English) instead of using standard EPOC help. But apart from this, the game is of good quality and is up to the standards that EPOC users expect. The in-game prompts and menus are entirely in French, but are few so it shouldn't be a problem to people who don't speak the language.

So why not give it a try? And let me know if you do manage to solve the 3x3 level!

By	Vincent Benner
URL	psion.cynningstan.org.uk
Licence	Freeware
Compatibility	Series 5/5mx
Rating	☆☆☆



Readers of last month's issue will remember that I reviewed Alignment, by PocketIQ. Its author David Steer has been in touch with me recently, and has kindly given a registration code for readers of *Epic Entertainer*. The details to enter into the Registration... option on the game's menu are:

- Name: **EPOC Entertainer**
- Id: **3201452121-2111330123**

This month there are two more reviews: Palmscape's Jumpy, and Vincent Benner's Gasp puzzle game. We also have the concluding part of the programming series The Small Screen.

From next month we'll take a break from programming for a while. There's a new series about playing games under emulation on your EPOC32 machine. In addition to that will be the usual reviews.

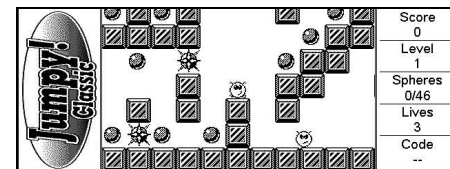
Let me know what you've made of the magazine so far. I'd be particularly interested to see if anyone has made a foray into programming as a result of the articles published so far in the magazine. You can contact me at the usual address.

entertainer@cynningstan.org.uk

The High Jump

Damian Walker reviews Palmscape's well-known platform game Jumpy! Plus, by Jon Read.

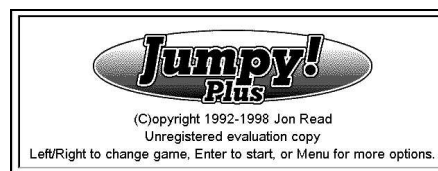
Platform games have a long history in and out of the arcades. Mr Do's Castle and many others were popular in the arcades. Owners of the ZX Spectrum will remember the classics Manic Miner and Jet Set Willy from the 1980s. Early in the following decade the IBM PC had the famous Commander Keen series from Apogee, attractively-drawn and full of humour. Games consoles entertained players with characters like Sonic the Hedgehog and the Mario Brothers. At

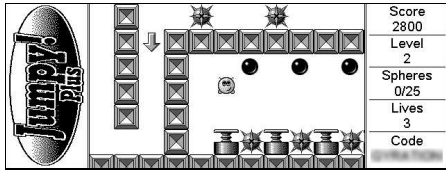


the same time, Jon Read released his *Jumpy!* game for the Psion Series 3 Classic.

In 1998 an update to *Jumpy!*, called *Jumpy! Plus*, was released for the Psion Series 5. It featured improved graphics, reminiscent of the Commander Keen series on the PC, and a whole alternative level set. It's this update that I'm about to review.

Jumpy! Plus is Shareware. Unfortunately it doesn't seem to be possible to register it any





more. The site that various sources link to appears to offer the game, but it gives a “page not found” error when a visitor tries to register. Neither has the author, or anyone else, released a generic registration code for the game. But in its unregistered state, Jumpy! Plus is still playable, with four levels each of the “classic” and “plus” level sets accessible.

“Jumpy” is the name of the cute, if a little obese, character in this game. He can run left and right along the various platforms, climb up and down ladders, collect spheres, and of course jump—about three times his own body height. The world in which he lives is a series of large levels, containing solid platforms, icy platforms, crumbling platforms, sticky floors, conveyor belts, ladders, spiky mines, nasty creatures, switches that affect disappearing walls and floors, and that most valuable commodity, spheres. In each level Jumpy must collect spheres, before progressing to an exit which remains locked fast until the last sphere is collected. As the levels are much bigger than the screen, the display scrolls around Jumpy as he moves. This scrolling is very smooth and the controls feel very responsive.

Back in the early 1980s platform games were probably my favourite genre. Most of them are simple, non-cerebral fun, and though I have a preference for strategy games now, I keep going back to the occasional platformer. It's examples like Jumpy that keep dragging me back there. It's easy to get into, and therefore good fun right from the start. There is a good range of platform types on successive levels, and the levels themselves are well-designed.

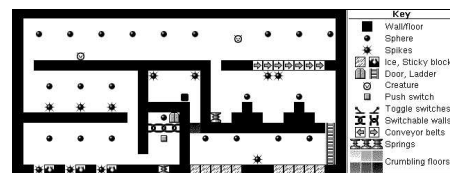
Pixel-perfect movement is occasionally required, with Jumpy having to poise himself

precisely before leaping across a ledge. This can be frustrating enough to ruin the game's addictiveness, and I've occasionally got fed up enough to stop playing and do something else. But using the keys Q, A, O and P, instead of the cursor keys, seems to make these pixel-perfect movements a bit easier.

The presentation of Jumpy! Plus is very good, and it is probably the best of the platform games available to EPOC32 gamers. The graphics, as I've said, are reminiscent of Commander Keen, the shading of the spheres evoking Keen's similarly shaped lollipops. The only thing that lets the game down in this respect is the very plain-looking scoreboard, which could have done with some better decoration.

Sound is supported, and there is a variety of options to control it. The usual volume control is there, but you can also choose between digital sound, the original Series 3 Classic beeps, or silence. With digital sound you can opt not to hear the spheres being collected, which makes the animation smoother on the Series 5 Classic. This isn't necessary on the Series 5mx, which is smooth with or without these noises.

It's a great shame that registration is no longer possible, and that the later levels can't be unlocked. But if you've never played Jumpy! before, then there's a good couple of hours of fun to be had with the unregistered version, and it's therefore well worth installing for a while.



By **Palmscape**
 URL psion.cynningstan.org.uk
 Licence **Shareware**
 Compatibility **Series 5/5mx**
 Rating **★★★★**

The Small Screen

Damian Walker concludes the programming series on screen optimisation.

So far we've looked at two ways of expanding or contracting our graphics to use the screen of any EPOC32 device. A third method requires some more effort in drawing, but can achieve the best results. This involves having separate bitmaps for each size of screen you wish to support—five in all, if you want to support all EPOC devices—and instructing the program to load the appropriate one for the machine it's running on. Variations of this method is used in games such as *Mille Bornes*, and my own *Pebbles on the move*, and can achieve spectacular results, because graphics can be separately drawn to get the best out of each resolution and colour depth supported.

In our demonstration, the existing *Rock* bitmap is suitable for the Osaris; for that machine rename it *320x200.mbm*. For other machines you need to draw separate images; the table shows what size the images need to be and what the file should be called for each machine:

Osaris	20x20	320x200.mbm
Revo	20x20	480x160.mbm
Series 5	30x30	640x240.mbm
Geofox	40x40	640x320.mbm
netBook	40x40	640x480.mbm

Note we've got some duplication here; these image sizes are the optimum to allow a field of 16x8 squares. If you want a playing field or window of other dimensions, then the image sizes will be different, and all five machines might use a different size.

Now for the program. You need to revert to the initial program given in the second article of this tutorial, and alter the first few lines of the

DrawField procedure to read as follows:

```
LOCAL currid%,id%,gridX%,gridY%
currid%=gIDENTITY
id%=gLOADBIT(NUM$(gWIDTH,3)+
  ↵ "x"+NUM$(gHEIGHT,3)+".mbm")
rockSize%=gWIDTH
gUSE currid%
gridWidth%=16
gridHeight%=8
```

We're back to the 16x8 playing field, as discussed already. But this time, instead of loading the same *Rock.mbm* image on each machine, the program checks to see what size of screen we're running on, and loads the corresponding bitmap. On all machines except the Geofox, there is a bit of screen to the right or below the playing field. You can fill this with a pattern as in an earlier example, or perhaps it could be used for scoring.

Each of the three methods discussed during this tutorial has its place. The first one, of keeping the display small, is probably satisfactory only for simple projects. The second, of implementing a large playing field and showing only a portion at a time, is recommended for games where the playing area is always going to be larger than the screen. The third method is useful for large and small games, but requires a lot more drawing. You have the basis here for a simple game, if you add the techniques from the *Animating OPL* and *Taking Control* tutorials. Feel free to experiment!

