

Spider for Windows

Abstract: Spider: According to *Hoyle's Rules of Game*, Spider is "considered to be the best of solitaires for it gives greatest opportunity to overcome the luck of the deal by skillful play." This computer version is written in Microsoft's Visual Basic and requires VBRUN100.DLL and works only with Windows 3.0 or higher. While it can run on a monochrome monitor, it works best with a color monitor. It requires a mouse. It offers two versions of the basic game and allows some "cheating" to facilitate play.

Foreword: I wrote this version of the solitaire game **Spider** as an exercise to test the flexibility of Microsoft's Visual Basic. Having done most of my previous programming in C or Pascal, I was impressed with the ease with which I could program this card game for the Windows 3.0 environment. There are limitations to Visual Basic and pitfalls in its implementation, but on the whole this development system is quite impressive.

Spider uses two decks dealt into ten columns with a reserve of fifty (or forty) cards to be dealt to the columns in subsequent play. The object of the game is to move cards between columns so as to create a sequence of cards in the same color (suit) from King down to Ace. When such a sequence is created, it may be removed from the field of play. When all eight sequences are removed, the game has been won. This implementation uses four *colors* rather than four suits (although players with monochrome monitors can use the suit option to have cards display the four suits in a particularly hideous fashion--use the Options choice under the Game menu item) to distinguish cards. For those with a color monitor, this makes the game actually easier than using "real" cards.

Standard Game: After shuffling, forty-four (44) cards are dealt face down in ten (10) piles--four cards in each pile and an additional one card on each of the first four piles. Another ten (10) cards are dealt face up on the ten piles. Another fifty cards are reserved in the deal pile for later play. Cards may be moved from column to column if the card moved is one less in rank to the card to which it is moved. For example a green 10 may be moved to a yellow Jack (J). To move a card, place the *tip* of the mouse pointer on the card, press down the left mouse button and hold the button until the *tip* of the mouse pointer (and the card "square") has been moved on to the target card; when the *tip* of the mouse pointer is on the target card, release the left mouse button. A sequence of cards may be moved as a unit if all the cards in the sequence are of the same color (suit). Any card (or sequence of cards of the same color [suit]) may be moved into a blank [white] space. Once all moves have been made and all blank columns are filled, the player must click on the control button labeled "Deal" to deal another ten cards. These cards are dealt, one each, on each of the ten columns. In the standard game, there are five deals before the reserve is exhausted.

"Easier" Game: Under the Game menu item, the player may chose the Options of having only Three cards dealt face down (plus an additional four cards in the first four columns). This *may* be an easier game since it allows blank spaces to be opened more easily. It also gives the player one more round of random cards to be played to each column. This may actually make the game more complicated. Each player may decide which option is "easier." In the "easier" game, there are six deals before the reserve is exhausted.

Peeking: Above each column there is a "peek box." This box indicates the number of face-down cards under each column. If the player places the mouse on the peek box and presses down the left button, the box will show the next face-down card in the column. Technically, this is cheating. It does make play more rational, so take your choice: cheat or trust to chance.

Undo: Under the Game menu choice is Undo. When chosen, this command

"undoes" (ugh) the previous move *providing* that the previous move did not turn up a previously face-down card or the previous move was not a deal. To undo either of the disallowed moves would allow the player to "cheat." This is, of course, inconsistent with the peek boxes, which are a form of "cheating." Programmers may remove this inconsistency if they wish.

Scrunching: The menu item Scrunching allows the player to handle long columns on a short monitor. With 104 cards in play, it is possible to create long columns--longer than most monitors can display. Since Visual Basic does not allow the user to scroll the form (as far as I can tell), I have overcome this handicap by giving players the option of "scrunching" their columns. The "One Quarter" choice overlaps cards by one-quarter (seventy-five percent of each card remains visible). The "One Third" choice overlaps cards by a third. The "One Half" choice--things are getting hurray on a EGA monitor--overlaps cards by a half. "Scrunching" is a kludge. If someone out there can develop a better system, I would appreciate receiving a copy of the code.

Remove: The Remove button allows the player to remove a completed sequence. This removal is not done automatically because players may, in some cases, wish to use parts of a completed sequence to rearrange other columns. I have never wanted to do this, but my wife, an avid Spider player, assures me that this is vital.

Responsibility: This code is release to the public domain. I had fun writing the program and hope that you have fun using it. I take no responsibility for time lost playing this addictive game (or any other responsibility for this program). I would appreciate it, however, if people who chose to modify the code share their modifications with me. I am curious how one might overcome the limitations I encountered with Visual Basic.

Mark U. Edwards
Channel One in Cambridge, Massachusetts
CompuServe 70323,1451