

# WordsWorth v2.0

## The 'Ultimate' CrossWord Game

by Graham Wheeler  
(c) 1992--1997 All Rights Reserved

### Contents

[Introduction](#)  
[Game Overview](#)  
[Playing the Game](#)  
[The Game Buttons](#)  
[Invalid Moves](#)  
[Board Edit Mode](#)  
[The File Menu](#)  
[The Options Menu](#)  
[The Tools Menu](#)  
[Autoplay Level Options](#)  
[Dictionary Consultations](#)  
[Concept Keys](#)



## **Introduction**

Welcome

Variations on a Scrabble Theme

WordsWorth Players



## **Game Overview**

The rules of the game are quite straightforward. They are described in their loosest form here, as they vary somewhat depending on the configuration. The scoring method is explained in stages in the following sections.

[The Objective](#)

[The Board](#)

[Hot Squares](#)

[The Tile Pool](#)

[Making a Move](#)

[Blank Tiles](#)

[The First Move](#)

[Subsequent Moves](#)



## **Playing the Game**

This section gives a brief overview of how to use WordsWorth, intended to get you up and playing as soon as possible.

[Starting a Game](#)

[What you see on the Screen](#)

[Making WordsWorth Play First](#)

[Playing the First Move Yourself](#)

[Passing](#)





## The Game Buttons

The full set of panel buttons and actions are:

- *Pass*, which tells WordsWorth that you want to pass this round. You will be given a chance to discard tiles; click on the 'Done' button to complete the move.
- *Done*, which tells WordsWorth that you have finished your move. The move will be checked for validity, and if it is admissible, the next player will be given a chance to play. If there is a problem with the word, a description will be shown on the status line, and you will have to try again.
- *Auto*, which tells WordsWorth to play the current move.
- *Next*, which tells WordsWorth to play the current move, or, if a 'Next' or 'Auto' has already been done, to undo the last move and play it differently (WordsWorth stores up to twenty possible moves which it chooses from. If you keep clicking on 'Next' it will eventually cycle back to the original move).
- *Run*, which tells WordsWorth to make all moves for the current player.
- *Stop*, which tells WordsWorth to stop autoplating moves for any players for which you have clicked on 'Run'.
- *Undo*, which undoes the last move. You can undo multiple moves all the way to the start of the game.
- *Redo*, which tells WordsWorth to redo the last undone move. You can redo multiple moves. Note that if you undo one or more moves, and then play a different move, you will not be able to redo the undone moves.

If you specified at the start of the game that cheating was not allowed then the 'Auto', 'Next' and 'Undo' buttons will have no effect during that game.



## Invalid Moves

If you enter an invalid move, WordsWorth will not accept it when you click on 'Done'. Instead, an error message will appear in the status area. The possible errors are:

**1st word must cross row <n>** You are playing the first word on the board and you have to play it according to constraints which you have violated. These constraints are specified in the configuration.

**1st word must cross col <n>** You are playing the first word on the board and you have to play it according to constraints which you have violated. These constraints are specified in the configuration.

**Word has no anchor square!** The word does not attach to the existing word structure on the board. You must build on to what is already there (but see below).

**Word conflicts with board!** You entered a word which contains a letter which conflicts with a letter already on the board.

**You don't have a blank!** The word you entered requires that you have a blank tile, which you don't.

**You don't have a <char>** The word you entered requires that you have a <char>tile, which you don't.

**Bad Xword** The word you entered does not form all valid cross words.

**Can't start there!** The word you entered must have its first letter preceded by an empty square or the edge of the board, not another letter.

**Can't end there!** The word you entered must have its last letter followed by an empty square or the edge of the board, not another letter.

If one of these errors occurs, you will simply have to play a different move. You will be given the option of overriding WordsWorth's dictionary lookups of the main word and any cross words, in case you wish to play a move that is not in the dictionary.



## Board Edit Mode

WordsWorth is always in one of two 'modes': game mode, or edit mode. Game mode (the default) is used for playing games, while edit mode is used for creating new board types for games.

The menu bar at the top is different for each mode, as is the panel on the right. In game mode, the panel contains the play buttons, the round and move number, the number of tiles left in the pool, and the player scores, while in edit mode the panel is empty.

To enter edit mode, click on 'Board Layouts' in the Options menu in game mode.

You will then be presented with a dialog box which allows you to specify which board you are editing, its size, and any constraints on the first move. When you have set the values appropriately and click on 'OK', the edit board will appear. You can then click on any square on the board; a dialog box will appear which allows you to set the attributes for that board square. To return to game mode, select 'Save and Exit' or 'Exit without Saving' (whichever is appropriate) from the File menu.



## The File Menu

When in game mode, the File menu has the following options:

**New Game** allows you to start a new game. If a game is already being played, you will be asked whether you want to discard it. You will then be presented with a dialog box which allows you to choose the game type, the number of players, and whether or not cheating is allowed (if you disallow cheating, several WordsWorth options will be disabled; this is intended for multi-player games where there are two or more human players participating).

**Load Game** allows you to restore a saved game and continue playing.

**Save Game** allows you to save the current game being played to disk.

**Quit** exits WordsWorth.





## The Options Menu

The Options menu allows you to design new game types, and change some options that affect game play.

In WordsWorth, a 'game type' consists of:

- o a board type;
- o a tile pool type;
- o a set of rules.

Board types and tile pool types are defined separately, so that they can be reused (for example, we may wish to create several different game types which all use the same board and pool types, and differ only in the rules).

A board type has a name, a size (in rows and columns), a layout, and a set of restrictions on allowed opening moves.

A tile pool type has a name, and the number and score of each of the 27 different tiles.

From the Options menu, you can create new game types, new board types, and new pool types. You can also editing existing games, boards and pools. In addition, you can change how tiles are displayed on the screen, how WordsWorth will choose automoves for each of the (up to) four players, change some settings affecting whether tiles are exposed or not, and so on. The options are:

**Misc** pops up a dialog box allowing you to set the default number of players (used to initialise the dialog that pops up when you select 'New Game'), and the path of the dictionary file. You can also enable or disable three options:

**Hide racks between moves** If this option is selected, then players will have to click on their racks to see the tiles. This is for when there are multiple human players, and ensures that when a player completes their move they don't get to see the next player's tiles (at least not without the other player knowing that they've seen them!).

**Show auto-player racks** If this option is enabled, you will be able to see the contents of the rack when a computer opponent is making a move.

**Use different colours for players** This option draws each players tiles in a different colour. It is automatically enabled in territorial play games.

Note that changes made to these settings take effect immediately. On the other hand, changes to the configuration of a game, board, or pool, will only have an effect from when the next game is started (so you can't change the layout of the board while a game is in progress).

**Board** lets you edit an existing board type, or create a new board type. You will be presented with a dialog box which allows you to choose which board type you want to edit. The dialog box also allows you to set or change the name of the board type, the size of the board, and the constraints on the first move.

To create a new board type, you should click on the 'Board' pull-down list box and select the 'Create New' board type.

Once you have set the values in the dialog box to your satisfaction, click on the OK button. WordsWorth will switch into Edit mode.

**Pool Size** lets you create a new pool type, or edit the name and/or letter counts of existing pool types. A dialog box will appear which has 27 sliders for setting the tile counts. You can choose which pool you want to edit from the list box at the top left of the dialog; select 'Create New' to create a new pool.

Note that you must click on 'Save Settings' for the results to be saved. Clicking on 'Done' exits the dialog box but does not save any changes that have been made.

**Letter Scores** is almost the same as the Pool Size option, except it allows you to set the scores of the letter tiles, rather than the number of each tile.

### **Letter Weights**

**Level** lets you change the settings WordsWorth uses to choose automoves for each player.

**Game** allows you to create new game types or edit existing game types. Once again you can choose the game type to edit from a list box, and select 'Create New' to create a new game type. The options that you can set for a particular game type are:

- o the name of the game;
- o the name of the board type to use;
- o the name of the pool type to use (may be unnecessary);
- o how new words must be played - whether the game is crossword-like, territorial, or free-form;
- o whether the player is playing from a rack, directly from the pool, or whether any letter can be played at any stage (in the last case you need not specify a pool type name above);
- o the number of tiles in the rack; this is only relevant if playing from a tile rack;
- o the number of tiles that must be played to get a score bonus, and the number of bonus points to award when that many tiles are played.
- o whether the player can choose which tiles they draw after playing a word; if not selected tiles are drawn randomly from the pool. Again, this option only has an effect if tiles are to be played from a rack, as in the other two cases players do not draw tiles.



## The Tools Menu

The Tools menu has two commands:

**Show Pool** pops up a window showing the current set of tiles that are left in the tile pool.

**Consult** allows you to perform dictionary consultations.



## Autoplay Level Options

You can set the autoplay levels for each player from the 'Level' item in the 'Edit' menu. A dialog box will appear with a number of options:

**Player** Use this listbox to select which player's settings you wish to change;

**Use adaptive strategy** If you select this option, the player's strategy will change depending on how well it is doing with respect to the other players.

**Avoid opening up hot squares** If you select this option, the autoplayer will avoid making moves which make it easy for the next player to score double/triple/etc word scores. This does tend to make the playing style more cramped, but this can be compensated for by increasing the weighting on word length.

**Use letter weights** If this option is selected, then the letter weights will be used as well as the score in choosing a move (if it is not selected, letter weights are only used when an autoplayer passes a move and discards tiles). This can be used, for example, to make an autoplayer tend to play some letters as soon as possible (such as a Q, for example), while tending to hold on to other letters (such as blanks and S's). The rationale is both to do with penalties for unplayed tiles at the end of the game, and the idea that blanks and S's can often be best used in high-scoring plays (where the score will compensate for the weight).  
If this all sounds like greek to you, don't worry - XWord by default comes with useable weight settings. You can turn this option on if you like, just to see the effect, but it isn't too important except for those interested in experimenting with different strategies.  
Note that this option is sometimes (but not always) used in Adaptive Play.

**Minimum allowed score** This lets you set a lower limit on the acceptable score for an autoplayer's moves. Be warned that if you set this too high the player will tend to pass more often than play a word.

**Maximum allowed score** This lets you set a ceiling on the score that an autoplayer can get with a move. It gives human players an unfair advantage, but if you're not a great player you may appreciate being able to do this.

**Minimum score to play blank** This option is similar to the letter weights in intent. It is used to cause an autoplayer to hold back on using a blank tile until it can use it in a high-scoring play.

**Minimum allowed length** This option lets you specify the minimum allowed length of autoplayer moves.

**Maximum allowed length** This option lets you specify the maximum allowed length of autoplayer moves.

**Word length weight** This option lets you specify how much importance an autoplayer will attach to the length of the word it plays. Essentially this is used to modify the score by an amount equal to the product of the value you specify and the length of the word. For example, if you set this to two, and WordsWorth finds a two-letter move with score 10, and a five-letter word with score 6, it will choose the five letter word as it considers it to have a score of  $6+2*5=16$  while the other word it considers to score  $10+2*2=14$ .

**Maximum new cross words** WordsWorth has an infuriating habit of playing several adjacent words in the same direction, forming lots of two letter and then three letter cross words (it can do this as it has so many two and three letter words in its dictionary). This can make it quite hard for

a human player at times. This option lets you limit the number of cross words that it can make per move. Setting this to one or two makes the autoplayer play a more 'open' game. I find this a very useful option.

**Dictionary filter factor** This is a real technical one to explain. Suffice it to say that it allows you to force WordsWorth to only look at some of the words in the dictionary. The higher you set it, the less words it will check. The actual choice of words is random, but will tend to favour shorter words. While not quite accurate, you can think of the value as being the percentage of words that are skipped.

Generally I don't advise the use of this option, unless you are playing a free-form game. In that case XWord can take a long time to play as there are so many possible moves. Setting the dictionary filter factor can be used to cause significant speed ups (but of course the autoplayer will not play that well).

**Anchor filter factor** This is another option to speed up play, at the cost of weakening the autoplayer. When used moderately, it is less drastic than the dictionary filter factor. The concept of an anchor square was described earlier. The anchor filter factor is the percentage of anchors that should not be considered for play. Again, in free-form play this can cause significant speedups. Setting it to ninety should result in a ten times speedup; setting it to ninety-five should cause a twenty times speedup, and so on (put another way, setting it to ninety-five means only five percent, or one-twentieth, of the anchor squares will be considered).

I don't recommend setting this to much higher than fifty, except in the case of free-form play.

Setting any of the sliders to zero causes that option to have no effect.

After changing the settings for a player, click on 'Save Player'. When you have finished making all your changes, click on 'Done'. Note that clicking on 'Done' does not itself result in any changes being saved; you must click on 'Save Settings' for that.





## Dictionary Consultations

The dialog box for dictionary consultations lets you generate lists of anagrams and words, given a set of letters. The set of letters you specify can include blank tiles, as well as things like 'any vowel' or 'any non-vowel'.

The method used is fairly complex but worth understanding if you want to make full use of it. Underlying each consultation are two pieces of information:

- a word template which specifies which letters are allowed in which positions in matching words;
- a letter pool, which specifies which letters are available for use in unconstrained entries in the word template.

[Some Examples](#)

[Consultation Match Types](#)

[Specifying Consultation Patterns](#)

[Special Pattern Characters](#)

[Letter Ranges](#)

[Consultation Background Processing](#)

[A Tip for Multi-Word Anagrams](#)

[Word Length and Count Constraints](#)



## Concept Keys

A new feature in version 2.0 is the use of 'concept keys'. In WordsWorth you can use these when doing dictionary consultations; in XWord you can do this as well as associate concept keys with words in the crossword grid.

Concept keys are essentially semantic categorisations of words, somewhat like a thesaurus. The file containing the keys is named "thes.dat" for this reason. This file contains semantic associations between nearly 25,000 words. When you specify a concept key, the dictionary will be constrained to match only words with semantic associations to the key. These semantic associations can be quite loose, as they are based on transitive relations. That is, a word A is considered to be potentially related to a concept key C if there is a word B such that A and B are related in meaning, and so are B and C. Furthermore, the way that concept key restrictions are implemented in WordsWorth/XWord is not infallible - extra words can be admitted that are not actually related, although the actual number of erroneous matches should be a very small percentage. This fallibility is a tradeoff; the algorithm used sacrifices complete accuracy for speed. Concept key restrictions have a very low impact on the performance of dictionary consultations (although somewhat more of an impact in XWord grids).

You can specify more than one key, separated by commas. Keys must be single words. If more than one key is listed, the intersection is used, but you can change this to a union by starting the list with "+". If there are no words satisfying the concept key(s), they are ignored. This is enforced more strongly in the XWord grid, where the word lengths are also used; i.e. if you specify concept keys for a six letter word in the grid, and there are no six letter words matching the keys, then the keys are ignored.

To illustrate the use of concept keys, here is an example. Say you have a six-letter word for which the clue is "A swamp". You know that the second letter is "O". Then, either by entering it on the grid via the word settings dialog, or in dictionary consultations, you can specify the pattern ":O::::" and the concept key "swamp" to restrict the search. This will result in the following matches:

BOLDLY  
FOXILY  
MORASS  
NORMAL  
TOPPLE

If, on the other hand, you had specified the concept key "bog", then the following words would be matched:

COARSE  
COBWEB  
HOOKER  
HOOPLA  
LOTTED  
MODERN  
MOLDER  
MORASS  
MOTHER  
ROTTEN  
SOILED  
TOILET

The combination of the two ("swamp,bog") gives the best results, namely the intersection of these two lists which is the single (correct) word "MORASS".



## Welcome

Welcome to WordsWorth!

WordsWorth was inspired by an article in Communications of the ACM in May 1988, entitled 'The World's Fastest Scrabble Program'. I had always been a frustrated Scrabble player. I take ages to move, generally play pretty badly anyway, and dislike waiting for my opponent. As a result, I decided to write my own program, which would have both infinite patience and lightning response time ; - ) .

It soon became clear that the principles involved were not bound by the rules of Scrabble as such, but more by the concept of crosswords. Imposing rules simply implied restricting the set of possible moves.

The idea of a general-purpose crossword game playing engine was born!

Please send your comments, bug reports, flames, and anything else to me at [gram@oms.co.za](mailto:gram@oms.co.za). I have tried to make WordsWorth as generic and general as possible; if you have suggestions, particularly about further generalisation, let me know! I am also interested in hearing about erroneous words or omissions in the dictionary.

If you have word lists in other languages that you would like made into WordsWorth dictionaries, send them to me and I will do it for you (e-mail only please). The dictionary builder runs only under UNIX, takes many hours to run, and requires about 16Mb of RAM, so it is not distributed with WordsWorth.

The last public release of WordsWorth before this was v1.3. It and all prior versions were MS-DOS-based. WordsWorth v2.0 is the first version to run under both MS-Windows and UNIX. MS-DOS is no longer supported. The dual platform nature of WordsWorth v2.0 has been made possible through the use of wxWindows, a cross-platform C++ application framework.

A few features of v1.3 are not available in v2.0. These are:

- the hint facility is no longer present. The same effect can be achieved with a combination of 'auto', 'next', and 'undo'.
- the ability to exclude words from the dictionary has been removed. The dictionary now has far fewer bad words than in versions earlier than v1.3, and the performance penalty of using an exclusion dictionary is no longer worth paying. If WordsWorth plays an invalid word, use the 'next' button to force it to play a different move.
- blank tile exchanges are no longer supported.

Some new features have been added:

- the program is now almost entirely mouse driven. Only in crossword solving mode is keyboard input necessary.
- WordsWorth now supports from one to four players; previous versions were fixed at two.
- more level control options have been introduced.
- you can now view the contents of the tile pool at any stage.
- all of the configureable options can now be set from within the program.
- a new option allows you to explicitly choose which tiles you want to draw at the end of a move (computer players still use random drawing only).

DISCLAIMER - This software is distributed without a warranty of any sort. You use this software at entirely your own risk. I will not be responsible for any loss, damage, injury, death, disaster, plagues, famine, divorces, disagreements, etc that may result from the use of this software.

## Variations on a Scrabble Theme

WordsWorth allows you to design your own crossword games. You can create any size and shape board up to a 21x21 square, and pepper it with black crossword squares that can't be played upon. You can make any square a single, double, triple, quadruple, etc letter or word score type. You can play free-form on the board, or be forced to build on to the existing structure (as in Scrabble). You can play a territorial game in which you have to build on to your own structure. You can specify what letters are available, how much they are worth, and give them a strategic 'weight'. You can save and restore games, and undo and redo all the moves. If you like Scrabble, you can use WordsWorth as a Scrabble opponent (although the dictionary is not exactly the same as the OSW).

## **WordsWorth Players**

WordsWorth supports from one to four human or computer players, and provides several options for assisting in multi-player human games. You can control the level and style of your computer opponent(s). A future version of WordsWorth will allow multi-player games over the Internet.

## The Objective

The basic aim is to make words in a crossword-like fashion from a set of letter tiles which you draw out of a common pool. You alternate plays with the program, and when the pool is exhausted and one player has no more tiles, or both players are stuck, the game ends. You score according to the word you make, as well as other new words that may result from your play. For example, if the board contains:

DOG

and you play the word STOP thus:

DOGS

T

O

P

you get score for both DOGS and STOP. On the other hand, you *must* make valid words with your play, so

S

DOGST

O

P

is not a valid way to play the word STOP, as DOGST is invalid.

Hyphenated words and proper nouns (capitalised words) are not allowed. The rules of Scrabble now admit the American '-ize' as well as the English '-ise' spelling of words, and WordsWorth follows this policy.

At the end of the game, the scores are compared to see who wins. The scores are calculated according to the location of the word (that is, what special squares did it cover), and the letters played, each different letter having a different score value. If either or both players still hold unplayed tiles, the score of those tiles is deducted to reach a final score.

Usually, you must build on to the existing words on the board (the exception being the first move, of course). WordsWorth offers some flexibility in these matters, however.



## The Board

Play takes place on a board which can be any size that will fit within a  $21 \times 21$  square area. If you want a rectangular board you specify the dimensions; if you want some other shape board you specify the size of a rectangle that can contain it, and use black squares to create the shape you desire.

## Hot Squares

Apart from black squares, the scoring of plays is affected by the type of the squares on which the play takes place. Thus, a square may be a 'double letter square', 'triple letter square', and so on, and similarly for 'double word', etc. We will call such squares 'hot squares'. If your move results in one or more hot squares being covered, you take this into account when working out the score for the move. Hot squares that affect letter scores are taken into account before those that affect the whole word.

Thus, if you play a two-letter word in which one letter has score 1, and the other score 4, and the first letter is played on a double word square while the second is played on a triple letter square, then the score for that word is  $2 \times (3 \times 4 + 1) = 26$ . Note that this is just for that word; the score for that move will usually involve score for one or more other 'cross words' as well. Once a hot square is covered with a tile, its type ceases to have any effect on scores; thus, only when you actually place a tile on a hot square does it have an effect on the score (once again, WordsWorth allows this to be changed).

## **The Tile Pool**

Each player draws tiles out of a common pool. Usually this is fixed in size, and each player draws a fixed number of tiles randomly from the pool at the start of each game, replenishing their tile rack at the end of each move. In Scrabble, for example, each player holds seven tiles at a time.

WordsWorth also allows you to play any tile from the pool without drawing a small set of tiles each move. The pool can also be unlimited, allowing you to play any letter you like at any time, until all players get stuck anyway.

## **Making a Move**

The basic idea behind making moves was described earlier. There are also some other types of moves. For example, if you cannot make a good word, you can pass your turn. If you do so, you have the option of returning any or all of your current tiles to the pool, and drawing replacements (assuming you are drawing tiles, and not playing freely from the pool as described in the previous section).

## Blank Tiles

There is a special tile that plays a similar role to a joker in cards. This tile is blank, and can be used to represent any character. If you have such a tile and play it, you must state what character it is representing. When such a tile is on the board the letter it represents is shown in red; other tiles are displayed using black text. If you are passing your turn and wish to discard a blank (a rare event!), you must enter it as an underscore '\_', as it does not represent any particular character (other than itself) when being discarded.

## **The First Move**

The first move may optionally have some constraints placed on how it should be played. For example, it could be specified that the word must be played in the third row, or that it must cross the center square, or that it may be played anywhere at all. When using a square board, these constraints would usually stipulate that the first word may be played in either direction and must pass through the center square on the board (assuming the board has an odd-numbered size). Alternative constraints are most useful when a board is being used which is not square, or which has some unorthodox layout built using black squares.

## Subsequent Moves

Usually each move other than the first move must build on to the existing words like a crossword. Other possibilities are territorial play (where each player builds on their own words), and free-form play, in which play may occur anywhere. In the former case, attachment to the opponents words is allowed, but every play must include a word with a previously- placed letter that was placed by the same player.

A more precise description, useful in understanding some of WordsWorth's level-control options, is that every word played must include at least one 'anchor square'. Exactly what constitutes an anchor square depends on the type of play. For normal, Scrabble-like play, any square which contains a played tile is an anchor square. For territorial play, any square that contains a tile played by the current player is an anchor square. For freeform play, all board squares are anchor squares.

## Starting a Game

To start playing a game, you must select 'New Game' from the 'File' menu. A dialog box will pop up, asking you to choose the type of game, the number of players, and whether cheating is allowed. When you are happy with the selections, you should click on the 'OK' button.

The board will then be drawn, with the first player's tile rack being shown below. The tiles may have their faces exposed, or may all appear blank; the latter is intended to ensure privacy when there are multiple human players; clicking on any the tiles will expose them to view.



## **What you see on the Screen**

To the right of the board is a panel containing several buttons. For now, we will concern ourselves with only three of these: 'Pass', 'Done', and 'Run'.

At the bottom of the WordsWorth window is a status line. At this stage it should display a message saying either 'Player 1 get ready' (if the tiles are hidden), or 'Player 1 make your move' (if they are exposed).

## **Making WordsWorth Play First**

If you want the role of player 1 to be adopted by the program, you should click on the 'Run' button. WordsWorth will make the first move, and all subsequent moves for the first player. On the other hand, if you click on the 'Auto' button, WordsWorth will autoplay a single move.

## Playing the First Move Yourself

To play the move yourself, you need to click on each tile in the rack that you want to play, followed by the location on the board where you want it to be played. As you click on a rack tile, it will turn red to show that it is selected; when you click on the board location it will turn grey, and a copy (also grey) will be placed on the board. You can also select a tile from the rack by pressing the corresponding key.

If you click on the wrong location on the board, just click on the right one, and the letter will be moved. To undo a letter placement, click on the grey letter on the board (if you are playing from a rack, as opposed to playing from the tile pool or an unlimited pool, you can also click on the grey letter in the rack to undo; in the other cases clicking on the rack letter will instead select a second copy of the tile to play). With a little bit of experimentation you'll quickly get the hang of it.

Once you have placed all the tiles, you should click on the 'Done' button to tell WordsWorth the move is done. The move will be checked, and if acceptable the next player will be prompted to move. If there is a problem an error message will appear on the status line and you will have to try again.

If the word that you play (or a new cross word) is not in the dictionary, you will be given the option of overriding the dictionary and forcing WordsWorth to accept the move.

## Passing

If you want to pass instead of playing a word, click on the 'Pass' button. A message will appear on the status line asking you to select what tiles you wish to discard. Click on the tiles, and then on 'Done'.

If you want to save your game you can do so by selecting 'Save Game' from the File menu; use 'Load Game' to restore a saved game.

## Some Examples

A few examples can be used to illustrate the idea:

- for the literal word ROPE the word template is "ROPE" and the letter pool is empty;
- for anagrams of ROPE the word template is unconstrained (we will write this as "####") and the letter pool consists of the letters R, O, P and E;
- for anagrams of ROPE that begin with R and have P in the third place the word template is "R##P#" and the letter pool is "O,E".

Given a word template and letter pool, WordsWorth tries to find matches by plugging different letters from the letter pool in to '#' positions in the word template.

## Consultation Match Types

In the consult dialog, the first choice you have is the type of match to use. There are four possibilities:

**Use all letters** Matches must match the whole word template;

**Match prefixes** Matches may match any leading subsection of the word template;

**Match suffixes** Matches may match any trailing subsection of the word template;

**Multi-word anagrams** This is a special case. The whole word template must be matched but the match may consist of multiple words.

The first and last case should be clear; the prefix and suffix matches can be illustrated by continuing with our last example. That is, we have word template "R#P#" and letter pool "O,E". Using prefix matching, WordsWorth would find words that match any of the following situations:

Word Template	Letter Pool
R	OE
R#	OE
R#P	OE
R#P#	OE

Using suffix matching, WordsWorth would instead use:

Word Template	Letter Pool
#	OE
P#	OE
#P#	OE
R#P#	OE

Thus in the first case, WordsWorth will see if any of the following are valid words:

R RO RE ROP REP ROPE REPO

In the suffix case, WordsWorth will check the following:

O E PO PE OPE EPO ROPE REPO

Note that in the case that the word template is unconstrained (that is, all 's), then prefix and suffix matching are identical, both having the meaning "words consisting of one or more letters from...". Reread the last sentence if you didn't understand it.

## Specifying Consultation Patterns

When making consultations, you don't explicitly specify the word template and letter pool. Rather, you specify a single pattern. In the simplest case a pattern is a sequence of letters (whitespace may be present but is ignored). The letters may be either upper case or lower case, the former meaning that the letter must occur in that position (i.e. is a word template constraint), while lower case means the letter need not be in that position (i.e. it is in the letter pool). Thus our example above would be specified as "RoPe" (or equivalently as "RePo"). A few more examples should help here:

Pattern	Word Template	Letter Pool
informatION	#####ION	informat
MISinformed	MIS#####	informed
Treated	T#####D	reate

The first two are typical examples of patterns that would be used for suffix and prefix matching, respectively.

## Special Pattern Characters

Patterns can also have some special characters. Each has an 'upper case' and a 'lower case' form, analogous to letters. The special characters are:

Meaning	Upper Case	Lower Case
Any letter	:	.
Any vowel	*	+
Any consonant	=	-



## Letter Ranges

Patterns can also have 'letter ranges' enclosed in square brackets, such as '[A-Z]' (any letter), '[AEIOU]' (any one vowel) or '[A-MZ]' (any one of A through M, or Z). To specify the opposite of a range, start the range with an exclamation mark or caret. For example, '[!AEIOU]' matches any non-vowel. The letter case of the first letter in the range determines whether the range will be used as a word template constraint or as a letter pool entry; other than this letter case is ignored. Thus, for example, "[a-Z]" is legitimate and equivalent to "[a-z]". Similarly "[A-z]" is equivalent to "[A-Z]". You cannot use the other special characters within a range (thus for example, "[\*y]" is not valid).

## **Consultation Background Processing**

WordsWorth does dictionary consultations in the background, to give you the opportunity to interrupt, or to continue playing a game while a consultation is in process. You can interrupt a consultation by starting a new one, or by clicking on the 'Stop' button in the panel. The matches WordsWorth has found up to that point will then be displayed.

## A Tip for Multi-Word Anagrams

Quite often there will be thousands of possible anagrams and it becomes impossible to search out the good ones. There is a technique to help in this case - look through some of the possibilities, and select a word you think is appropriate in the final anagram. Then subtract this word from the input, and see what words can be made from the remaining letters. For example, consider VEGETARIAN. We see that some anagrams have the word RAVE, and decide to use this. This means we must find anagrams of the remaining letters, namely GETIAN. We find the word EATING. Thus VEGETARIAN=EATING RAVE.

## **Word Length and Count Constraints**

You can specify minimum and maximum word lengths for prefix, suffix and multi-word anagram matches. For the latter you can also specify the minimum and maximum number of words allowed per match. If these are set to zero (the default) they will have no effect. They can be useful when generating multi-word anagrams with patterns that contain several A's or I's, as WordsWorth will likely churn out thousands of matches in these cases starting with the words 'A', 'AA', and/or 'I'. Setting the minimum length to three (for example) can help prevent this.







