

WormWars

James R. Jacobs

COLLABORATORS

	<i>TITLE :</i> WormWars		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	James R. Jacobs	August 24, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	WormWars	1
1.1	Worm Wars	1
1.2	Overview	1
1.3	New Features	2
1.4	Usage	2
1.5	System Requirements	2
1.6	Startup	2
1.7	Workbench ToolTypes	3
1.8	CLI Arguments	3
1.9	Title Screen	5
1.10	Field Editor	6
1.11	Hints for Creating Fieldsets	9
1.12	Other Information	9
1.13	Contact Details	9
1.14	Amiga Development System	10
1.15	Source Code	11
1.16	History	11
1.17	Other Software	13
1.18	How to Play	13
1.19	Creatures	14
1.20	Controls	15
1.21	Squares	17
1.22	Numbers	18
1.23	Objects	18
1.24	Affixer	20
1.25	Ammo	20
1.26	Armour	21
1.27	Bias	21
1.28	Bomb	21
1.29	Bonus	22

1.30 Brakes	22
1.31 Clock	22
1.32 Converter	22
1.33 Cutter	22
1.34 Cyclone	23
1.35 Encloser	23
1.36 Glow	23
1.37 Grower	23
1.38 Healer	24
1.39 Ice	24
1.40 Life	24
1.41 Lightning	24
1.42 Magnet	25
1.43 Missile	25
1.44 Multiplier	25
1.45 Powerup	26
1.46 Protector	26
1.47 Pulse	27
1.48 Pusher	27
1.49 Remnants	27
1.50 Sideshot	27
1.51 Slayer	28
1.52 Slower	28
1.53 Switcher	28
1.54 Treasure	28
1.55 Umbrella	29
1.56 Strategy	29
1.57 Worms	30
1.58 Orbs	32
1.59 Otter	33
1.60 Giraffe	33
1.61 Goat	33
1.62 Lion	33
1.63 Slime	33
1.64 Drips	34
1.65 Fragments	34
1.66 Penguin	34
1.67 Whirlwind	34
1.68 Dog	35

1.69 Bird	35
1.70 Cloud	35
1.71 Teleport	35
1.72 Timebomb	35
1.73 Octopus	36
1.74 Fish	36
1.75 Scoring	36

Chapter 1

WormWars

1.1 Worm Wars

```

                                     #*=====*#
#|   W O R M   W A R S   |#
#|     for the Amiga     |#
#|   Version 7.21a      |#
#|   Mon 21 Oct 2002    |#
#|                       |#
#|   by James R. Jacobs |#
#*=====*#
```

Overview

New Features

Usage

Other Information

Title Screen

Field Editor

How to Play

1.2 Overview

Worm Wars is an arcade game for up to 4 players, who travel around a series of rectangular maze leaving a deadly trail behind them, fighting 16 types of creature and collecting 32 varied object types.

The integral field editor allows you to load, edit and save user fieldsets, for greater lasting attraction. There is support for playing MED and IFF 8SVX files as music and sound effects respectively.

It is enjoyable either for one player, or for competitive multiplayer

games, and demo mode is available. Amiga or human control can be specified for any worm. Two keyboard players, four joystick players, and/or four CD32 gamepad players are supported. It is system-friendly and style compliant.

1.3 New Features

- . Fixed: corrupt graphics in deep screens.

1.4 Usage

System Requirements

Startup

1.5 System Requirements

Hardware	Required	about 512K free RAM
	Recommended	Colour monitor Mouse Joystick(s) CD32 controller(s) Flicker fixer about 1Mb free RAM Keyboard with numeric keypad Battery-backed clock Parallel port joystick adapter
Firmware	Required	Kickstart 2.0+
	Recommended	Kickstart 3.1+
Software	Required	OS2.04+
	Recommended	OS3.5+ MEDPlayer.library (included)

You need at least Installer 42.9 to run the installation script.

1.6 Startup

Workbench ToolTypes

CLI arguments

Startup

The default behaviour is to load the music and sound effects automatically at startup. The `-n` (NOPRELOAD) CLI argument prevents loading of the sound effects and music until they are needed.

If WormWars.font cannot be loaded at startup, Topaz 8 is used.

If the specified default fieldset cannot be loaded at startup, a blank, five-level fieldset is used instead, with a treasury floored with silver.

The user's preferred screen mode, initial worm controls, and settings for "Animations?", "Create Icons?", "Thick Tails?" and "Shuffle Levels?" are loaded from WormWars.config, if possible.

If the WormWars.config file does not exist, and you are running OS2.1+, you will be asked to select your preferred screen mode. At least 640x512 in 16 colours is required. There is no advantage in selecting a larger or deeper screen than this.

1.7 Workbench ToolTypes

The following can be specified in the game's .info file. All are ← optional.

TOOLPRI=<priority>	eg. TOOLPRI=5
FILE=<fieldset>	eg. FILE=foo.fset
GREEN=JOY GAMEPAD AMIGA NONE KYBD	eg. GREEN=GAMEPAD
RED=JOY GAMEPAD AMIGA NONE KYBD	
BLUE=JOY GAMEPAD AMIGA NONE	
YELLOW=JOY GAMEPAD AMIGA NONE	
NOPRELOAD	eg. NOPRELOAD
NOANIMS	
NOMUSIC	
NOFX	
NOICONS	
THICKTAILS	
SHUFFLE	
QUIET	

These are equivalent to the relevant CLI arguments

.

If you double-click on a fieldset file, the game will load with that fieldset. In that case, ToolTypes are of course, taken from the fieldset's .info file.

1.8 CLI Arguments

Command Information

WormWars

Format: WormWars [-p=<priority>] [-n|nopreload]
[-a|noanims] [-i|noicons] [-s|shuffle]


```

[green=JOY|GAMEPAD|AMIGA|NONE|KYBD]
[red=JOY|GAMEPAD|AMIGA|NONE|KYBD]
[blue=JOY|GAMEPAD|AMIGA|NONE]
[yellow=JOY|GAMEPAD|AMIGA|NONE] [-t|thicktails]
[[file=]<fieldset>]]

```

Template: WORMWARS -N=NOPRELOAD/S, -A=NOANIMS/S, -I=NOICONS/S,
-P=PRI/K/N, -S=SHUFFLE/S, -Q=QUIET, GREEN/K,
RED/K, BLUE/K, YELLOW/K, -T=THICK, FILE

Purpose: To run the Worm Wars game.

Specification:

-p: the priority that Worm Wars will be run at. Higher priorities can help the game to run closer to full speed. Lower priorities can help the system multitask, by giving less processor time to Worm Wars.

The allowable range is -128 to +5, with the default being 0.

-n: don't load sound effects and music until they are needed.

-a: turn off animations. This is not needed on 68000/68010-based systems, as animations are turned off by default on such systems.

-i: don't create icons when saving.

-s: shuffle levels.

-q: don't report errors during startup.

colour=KYBD|JOY|GAMEPAD|AMIGA|NONE:

the initial controls for the worms. For example, GREEN=KYBD will select keyboard control for the green worm. Unless otherwise specified, RED=KYBD, YELLOW=AMIGA, GREEN=NONE and BLUE=NONE. KYBD is not available for the blue and yellow worms.

-t: draw thick worm tails instead of thin ones.

<fieldset>: the pathname of your default fieldset. This is taken exactly as entered; an '.fset' extension is not automatically appended. If no argument is given, the default is PROGDIR:WormWars.fset.

?: displays the command line argument format.

The order of the arguments is interchangeable.

Examples:

```

WormWars /foo.fset          load foo.fset from parent
                             directory, and run at priority level 0

```

```

WormWars -f -p-3 bar.fset
    load bar.fset from current
    directory, and run at priority
    level -3, without loading sound effects

WormWars HD1:foo.fset -m -f -p4 -o
    load foo.fset from root
    directory of HD1:, and run at
    priority level +4, without loading music or
    sound effects, and set overhead viewpoint

WormWars -p+5 /fsets/bar.fset
    load bar.fset from fsets/
    sibling directory, and run at priority
    level +5

WormWars ?
    display command format

```

1.9 Title Screen

Keys

```

Esc F1 F2 F3 F4 .. .. .. .. ..
.... 1 2 3 4 . . . . . <- ... Hel . . . .
..... Q . . . . . O . . . Ret . . . .
... .. A S . F . H . . . . . ur . 4 . . .
Shift . . . . . N M . . ? Shift . . . 1 2 3 E
.. .. -----Spacebar----- Am .. -. . n

```

1-4 or F1-F4 cycles the relevant worm controls

through the various

control options. If Shift is held as a qualifier, it cycles backwards.

Return or Enter begins the game.

M toggles the music on or off.

F toggles the sound effects on or off.

S toggles level shuffling on or off.

Menu

Project New	Amiga-N	Creates a blank, 5-level fieldset titled WormWars.fset.
Project Open...	Amiga-O	Loads a fieldset.
Project Revert	Amiga-R	Reloads the current fieldset.
Project Save	Amiga-S	
Project Save As...	Amiga-A	
Project Delete...	Amiga-D	Deletes a fieldset.
Project Quit	Amiga-Q, Esc	Exits to CLI/Workbench.
Settings Animations?	Amiga-T	Animations on/off.
Settings Create Icons?	Amiga-I	Create icons when saving?
Help Creatures...		List of creatures.
Help Objects...		List of objects.
Help About...	Amiga-?, Help	Credits window.

Gadgets

Key	Colour	Human Controls
1	Green	Joystick 3/Gamepad 3/Left Keyboard
2	Red	Joystick 4/Gamepad 4/Right Keyboard
3	Blue	Joystick 2/Gamepad 2
4	Yellow	Joystick 1/Gamepad 1

Any combination of controls is acceptable, but at least one worm ↔ must exist.

1.10 Field Editor

The field editor allows you to load, edit and/or save the ↔ playfields

which the worms compete on.

You are allowed to have up to 50 levels, plus the bonus level, in each fieldset.

The high score table for each fieldset is saved with the fieldset. Making any modification to the fieldset will clear all the fieldset's high scores.

Hints

Keys

Esc	F1	F2	F3	F4	F5	F6	F7	F8										
....	1	2	3	4	5	6	7	8	9	0	Del	Hel	()	..	
.....	Q	.	E	R	T	.	U	I	O	Ret								
Ctrl	..	A	S	D	F	ur			U					
Shift	..	C	N	M	..	?	Shift					L	D	R				
Alt	..	----	Spacebar	----	Am	Al														

F1	empty
F2	silver
F3	gold
F4	dynamite
F5	wood
F6	stone
F7	metal
F8	frost

M toggles the music on or off.

F toggles the sound effect on or off.

Delete and Help change the current level (lower and higher, respectively). When shifted, they change to level 1 or the final level, respectively.

C centres the cursor in the centre of the field.

X and Y are symmetry functions. The cursor will move to the other side of the field, at the same distance from the other edge as it was originally from the original edge. This is useful for making symmetrical fields.

Numeric . stamps the current brush at the current cursor location.

Numeric 0 toggles 'sticky mode'. This means that the 'brush' is held down: any single-square keyboard cursor movements will alter squares that the cursor passes over to the current brush. Toggling sticky mode 'on' also does a 'stamp' of the current brush at the current cursor location.

F1-F8, when unshifted, change your brush type to the appropriate type, as if the gadget had been clicked. When shifted, the level will be filled with the requested square type, without change to the current brush.

Alphanumeric 1-8 change the square under the cursor to the relevant type, without affecting the current brush.

The cursor keys or numeric keypad move the cursor around the field. The default movement is one square. (If sticky mode is on, copies of the brush are left behind). Holding Alt moves the cursor in increments of five squares. Holding Shift or Ctrl moves the cursor to the relevant edge of the field. (The cursor is shown as a white dot, or in sticky mode as a red dot.)

Numeric (and) cycle the brush and square under the cursor through the available objects
, lower and higher, respectively.

Menu

Project New	Amiga-N	Clears the fieldset.
Project Open...	Amiga-O	Loads a fieldset.
Project Revert	Amiga-B	Reloads the current fieldset.
Project Save	Amiga-S	Saves current fieldset.
Project Save As...	Amiga-A	Saves with new filename.
Project Delete...	Amiga-D	Deletes a fieldset.
Project Quit	Amiga-Q, Shift-Esc	Exits to CLI/Workbench.
Edit Cut	Amiga-X	
Edit Copy	Amiga-C	
Edit Paste	Amiga-V	
Edit Erase	Amiga-E	Clears this level.
Edit Delete		Deletes this level.
Edit Insert		Inserts blank level here.
Edit Append		Adds blank level to end.
Settings Animations?	Amiga-T	Animations on/off.
Settings Create Icons?	Amiga-I	Create icons when saving?
Settings Thick Tails?	Amiga-H	Draw thick worm tails?
Help Creatures...		List of creatures.
Help Objects...		List of objects.
Help About...	Amiga-?, Help	Credits window.

Mouse control

The gadgets to the left of the field will change your current 'brush' to the appropriate type. This brush is assigned to the left mouse button, and the keyboard and joystick. The pointer imagery will be altered to reflect your brush type, for some brush types.

The right mouse button, when over the playfield, will function as 'empty' rather than bringing up the menus.

Clicking on the field will change the current cursor location to the chosen square, and change that square to the current brush (or to 'empty' if the right mouse button is clicked).

You can click on any object in the Help|Objects... requester, and that object will become your current brush.

Teleports:

Two teleports may be defined in the field editor. These teleports will teleport to each other during gameplay. They are independent of any teleports that may be created randomly as the game progresses. You use the T key for teleports. ←

Start:

The starting location for all worms. This is invisible during gameplay. Usually it is best left where it is, in the centre of the field. You use the S key to set the starting location.

```

      -
     / \
      |
      |
     Blue
<--- Green START Red --->
     Yellow
      |
      |
     \_/
  
```

Octopi:

Each level may have up to 12 user-defined octopi. You use the O key for octopi.

Arrows:

You use the U key for up-arrows, and the D key for down-arrows. Arrows change the speed of the worm; eg. if a worm travelling up (north) hits an up-arrow, its speed is increased; if a worm travelling down (south) hits an up-arrow, its speed is decreased.

1.11 Hints for Creating Fieldsets

All worms start from the same square. Worms may emerge from it in any orthogonal direction. Leave enough space. Fields should not give an unfair advantage to any particular worm(s). The easiest way to ensure this is through the use of symmetry.

Mazes one square wide are difficult. 3-5 squares width is usually more appropriate for less skilled players.

Point-squares (ie. silver and gold) should be used with restraint, and generally only as a reward for difficult maneouvering. The average field should give have only about 5% point-square density. Richer fields can be made for experienced players, but they should be as a reward for skill.

Goats only fire when they are on the edge of the stone/wood/metal/tail/slime, but they move quasi-randomly. Therefore, some configurations will be more deadly than others. For example, thin lines provide maximum contact between worms and goats, and thick square areas will quickly generate new goats.

Lightly sprinkle your treasury with objects. Use objects very sparingly or not at all in other levels.

Metal squares which are aligned with each other can be deadly, as fragments can bounce between one and the other endlessly until a collision occurs. Remember also that metal is unjumpable.

1.12 Other Information

Contact Details

Development System

Source Code

History

Other Software

1.13 Contact Details

Licence

Worm Wars for Amiga is freeware. It is available as a service to the Amiga community. There are no limits on usage, distribution or modification, except that you are not allowed to modify and/or distribute it for commercial purposes or port it away from the Amiga without consent.

Permission is hereby granted to the Amiga community in general to use and distribute this program as they deem appropriate.

Bugs

Amiga development and style guidelines have been adhered to, using the official Amiga Developer CD 2.1 as authoritative reference.

Please contact us immediately if any bugs are found. You can use our utility, Report+, to generate bug reports for Worm Wars, though this is not required.

Submissions

Fieldsets, etc. can be submitted to Amigan Software and may be distributed with a future release and/or offered for download, with acknowledgements to the contributor(s).

Users are invited to contribute suggestions for the future of the game. Special thanks to Jilles Tjoelker.

Future

Amigan Software have demonstrated their commitment to the Amiga for many years. We intend to continue further AmigaOS programming.

Contact details

E-Mail amigansoftware@abime.net

Website <http://www.users.bigpond.com/james.jacobs/amigan.html>

Mail James Jacobs
 Amigan Software
 11 Yate Gdns
 RIVETT ACT 2611
 Australia

Voice Australia (02) 6287 4917

1.14 Amiga Development System

Hardware Virtual A1200 (MC68020+FPU on 233MHz Pentium)
 1.2Gb 3.5" IDE hard disk (2Gb compressed)
 2Mb chip RAM
 8Mb fast RAM
 56Kbps Dynalink (Rockwell) modem
 ESS MF-1868 (SoundBlaster Pro-compatible) sound card
 AcerView 56L monitor running as 15Hz AGA PAL (640*512)
 Battery-backed clock

Firmware Kickstart 3.1

Software WinUAE 0.8.8 R9 and 0.8.17 R3
 AmigaOS 3.9 with all Boing Bags and other updates
 SAS/C 6.58 with SLink and SAS/C Editor and CodeProbe
 Amiga Developer CD 2.1 and 3.9 NDK
 Amiga Lint 2.0b
 BlowUp

CheckGuide
 CodeWatcher 1.4
 CXXC 1.4
 CygnusEd Professional 4.2
 Deluxe Paint 5
 IFF 2 Source 1.0
 IO_Torture
 KingCon 1.3
 LhA 1.51
 MungWall
 OctaMED 5 and MEDPlayer Programmer's Sources
 PatchWork
 PrintA
 Report+ 5.63
 Sashimi
 SoundBox 2.9 beta
 StackSnoop
 VirusChecker][2.2 (BF 2.18)

Thanks to all those whose software was used to create Worm Wars.

1.15 Source Code

The Amiga version is written in hybrid C/assembler (SAS/C 6.58), whilst the IBM-PC version is written in hybrid C/C++ (Microfilth Visual C++ 5.0), using Microfilth Foundation Classes.

Both platforms use the same game engine, written in C, with API calls for the relevant OS in other modules.

SAS/C 6.58 source code is provided, with which you may do as you wish for non-commercial purposes.

The Amiga source files are:

system.c,o	Most Amiga-specific code.
fe.c,o	Field editor.
engine.c,o	Code common to the Amiga and IBM-PC versions.
tails.c,o	Graphics for the thin worm tails.
squares.c,o	Graphics for all other squares.
amiga.h	Amiga-only header used by all modules.
diff.h	Platform-specific header used by system, engine modules.
same.h	Header file common to all modules and both platforms.
ww.link	AmigaDOS script to link modules into executable.
SCOPTIONS	SAS/C options we prefer to compile with.
libproto.h	MEDPlayer.library C header.
stdafx.h	Required for IBM-PC compatibility. :-(

1.16 History

7.21a [Amiga 2.04+]. Mon 21 Oct 2002.
 7.21 [Amiga 2.04+][Windows 95/98/NT/2000/ME/XP]. Wed 11 Sep 2002.

7.2	[Amiga 2.04+].	Mon	9	Sep	2002.
7.1a	[Amiga 2.04+].	Mon	19	Aug	2002.
7.1	[Amiga 2.04+].	Thu	8	Aug	2002.
7.02	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Sat	6	Jul	2002.
7.01	[Amiga 2.04+].	Sun	2	Jun	2002.
7.0	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Mon	8	Apr	2002.
6.9	[Amiga 2.04+].	Thu	7	Feb	2002.
6.81	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Tue	25	Sep	2001.
6.8	[Amiga 2.04+].	Mon	10	Sep	2001.
6.7b	[Amiga 2.04+].	Sun	12	Aug	2001.
6.7a	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Mon	25	Jun	2001.
6.7	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Tue	22	May	2001.
6.6	[Amiga 2.04+].	Sat	21	Apr	2001.
6.5	[Amiga 2.04+].	Sun	31	Dec	2000.
6.4	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Sat	7	Oct	2000.
6.3a	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Fri	18	Aug	2000.
6.3	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Tue	8	Aug	2000.
6.2	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Thu	22	Jun	2000.
6.1	[Amiga 2.04+].	Thu	15	Jun	2000.
6.0a	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Thu	8	Jun	2000.
6.0	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Thu	1	Jun	2000.
5.60	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Wed	26	Apr	2000.
5.6	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Wed	19	Apr	2000.
5.51a	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Wed	29	Mar	2000.
5.51	[Amiga 2.04+].	Tue	28	Mar	2000.
5.5	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Tue	14	Mar	2000.
5.4	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Sun	5	Mar	2000.
5.3	[Amiga 2.04+].	Sun	13	Feb	2000.
5.21	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Wed	5	Jan	2000.
5.2	[Amiga 2.04+].	Wed	29	Dec	1999.
5.1a	[Windows 95/98/NT/2000/XP].	Tue	28	Dec	1999.
5.1	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Tue	23	Nov	1999.
5.0a	[Windows 95/98/NT/2000].	Tue	2	Nov	1999.
5.0	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Mon	1	Nov	1999.
4.4	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Wed	22	Sep	1999.
4.3a	[Windows 95/98/NT/2000].	Wed	8	Sep	1999.
4.3	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Sun	5	Sep	1999.
4.2	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Sat	7	Aug	1999.
4.1	[Amiga 2.04+] [Windows 95/98/NT/2000/ME/XP].	Sun	6	Jun	1999.
4.0	[Amiga 2.04+].	Sun	1	Nov	1998.
3.3	[Amiga 2.04+].	Wed	29	Jul	1998.
3.2	[Amiga 2.04+].	Tue	3	Dec	1996.
3.1	[Amiga 2.04+].	Tue	16	Apr	1996.
3.0	[Amiga 2.04+].	Thu	18	Jan	1996.
2.2	[Amiga 1.2+].	Sun	5	Feb	1995.
2.1	[Amiga 1.2+].	Tue	27	Dec	1994.
2.0	[Amiga 1.2+].	Sat	3	Dec	1994.
1.7	[Amiga 1.2+].	Wed	26	Oct	1994.
1.6	[Amiga 1.2+].	Sun	9	Oct	1994.
1.5a	[Amiga 1.2+].	Thu	25	Aug	1994.
1.5	[Amiga 1.2+].	Sun	10	Jul	1994.
1.4	[MS-DOS].	Wed	9	Mar	1994.
1.3	[MS-DOS].	Fri	11	Feb	1994.
1.2	[MS-DOS].	Thu	4	Nov	1993.
1.1	[MS-DOS].				1993.
1.0	[MS-DOS].				1993.

1.17 Other Software

Report+ 5.63

Report+ is a freeware ReAction-based utility with ten functions:

1. It is an enhanced, reverse-engineered, 100% byte-compatible replacement for the official Commodore bug reporting tool (40.2).
2. It can generate/edit Aminet-style readmes.
3. It can administer the Amiga Certified Software Engineer test.
4. It can perform batch processing on icons.
5. It can access the official manufacturer and product ID registries.
6. It can view IFF FORMs and their component chunks.
7. It can convert between various end-of-line (EOL) formats, optionally also detabulating and/or unwrapping.
8. It can show directory byte usage for any path, optionally also fixing protection bits.
9. It can edit A3000/A4000-type battery-backed memory.
10. It can check your OS3.9 installation (or any other path) for missing, surplus and/or outdated files.

Saga 1.2a

This is a conversion of the 1981 TSR boardgame "Saga: Age of Heroes", for up to 6 players. Players can be controlled by human or Amiga.

In Saga: Age of Heroes, players defeat evil monsters, win fabulous treasures and rule great kingdoms in their quest to gain glory and be the one hero whose memory will live on forever in the songs and stories of the age.

It is currently available in English and German. Full SAS/C source code is included.

Worm Wars for Windows 95/98/NT/2000/ME/XP

The Amiga version of Worm Wars has long been popular in the Amiga community, acknowledged as the premier snake game for many years. It is also available to the masses of benighted Windows users. That version includes dockable toolbars, status bar, automatic saving/reloading of configuration information, and a larger field.

1.18 How to Play

Worm Wars is an arcade game for 0-4 human players. Each player

controls

a

worm

. They move around a semi-toroidal

rectangular (45x40) playfield, leaving their tails behind them. Each

worm

aims to amass the most

points

, to

complete the levels and to survive as long as possible.
The rightmost column is not actually part of the fieldset but rather indicates the vertical position of the current number and all worms.

Controls
Squares
Creatures
Numbers
Objects
Scoring
Strategy
Fragments
Teleports
Timebombs

1.19 Creatures

	Type	Points	Damage
Bird	40	3	
Cloud	20	3	
Dog	20	5	
Drip	100	5	
Fish	20	2	
Giraffe	40	3	
Goat	50	2	
Lion	50	5	
Octopus			

	50	5
Orb	40	5
Otter	150	5
Penguin	20	3
Protector	0	5
Slime	0	2
Whirlwind	100	5
Worm	0	5

1.20 Controls

In-game keyboard controls

```

Esc . . . . .
. . . . .
..... Q W E . T . . . . P . . . .   7 8 9 .
... .. A S D F . . . . .           U   4 5 6 .
Shift . Z X C . . . M . . . Shift   L D R   1 2 3 E
.. .. -----Spacebar----- .. ..   -0- . n

      Green worm           Either           Red worm

```

Esc exits to the
title screen

Shift-Esc exits to CLI/Workbench.

P pauses until P is pressed again.

M toggles the music on and off.

F toggles the sound effects on and off.

T (for turbo) will run the game as fast as possible, for the rest of the level, when there are no human worms alive. The clock will be lit whilst in this mode.

Worm control principles

Worm control is an acquired skill and is the single most important factor in playing successfully and enjoyably.

A worm has direction and speed. It is unable to ever stop completely, but its speed is adjustable between three or four settings. It can

turn 45° , 90° or 135° in any direction and can move orthogonally or diagonally.

The worm can pass through most types of square, though some cause damage. Some squares slow the worm down, halving its speed for every square traversed (to a minimum of 'slow'). Stone, metal, goats, octopi, fish, and blocked

teleports
will block the worm's motion as well

as slow it.

Control of the worms is by keypresses; that is, press and release in a normal keystroke motion. There is no reason to ever hold down any key. Only fresh keystrokes are ever looked at by the program. Also, holding down a key may lock out your input (see below).

For joystick control, it is best to nudge the joystick briefly in the direction you want, and let it return to the central upright position.

Each move, a worm is able to do any one of five things:

- . fire (in the current direction);
- . change direction;
- . change speed (one speed in either direction);
- . jump; or
- . nothing.

The worm will move one square regardless. This means you can:

fire in your current direction	and move in your current direction,
change direction	and move in the new direction,
change speed	and move in your current direction.
jump	leaping 2-4 squares and then landing, in your current direction.
nothing	and move in your current direction.

Speed changes are accomplished using the same keys as for turning. Pressing the direction you are already travelling increases your speed; pressing against that direction decreases your speed.

Keyboard [green and red]

If only one human is selected for the keyboard, all three sets of keyboard controls are available for that player, whether it is the green or red worm. When only one human is selected for the keyboard, we suggest using the cursor keypad, as it has the best rollover properties (ie. you can press a key whilst another is already held down). However, the cursor keypad obviously does not support diagonal movement.

Two players may use the keyboard, playing at each end. If there are two humans selected for the keyboard, the cursor keypad cannot be used, due to keyboard hardware limitations. The green worm must use the leftmost controls (QWEASDZXC, spacebar) and the red worm must use the rightmost controls (the numeric keypad).

Note that the leftmost and rightmost controls do not rollover properly, due to keyboard hardware limitations. This means that you must release all keys before pressing another one, or the new keystroke is ignored and so is all further input from that player until all keys are released. The other player's keystrokes are completely unaffected by what you do; you can only cause this problem for yourself.

X and C are both down-right for the green worm.
Spacebar is the firebutton for the green worm.

5 and 2 are both down for the red worm.
0 and Enter on the numeric keypad are both firebuttons for the red worm.

Joystick [all]

The green and red worms can use joysticks plugged into a parallel port joystick adapter. These extra joystick ports are ports '3' and '4', respectively. This uses direct hardware access, rather than using the parallel.device.

The blue and yellow worms can use joysticks plugged into port '2' and '1', respectively. This uses direct hardware access, rather than using the gameport.device.

CD32 gamepad [all]

You can use a CD32 controller to control any worm. The relevant ports are the same as for joystick play (see above).

1.21 Squares

The various types of square are as follows:

Empty: Harmless. Worth 1 point.

Silver: Harmless. Worth 10 points.

Gold: Harmless. Worth 20 points.

Dynamite: Harmless. When touched, sets off adjacent dynamite in a chain reaction. Each square dynamited is worth 5 points.

Wood: Painful. Slows movement.

Stone: Painful, slows movement, blocks movement.

Metal: Painful, slows movement, blocks movement, reflects fragments/bullets, can't be jumped over.

Frost: Worms can't turn when traversing frost, but it doesn't cause pain or slow/block movement.

Arrows: These change the speed of the worm; eg. if a worm travelling up (north) hits an up-arrow, its speed is increased; if a worm travelling

down (south) hits an up-arrow, its speed is decreased. You won't leave tail whilst traversing arrows; the arrows will remain in position.

1.22 Numbers

There will always be one, neither more nor less, number onscreen at any given time. When a worm collects it, the next number will appear.

The main objective of the game is to complete all the levels. The level will be completed when all the ninth number is collected. The level is considered to have been won by the worm with the most tail/remnants onscreen at the time of completion. That worm receives:

- 100 x old level
- 10 x seconds remaining
- 50 x how many numbers you collected
- 1 x tail squares of your colour onscreen

Only worms and their protectors go on to the next level; other creatures perish.

1.23 Objects

	Type	Points	Frequency
Affixer	60	Very rare	
Ammo	20	Common	
Armour	20	Common	
Bias	50	Common	
Bomb	30	Common	
Bonus	10	Common	
Brakes	10	Uncommon	
Clock	60	Rare	

Converter	50	Uncommon
Cutter	80	Uncommon
Cyclone	90	Uncommon
Enclosure	20	Uncommon
Glow	30	Rare
Grower	50	Uncommon
Healer	90	Very rare
Ice	60	Rare
Life	50	Uncommon
Lightning	40	Uncommon
Magnet	80	Rare
Missile	40	Uncommon
Multiplier	50	Rare
Powerup	30	Uncommon
Protector	50	Rare
Pulse	40	Uncommon
Pusher	50	Uncommon
Remnants	40	Uncommon
Sideshot	30	Rare

Slayer	40	Rare
Slower	40	Rare
Switcher	70	Rare
Treasure	100	Very rare
Umbrella	100	Very rare

1.24 Affixer

Affixer

This causes your
 protectors
 to stop rotating and
 zigzagging. They are 'fixed' into position at whatever position they
 were in at the time. This makes them more useful, as you can more easily
 predict where they will be at any given time in advance.

1.25 Ammo

Ammo

Ammo provides 2-6 bullets.

Bullets are fired by pressing your appropriate fire
 control

.
 If a bullet is available, you will fire. This bullet is instantaneous. If
 you have no bullets, you may
 jump

.
 The width of your bullet depends on
 powerups
 . Normally,
 it is 1 square wide, but it can be up to 7 squares wide. In reality,
 up to 7 independent bullets are fired.

A bullet which hits a
 teleport
 will be teleported and continue on its way.
 You will get the teleport
 skill bonus

, too. That segment of the bullet will thereafter leave a trail of gold behind it.

A bullet which goes through the same teleport twice, or is reflected off metal twice, will be destroyed.

In the absence of obstructions, a bullet continues going until it reaches the edge of the field.

1.26 Armour

Armour

Armour will protect worms from being hurt by many different causes. For a list of them, click

.

Whenever a worm with armour goes over a tail square it receives:

5 points	if it eats friendly tail. Tail turns to silver.
10 points	if it eats enemy tail. Tail turns to gold.

Armour is only temporary. Its power counts down at a constant rate, and can be extended by the acquisition of more armour.

While you are have armour your head-image will be different.

1.27 Bias

Bias

Provides these benefits:

birds will not chase you.
 vampirism: whenever you cause a worm to take damage, you gain as many lives as they lost. Whenever you kill a creature, you gain a life.
 otters will leave dynamite instead of stone.
 when you get (trigger) a bomb, what you destroy will turn to dynamite.

Bias counts down at a constant rate. It can be extended by the acquisition of more bias.

1.28 Bomb

Bomb

As soon as you go over one of these the bomb will go off, clearing the area around you. The blast radius is random, but is always constrained by the field edges.

1.29 Bonus

Bonus

You will be awarded the currently displayed
number
, and the currently
displayed number will be incremented by one.

1.30 Brakes

Brakes

You will be able to use the 'very slow' speed. Brakes can't be lost.

1.31 Clock

Clock

If there is still any time remaining, the amount of time remaining is increased.

1.32 Converter

Converter

This will instantly convert all currently active
fragments
into
friendly
missiles
.

1.33 Cutter

Cutter

This allows you to 'cut' or 'tunnel' your way through such things as tail. Every move, the squares in front of you, the two to your side and the two diagonally in front behave as though affected by a

```
bombblast
```

except that objects are not destroyed.

```
...
.#.
|
|
```

One thing to keep in mind is that whilst you have cutter you cannot make enclosures. Cutter counts down at a constant rate.

1.34 Cyclone

Cyclone

When you get this, a
whirlwind
will be created somewhere on the
playfield. It is just as dangerous to you as to everything else.

1.35 Encloser

Encloser

This allows you to form valid enclosures even if the an axis is one square longer than allowable, ie. slightly rectangular enclosures. For example, you can do a 9x10 enclosure, or a 4x5 enclosure, or a 7x6 enclosure. This ability remains until the end of the level.

1.36 Glow

Glow

Whilst this object is operating, the tail that you leave will glow, causing extra harm to enemy worms and no harm to yourself.

1.37 Grower

Grower

Grower will cause all gold, silver and dynamite onscreen to 'grow', expanding into their neighbouring squares if those neighbouring squares are empty.

1.38 Healer

Healer

If you have less than your starting lives (100), boosts you back to 100.
If you have 100 or more, boosts you to 200.

1.39 Ice

Ice

Ice will freeze all creatures, except missiles and worms, for a short period of time. Additional ice collected will be added onto the amount remaining. It does not matter which worm collects the ice.

1.40 Life

Life

Life provides 2-6 lives.

1.41 Lightning

Lightning

When you get lightning, most squares adjacent to your tail flash momentarily. This will harm:

unarmoured enemy worm heads	
unarmoured orbs	replaced by bonus
goats	replaced by bonus
slime	replaced by empty
fragments	replaced by empty
drips	replaced by empty
objects	replaced by empty
penguin	replaced by empty
enemy missiles	replaced by empty

The things that are not flashed are:

stone

metal
wood
numbers
graves
frost

1.42 Magnet

Magnet

This causes some or all objects to travel towards you in a guided fashion. Apart from the fact that these objects are now 'magnetized' (ie. moving) they remain normal objects in every other respect. Objects which are magnetized may not be remagnetized by other worms.

1.43 Missile

Missile

Missiles are automatically guided. They hunt the following things:

worms
enemy missiles
orbs
goats

The missile hunts whatever is nearest, and recalculates its target every move.

Missiles will not cross field edges.

If you collect a missile whilst there is nothing for it to chase, you will receive only points; no missile will be generated.

1.44 Multiplier

Multiplier

Each time you get a multiplier, your scoring will be doubled. If you get more than one, it is doubled again each time. You can collect up to three (3) multipliers.

Your object points for actually getting the multiplier are affected by the new multiplier itself.

Multipliers	Scoring
0	*1
1	*2
2	*4

3 *8

At the end of each level, your multiplier will be reduced by one (ie. half).

1.45 Powerup

Powerup

This is power for your bullets. Each bullet you fire becomes wider, requiring less aiming accuracy and causing more general devastation.

Symbol	Meaning		
<number>	bullet number		
:	worm head, facing up (north)		
:	worm head, facing up-right (northeast)		
#	worm tail		
1	213	42135	6421357
1	213	42135	6421357
1	213	42135	6421357
1	213	42135	6421357
:	:	:	:
#	#	#	#
#	#	#	#
Single (no powerups)	Triple (one powerup)	Quintuple (two powerups)	Septuple (three or more)
1	312	53124	7531246
1	312	53124	7531246
1	312	53124	7531246
1	312	3124	31246
:	:2	:24	:246
#	#	#	# 6
#	#	#	#

1.46 Protector

Protector

Acquisition of this item will give you a companion who stays near you, collecting

objects for you exactly as if you had moved over them yourself, and defending you from some forms of damage.

You can have up to three protectors. The first two orbit around your head. The third will be a 'nose', suspended three squares in

front of your head, which zigzags. The rate of orbiting/zigzagging is the same as the worm's speed; that is, they move when the head moves.

Protectors are invisible over friendly tail, friendly remnants and teleports.

An

affixer

will stop the protectors from rotating and

zigzagging.

1.47 Pulse

Pulse

Up to 7 fragments are fired outwards when a pulse is collected.

1.48 Pusher

Pusher

This enables you to push stone, goats, metal and tail. You will push the square in whatever direction you are facing. Pushing is similar to the pushing of

timebombs

. You can push things off the field if

you so desire. You are not allowed to push more than one square at a time. You can push over objects and empty/silver/gold. Pusher is kept only until the end of the level.

1.49 Remnants

Remnants

This enables your bullets to leave their remnants behind them as they travel.

These images of the bullet do no harm to the worm that fired it, nor to any other creature except enemy unarmoured worms. Remnants are kept only until the end of the level.

1.50 Sideshot

Sideshot

This means that whenever you fire, in addition to any forward bullets you will also fire a bullet to each side.

1.51 Slayer

Slayer

Slayers are used instantly. They will destroy all
goats
,
slime
,
drips
,
orbs
,
penguins
,
octopi
and enemy
missiles
, and cause damage to all
unarmoured
enemy worms.

1.52 Slower

Slower

Slower will slow all fragments, orbs, goats, drips and penguins to half their previous speed. The speed of these is reset at the start of each level.

1.53 Switcher

Switcher

All enemy tail onscreen is instantly changed to friendly tail. If it was already friendly tail, it glows.

1.54 Treasure

Treasure

The worms will enter a bonus level for a limited amount of time. At the conclusion, the next level is reached. The full time bonus (2:00, which is 1200 points) is given at the end of the treasury.

There are three types of bonus level: treasury, drips and penguins.

1.55 Umbrella

Umbrella

This skips 2-3 levels.

1.56 Strategy

Goats prefer to be near free space (empty, gold or silver), so that they can shoot.

Faster speeds are useful for:

- racing another creature for a number or object;
- hunting another creature when you have armour;
- evading missiles and fragments;
- laying tails to trap other creatures;
- creating gold/silver with armour;
- amassing more square-points by moving;
- amassing more square-points at level completion;
- making large jumps;
- causing protectors to rotate faster.

Slower speeds are useful for:

- making small jumps;
- waiting in a certain region of the screen;
- not filling the playfield;
- enclosure;
- easier control.

Remember that bullets, bombs, missiles, fragments and drips will not wrap around playfield edges. If you do, they cannot follow.

Certain objects work well together. Eg.:

- bullets, powerups and bias;
- multipliers and armour;
- armour and grower;
- treasure and ice;
- protectors and affixer.

To avoid a missile or bird, get a creature, or a solid barrier such as stone or metal, between you and it.

A good way to handle slime is to encapsulate it with your tail, so it cannot spread.

When you have armour, it is good to go over wood and slime as that is an easy way to destroy it.

1.57 Worms

Worms can be controlled by humans or the Amiga. For information on how to control them, click .

A worm consists of a head, one square in size, and a tail, which is formed in the wake of the head's passing. As the worm moves, the tail stretches out behind its head. Collision with any part of a worm by the head normally results in damage (to the head).

Worms can move in eight directions and at three speeds; getting brakes allows another speed (very slow).

Worms begin with 100 lives. Whenever you are taking damage your head will flash as an 'X'. When 0 lives is reached, the worm is dead and there will be a grave where you died. Every 1,000 points you will be awarded an extra life.

Graves

If a worm or protector robs the grave of a dead worm, the robbing worm will get all the objects that the dead worm had, in addition of course to what it already had itself.

Graves absorb any bullets, missiles, drips or fragments, and are immune to bombblasts.

Jumping

A worm with no ammo can jump instead. If the destination is stone, metal, a goat or a blocked teleport you will not jump, and metal cannot even be jumped over. The distance varies according to your speed, and is shown below.

Name	Speed	Jump distance
------	-------	---------------

* Very slow	quarter speed	1
. Slow	half speed	2
. Normal	normal	3
. Fast	double speed	4

* Very slow speed is only available if you have
brakes
.

'Worm motion is stopped': Every move in which the worm takes damage from that cause, its speed is halved (to a minimum of 'slow'). The worm is unable to actually enter the square.

Unarmoured

worms can also die by the following methods:

remnant	5	worm is slowed
worm bullet	5	
enemy missile	5	
enemy protector	5	
orb	5	
dog	5	
worm head	5	
bird	3	
fragment	3	if armoured, fragment is reflected
bombblast	3	
lightning	3	
enemy slayer	2	
slime	2	
wood	1	worm is slowed
worm tail	1	worm is slowed

Enclosure

If your tail forms an exact square which measures 4x4-12x12 including the tail (2x2-10x10 of interior area), this is known as enclosure. Any squares within the closed area which are empty, silver, gold or enemy tail are converted to your glow. You receive 10 points for each enclosed square which is thus converted. Enclosure is a basic ability of any worm.

Note that enclosure is only checked for when the worm's head is at a corner of the square. And remember also that the enclosure must be square: the same size in both x and y dimensions. Diamond enclosures (ie. an enclosure made diagonally) are invalid. This is the skill of enclosure. If you have an

encloser

you can make enclosures that are not quite square.

Symbol	Meaning
:	worm head
#	worm tail
.	enclosed area

```

5x5          6x6 interior  7x7 interior
#####      #####      :#####
#...#       #...#       #...#
#...#       #...#       #...#
#...#       #...#       #...#
#####:     #...#       #...#
              :#####   #...#
                          #####

```

1.58 Orbs

Orbs move diagonally and bounce whenever they hit a tail, metal, stone, wood, goat, octopus, or fish, so their motion is predictable.

Objects

Objects can be used by orbs, as follows:

Affixer	This will remove any affixer from any worms.
Ammo	Doubles the orb's speed.
Armour	The orb will split.
Bias	This will remove all bias from all worms.
Bomb	Same as worm.
Bonus	The orb will split.
Brakes	Doubles the orb's speed.
Clock	The time remaining is reduced.
Cyclone	Doubles the orb's speed.
Encloser	Doubles the orb's speed.
Freedom	All remnants onscreen will change to empty.
Grower	All wood onscreen will 'grow', expanding into their squares if those neighbouring squares are empty.
Healer	The orb will split.
Ice	The orb will split.
Life	The orb will split.
Lightning	Doubles the orb's speed.
Magnet	Any magnetized objects become demagnetized.
Missiles	All missiles will be destroyed.
Multiplier	Same as worm.
Powerup	Doubles the orb's speed.
Protector	This will slay all protectors.
Pulse	Doubles the orb's speed.
Pusher	This will remove any pusher from all worms.
Remnants	This will remove any remnants from all worms.
Sideshot	This will remove any sideshot from all worms.
Slayer	Doubles the orb's speed.
Slower	Doubles the orb's speed.
Switcher	All tail onscreen is instantly changed to wood.
Treasure	The orb will split.
Umbrella	The orb will split.

Splitting

The orb will split into up to four orbs, depending on the number of orbs already in play.

Each orb will be an exact clone of the original in all attributes, except (obviously) starting direction. Once born, they are completely independent.

1.59 Otter

Otters are created when the allowed time for the level has expired. They appear at the extreme horizontal edges of the field and gradually bounce up and down towards the centre, filling the area behind them with stone. If any worm has bias, they will leave dynamite behind rather than stone. They are almost indestructable, although a slayer will kill them and they can be frozen by a clock or ice.

1.60 Giraffe

Giraffes do not move. Any worm colliding with a giraffe will be kicked around 5 squares, if possible, or otherwise the giraffe will die and the worm will take damage.

1.61 Goat

Goats walk along the top of stone, metal, wood, tails and slime, ←
firing

fragments
randomly when they are at an edge (next to
somewhere they can't go). Goats prefer to be at an edge and tend to move
towards and stay at one. When not at an edge, they cannot fire.

1.62 Lion

Lions move orthogonally (horizontally and vertically) at random, ←
bouncing
the same as an
orb
and also bouncing off the field edges, and sometimes
changing direction for no reason.

1.63 Slime

Slime may be randomly created at during play. This slime can 'grow' into some adjacent squares over time. It never moves, only grows.

1.64 Drips

Drips appear near the top of the field, and fall downwards. They can be any of the four colours (green, red, blue and yellow).

Drips of your own colour are worth points. Drips of the wrong colour hurt you, unless you are armoured. In either case, the drip is destroyed.

1.65 Fragments

Fragments are generated by:

pulse
explosions.

goats
firing.

octopi
firing.

clouds
firing.

Fragments travel until they are absorbed or reach the field edges.

1.66 Penguin

Penguins are randomly created, independent creatures which move randomly. They are not aggressive but will harm things that collide with them. They only ever move onto empty (or silver/gold) squares.

1.67 Whirlwind

These are created when a worm collects the cyclone object.

They move very quickly in a random fashion, though they tend to drift slightly upwards over time. They are one of the more dangerous types of creature.

1.68 Dog

These appear randomly. They wait patiently until a worm passes over them. Then they begin to awaken and soon they chase the worm along its tail. They cannot cross field edges.

1.69 Bird

Birds hover in one square. When an unbiased worm gets too close, they swoop on that worm, chasing it in a kamikaze-style attack.

1.70 Cloud

These appear randomly. They only ever move horizontally, left and right, changing direction at each end of the playfield. Occasionally they will shoot out a pair of fragments: one up and one down.

1.71 Teleport

These always come in pairs. Going through one will teleport you to ↔ the other, and bestow points also.

If a teleport is blocked (there is stone, metal or a goat next to

its partner on the side where you would emerge) then the teleport cannot be used and is dangerous. You will slow down from hitting it and not be able to use it. It is obviously possible for a teleport to be blocked to some approaches and unblocked to others.

Protectors
are invisible over teleports.

Two teleports may be defined in the
field editor
; another

two may appear randomly during play. These two sets are completely independent of one another.

1.72 Timebomb

This ticks from 9 down to 0 before detonating. During ticking, it ↔ can be pushed around by worm heads. If pushed off the field edges, 100 points are given and the timebomb is destroyed without detonation. If pushed

against anything, or shot, or hit by most things, it will detonate immediately, with harmful consequences to anything within its radius. Points are never awarded for timebomb blasts. The explosion is the same as a normal

bomb

.

1.73 Octopus

These are created with the field editor

. Up to 12 can be

created for each level. They can also appear randomly during play. They do not move. Occasionally they will decide to spin around, firing fragments as they do so.

1.74 Fish

These live in the same types of squares that goats and octopi do (slime, metal, stone, wood, tail). They move randomly. Any tail which they swim over is turned into its corresponding colour of remnant.

1.75 Scoring

Square points

1	Empty
5	Dynamite
10	Silver
20	Gold
100	Grave
100	Number

The 'number' points are multiplied by the number, eg. getting number '5' is worth 500 points, getting number '6' is worth 600 points, etc.

Skill points

5	For each square you turn to silver with armour.
10	For each square you turn to gold with armour.
10	For each enclosed square.

There are also level completion bonuses.

Objects

and

creatures

are worth various amounts of points.

All points are multiplied by the number of worms selected for human and

Amiga control, and of course by any
multipliers

.
