GAME DESIGN'S GUIDELINES for PLAYTESTING GAMES

Introduction:

This document is a summary of advice and guidelines for playtesting games produced by Coleco's EDG Game Design group. Not all of these guidelines will apply to every game, but most are universal. Some of the guidelines are so important and essential that they should be regarded as rules. These are given special emphasis in the text.

1. KNOW THE GAME.

Do not attempt to playtest an unfamiliar game until you know what the game is supposed to do. Without an understanding of how the game is supposed to work, you won't be able to tell what is right and what is wrong.

A. Talk to the Designer. Tester and designer should have regular discussions about the game program. At the beginning of the playtester's interaction with the game, the designer should sit down with the tester(s) and "walk him through" the game, explaining what should happen and when, and why. The tester has to know what the designer is after. If the designer doesn't come to you to explain the game, go to him and set up a time to talk about it.

B. Read the Documentation. If you have time, read the Product Description of the game. At the very least, read the Feature Outline, Overview of Game Play and the **Glossary.** (The latter is very important: you must know the proper terms for the game to explain problems effectively.)

Read the Checklists. These communications between the game designer and the developers are essential to your understanding of the game's known problems. Assume that all problems not listed in the latest checklist are unknown to the designer.

In the final stages of debugging, you will have to check the information in the Instruction Guide vs. the actual contents of the game. Read the Instruction Guide very closely, and check every item -- don't take anything for granted.

The designer should make sure you get copies of all this documentation at the appropriate times. If he doesn't, pester him until he gets you what you need.

2. DOCUMENTATION OF TEST RESULTS.

Documentation is the heart of good testing. It is not enough just to play the game. It is not even enough to test the game well. You must communicate your test results to others in a clear, written form, or all your time testing was wasted. Some guidelines:

A. Write it down at once! Memory is notoriously bad, even after only a few seconds have passed. Don't assume you can remember it to write it down later. If the game has a working Pause feature, use that immediately to stop the game while you write.

B. Take lots of notes -- note everything! **Don't** be reluctant to write. The more you take notes, the easier it becomes. Testing one minute and then writing for two minutes is much better than testing for an hour and then writing for two minutes.

C. Be specific and detailed. Include as much detail and description of the situation as possible; any info might be important in helping the programmer track down the problem.

D. Organize your notes. Don't merely hand in your raw data. Synthesize your notes, reorganize them, cut out redundancies and compress them into a manageable, clear memo. Use the QUICK OVERVIEW REPORT FORM for simple items, then add as much detail as you can where it is appropriate. (Don't throw out the raw data -- it may be needed later if we must search for clues to a difficult problem.)

E. Type your notes. No handwritten memos, please.

3. ACCURACY COUNTS.

The more sophisticated game programs are extremely complex, so it is essential that reports of problems be completely accurate, or the programmer may not know where to look to solve the problem. A few pointers:

A. Use accurate terminology. Don't make up names for things if you don't know what they're called -- you'll just confuse everybody. Use the right word. This applies to two areas:
1) Specific Game terms, as found in the glossary in the Product Description (or Feature Outline, if there is no P.D.). Sometimes the game designer, artists or programmers may use other slang terms to describe certain things in the program, and these may acquire legitimacy by showing up on checklists. If you don't know what these words mean, find out, then use them.

2) Computer system terms. We are making games for several different computer systems now, and all of these systems have their own terminology, sometimes even using different names for the same things. Example: "background" means four different things on four different systems. Learn the terminology for the systems you are working on. If you don't know it, ask. But don't use "sprite" on a system that has no sprites, or "pattern block" on a system with "characters."

B. Pay close attention to numbers. It is often important to know how many times something happened before a problem arose. In addition, accurate numbering is critical when dealing with scoring or timer problems. When you can, deal in exact numbers rather than abstracts.

C. Get help if overloaded. Sometimes there is so much happening on the screen that the tester cannot accurately note certain occurences while continuing to play the game. When this occurs, borrow somebody to help you for a few minutes by playing the game while you note down the necessary information.

D. Use the VCR. If you have it on tape, you can go back over a problem as many times as you have to to get all the relevant details. The VCR is a great tool, but it is your interpretation of what it has recorded that is important.

4. BE THOROUGH.

Playing the game is only a small part of testing. You must do everything it is possible to do just to see what happens when you do it. Most problems are completely unpredictable results of rare combinations of events. Some suggestions:

A. Test all possible combinations of game options. In games with independent skill selections, test all possible player/skill choices. This may be tedious, but it must be done. Some problems are skill-related.

B. Test all possible controller inputs in every situation. Test every joystick position, action button, keypad button, or keyboard key on all option screens, during gameplay on every screen, when paused, when game is over, etc. Test for simultaneous inputs from more than one control source. If you find what seem to be controller-related problems, test them again with at least one completely different set of controllers to make sure it is not a hardware failure. (This is especially true of keypad functions.) Test variant controller inputs, like Super Action controller, roller ball, etc.

C. Test on more than one machine. Do not do all of your testing on one machine, even if it is "your" machine. You must be sure that problems are in the software and not the hardware on which you are playing. **Test ColecoVision programs on both ADAM and ColecoVision.** One or the other is not sufficient!

uniquely identify the specif
machine

D. Test with the sound on. You are also playtesting the sounds and music. Be aware of correlations between aural and visual bugs.

E. Try to set up unusual situations. Try things no rational game-player would ever do. Think weird! Commit suicide just as a timer runs out; Move your character to places where it makes no sense to go; Try contact checks in all parts of the screen; Try to roll over scores and counters, put too many things in one place, stay in one place without moving forward, etc. Deliberately do things the game designer did not intend you to do.

GAME PLAYTESTER'S QUICK OVERVIEW REPORT FORM

GAME: NAME ________ SYSTEM ________ PROJ. # ____ REV: ________ MEDIUM: PROM □ DDP □ DISC □
CHECKSUM: ________ DATE:

PLAYTESTER:
DURATION OF TEST:

A. TITLE SCREEN: Correct? YES □ NO □

B. GAME OPTION SCREEN(S): Correct? YES □ NO □

C. CONTROLLER FUNCTIONS Correct? YES □ NO □

D. SKILL LEVEL CHECK: Correct? YES □ NO □
(Note score for each level)

Skl 1, 1 Plyr	________	Skl 1, 2 Plyr	________
Skl 2, 1 Plyr	________	Skl 2, 2 Plyr	________
Skl 3, 1 Plyr	________	Skl 3, 2 Plyr	________
Skl 4, 1 Plyr	________	Skl 4, 2 Plyr	________
Skl 5,__ Plyr	________	Ind Skl, __ & __:	________
Skl 6,__ Plyr	________	Ind Skl, __ & __:	________
Skl 7,__ Plyr	________	Ind Skl, __ & __:	________
Skl 8,__ Plyr	________	Ind Skl, __ & __:	________

E. SCORING Correct? YES □ NO □

F. PAUSE FEATURE Correct? YES □ NO □

GAME PLAYTESTER'S OVERVIEW REPORT

G. RESET FEATURES Correct? YES □ NO □

H. HALL OF FAME Correct? YES □ NO □

I. GAME PLAY PROBLEMS
(Please use additional sheets as needed)

J. GRAPHICS PROBLEMS
(Please use additional sheets as needed)

K. GAME CRASH/FREEZE
(Please list all conditions current at time of crash/freeze)