Intervallo - What I learned from developing a simple game in Basic, after 35 years

The motivation

The Hungarian Commodore Plus/4 FB-group has announced a competition for creating games in three categories: basic, mixed and pure machine code. I was already planning to make something for my *home platform*, so I thought I'd enter.

I knew I wouldn't have enough time to make a complete game with graphics, music etc., so I wanted something simple.

I started programming - in Basic - at the age of 13, and two years later, when I got my first computer, a Commodore 16, I wrote some games for it, first in Basic:

- Dave's Little Car (cca. 1985, lost),
- Music Helicopter (1986) and
- Asteroid (1987, with Ákos Nausch)

Later I switched to Assembly programming, made two music editors, and created some songs with them, which made me kinda' famous in the early C16/Plus4 scene.

It has been *35 years* since my last Basic game.

What if I wrote the game in Basic for the competition? After 35 years?

Of course, the project smells of nostalgia, it would be a lie to deny it. *Time travel to my teenage years*, no doubt.

But there will be lessons learned. Decades have passed since then, which - in this profession, where a new era comes every few years - is a very, very long time. Hardware, programming and languages, software development methodologies have all undergone revolutionary changes.

I've changed too, from an enthusiastic autodidact gifted amateur kid
to an enthusiastic autodidact slightly-better-than-average professional software developer.

Adding the two together is certainly not without lessons.

The game

To put it short and immodestly: I'm the famous ancient musician of the Plus/4 platform, with decades of professional development experience, who starts writing a game in Basic.

Let's specify requirements:

- The game should be written in pure Basic, so it can easily be ported to other machines. Commodore Basic V2.0 seems a good choice, it's quite simple and standard, and also

Commodore 64 is an obvious target.
- The game should not rely on machine-specific features, should not contain graphics, just a plain console.
- The game should not require large assets, e.g. levels, graphics, game logic, texts... simply, I have no time for creating too much content.

Great, but then what kind of game? A music game. Without precise timing and gaphics? I know one: **interval guess game** - not too original idea, but it has educational value, and even possible that someone will play with it.

Now we need a good title. Both music and Plus/4 scene is linked to Italy, an Italian title would be appropiate.

> **Me**: *Bro, pls help me translating something. Despite that the lingua of the music is Italian, and I know some terms, like piano, forte, parlando etc., I don'tknow the name of "pitch distance". Probably, it's "intervallo", but maybe this is not a proper translation. Second, Quart, Quint and Octave, each is "intervallo". I hope, you know music as deep that you can tell the proper term, in Italian.*
>
> **Luca:** *Yes, it's the intervallo. And the pitch is Altezza.*

Intervallo - I like it, it's especially lucky that it's pretty close to the English name.

Obviously, one item had to added to the requirement list:

- the computer should be able to play an interval somehow.

The design

First of all, I wrote the design document, in which I specified the platform, the gameplay, and the development method. It was pretty stable, only a few things are changed during the implementation.

You can check the design document.

The **platform** was a given, I stuck to the text 40x24 text mode. The only extra is, that the good and bad answers are indicated by a short green or red flashing of the border color.

The **gameplay** was cleaned up during development:

- I set up four difficulty levels, each with its own parameters.
- The completely optional parameters only makes the GUI more complicated, but they don't make the game any better, so I left them out.
- Added a very difficult level, which is not shown in the level selection menu.
- Instead of custom choice of the player, it depends on the number of attempt, whether the notes are played toghether or separately.
- Added some text evalution based on points made.
- When the player answers wrong, the correct answer is shown, and the interval is played again.
- Added a sound test function, the player can check whether the TV set's or the emulator's audio is working correctly.

The **development** tools were given, I was using my favourite text editor, *petcat* and *VICE*, I had only one difficulty: text editors do not support line numbering.

Lessons Learnt

Avoid onboard

Lack of: VCS, editor, quick save

Renumber

vs Mumps, labels

Basic is

266 LOC