PREFACE

This book is about computer games. It's about how to develop them using the Python language, but the book also includes some design instruction, ideas about handling assets, and a host of things that should be useful for a game developer.

Python is a programmer's language in that it provides features that programmers usually want and often recode again and again in their various programs. Lists, dictionaries, sets, arbitrary precision integers, dynamic typing – it's an encyclopedia of the tools a programmer uses all the time, or would if it were convenient. Well now it is. *PyGame*, the module used to help a programmer create games, adds to that, a surface on which to draw, many graphics primitives, sound, animation, and interaction. It's a wonderful palette on which game developers can dip their brushes.

If you look up my name on the Internet you will see that I am a professor of Art. That's true, but I feel like a bit of a fraud, and the reason is that I have no training as an artist. I studied mathematics and computer science at university. So how did I end up in art?

I was known for work in image processing and vision in the 1990's as an academic. For some reason, the Game Developer's Conference interested me. In 1998 I registered and attended, and my life was changed. The energy there was incredible. People everywhere were completely enthralled by their work. They were having *fun*. They were doing things and speaking about things that I had not heard about in my academic venues, and those things were *fascinating*. Moreover, their work had an immediate impact on people.

True, the companies were in competition for a share of the commercial games market, but the people at the conference were excited about what they knew and about sharing it. Sharing means, in this context, bringing back something

as well as giving something to the others. Moreover, the group contained computer programmers, musicians and audio specialists, artists, designers, and business folks. A true meeting of multidisciplinary minds.

This book arose from my experience at GDC and my love of computer programming. I have written other books on game development, but for a casual programmer or home developer, I think that Python is a great way to proceed. Python is easy to learn, and *PyGame* has everything a 2D developer needs.

I do presume you have some proficiency in Python. That's necessary to keep the book under 1000 pages; to program games, one first needs to be a programmer. What the book will teach is still significant. Computer science degrees are useful, but few degree programs offer any treatment of *assets*: art, sound, graphics objects. Handling those is essential to any game, and assets are a key component of many practical computer programs.

This new edition has two new chapters, and two new games. An implementation of whist is the subject of Chapter 11 and is intended as an example of how card games can be implemented, and how they differ from other types of games. Chapter 12 deals with platformer games as a genre and specific implementation details that can be adapted to other games of the same type.

The main project in this book is by necessity incomplete. It is a boat race, 2D, and seen from above; but is has sound, animation, interaction, AI, and everything that a simple game should have. It could be more fun. It could have more features. I leave these things to you. As an instructional device I think it has everything that you need. As a *game* it still needs your touch.

There is a lot of code in the book. The code is also included in the companion files, along with the color figures and some very useful tools for asset creation. The programs are included so that you can play with them, modify them, and experiment. If you do not, then you are missing an element of the instruction that this book offers.

Playing games is fun. Making games is fun too, but can also be profitable, educational, and useful in a great many ways. The information you glean from studying game development applies also to other digital media. That's a bonus. As a marketer, web developer, artist, or app developer, what you can learn by studying game development is enormous.

And, of course, it's fun. If you are not having fun, then you are doing it wrong.

Jim Parker
December 2020