The code is in here

Introduction:

Welcome to our cross-cutting project!

The objective of this project was to make a video game in python with some constraints. The game had to use the pygame module and also had to contain and be able to realize trajectories thanks to time equations. Following these two constraints, we decided to make a player versus player fighting game inspired by the Super Smash Bros. game.

In our game, two characters confront each other on platforms and can each use an attack to damage the other player, which will not be reflected in a life bar but in a percentage system that will increase with the number of attacks received and will allow the player to suffer more or less from the impact of the enemy attack. When a player is attacked, he is thrown more or less far according to his critical percentage. Then, as soon as he leaves the screen, he loses one of his 3 lives. The first player to run out of lives loses the game.

The code:

We decided to use classes in order to have a cleaner program and to organize the code in the form of function files with executor files. However, our little or no experience with this area will ultimately make the code less clean than it already was. We are nevertheless happy to have been able to discover this way of writing, especially as it helped us a lot when discovering the structures in algorithmy and programming in C.

Finally we separated the code into these files:

game.py and menu.py are the files that form the menu of our game

player.py contains everything that happens after launching start game in the game menu. So the players and their classes, the objects (clouds), and obviously the background of the game

the sprite.py file is composed of a single class that manages the appearance of the photos during the game in order to have coherent animations
Finally the main.py file is quite short because it just calls the global function of the game which will call other functions etc..
The use of classes allowed us to segment the code into several functions: there is almost no bare code. And so the explanation and the commentary of the code is done rather easily.

Problems encountered:

During the realisation of our project, we encountered various problems. Since we want to make a Smash Bros type of game, it was a competitive game which requires technical aspects on the combinations of attacks and the course of the fight. However, this kind of function is sometimes complicated when you are a beginner and that's why this kind of technical game called competitive is so popular and often requires many hours of play before you can be very efficient.
The organisation was also a test from which we came out full of new tools and skills. Indeed, it was specified that we had to make a session report on the tasks performed at the end of each session. For this we opted for a version control software, the most known, git and github. You can also find more details about the design of the code, the difficulties encountered and more on our github.

What we will improve for the next time:

At the end of this project there are several possibilities for improvement. First of all, diversifying the maps, the possibility to choose between several maps at the beginning of the game, each with its own specificities, could add a new aspect to the game. The possibility of creating your own map by placing the platforms on which the characters fight or by adding various effects to these platforms such as frost to accelerate the characters or a gravity system that could make jumping more tedious or on the contrary improve it. We thought about implementing these new maps in the game but finally opted for one rather sober one for our game.

In order to diversify the battles between players, one can create and add a selection of characters to the menu and create a number of playable characters. Each one would have its own stats such as a different weight from the others which would make it slower when moving and jumping. Maybe even add a life to some of them in exchange for a high movement malus, but this could prove to be unbalanced. In addition, each character could have its own attack which, again, would allow for more diversity in the game between players.

In addition to each character's own attacks, a choice of several attacks could be implemented for the player. That is, each character would have a wider range of attacks and thus allow players to vary their attacks during combat. Each character would have one or two attacks of their own and the rest would be shared attacks with other characters.

To accompany our game, it would have been interesting to develop an artificial intelligence in order to be able to play alone against a computer, this seemed complicated and even if we had first thought of making a co-op mode.

We also thought of adding another game mode. This one would be a game mode for one player against bots. The goal would be to reach the end of the level by fighting the enemies on the road. This would be a scrolling level where the camera follows the player as he moves. This level has the same features as the combat maps, i.e. platforms, holes, traps, gravity zones, and frost zones. Enemies on this map will be simple AIs that will mostly just move around, but some of them might be more aggressive. This system of "extended" levels could be a story for each character, so each character would have their own (several) levels to play. In the end we stayed with the battle system only, this new game mode would be a good addition but would have been too different from the rest and would have taken a main place over the rest of the game. Nevertheless, the possibility of adding this feature is not to be overlooked. Other mini-games can also be added as bonuses that could use the characters in the game, but they won't be very developed and would be more small additions to the main content.

The idea of making a server to let two people play from two different machines was very appealing to us and we were even planning to do it. However, the project is due for a specific date and the work period is parallel to the course periods. We preferred to leave this idea as a bonus like many others. This is one more thing that this project has taught us, to start with what is required and add as much functionality as possible in the remaining time. That's how we organised our work. It's true that this way of working has a disadvantage, it's tempting to stop while time allows us to add more features. Even if we were not in this situation.

It would also have been interesting to add objects that players could interact with, such as damage zones or objects that could be used as projectiles or that gave specific skills.

We would have liked to add more than two players on the map. Or to create a menu where you could choose your character. But the lack of characters was there.

And it teaches us that when making projects like this, programming is by no means the only skill required, it is primary but we need artistic aspects to make the front end of our product development. We need management skills to organise the teams and the work, especially when there are five of us.

What we will personally improve for the next project:

For our next project we all think that the organisation of is to be improved. The organisation is linked to the use of tools that not everyone knows how to use and learning them is not to be neglected even if they are intuitive. Intra-session work is also more than necessary as it is more productive and less restricted by schedules. The group sessions are therefore used to pool the work that each person has done and to organise the next tasks to be carried out for each person. The installation of IDEs and modules is also not obvious to everyone. It is therefore necessary to prioritise it and complete it during the first session. We made the mistake of skipping this session.

Conclusion:

The Transverse project was a great discovery for all of us. Both in terms of teamwork and personal organisation. A discovery about what a project is, with its final date and its defense. But above all, it was a discovery about personal research, because in this project we did not receive any external help from the five of us. By coming out of this project we feel able to go and look for notions that are completely unknown to us on our own and this may prove to be very useful whether it is in the working world or for our next semester.

Contributors:

- Romain Plot
- Matthias Beausseron
- Jean Doutriaux
- Quentin Fourie
- Victor Julou