**Μάθημα**

Αλγοριθμική Σκέψη

**Εργασία:**

Tanks: Ένα Python Multiplayer Game,

με χρήση της Server – Client Αρχιτεκτονικής

**Υπεύθυνοι Καθηγητές:**

Μηνάς Δασυγένης

Πλόσκας Νικόλαος

**Φοιτήτρια:** Κωνσταντινίδου Αθηνά

***Συνεργάτης:*** *Κοσμάς Σουρλής*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ΠΕΡΙΕΧΟΜΕΝΑ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ΕΝΟΤΗΤΑ 1η**

# ΕΙΣΑΓΩΓΗ ………………………………………………………………………………………….3

# ΓΕΝΙΚΗ ΑΝΑΣΚΟΠΗΣΗ …………………………………………………………………….….3

# ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ ………………………………………………………….….3

# ΧΡΗΣΗ & MECHANICS ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ ………………………………………….….3

# ΟΔΗΓΟΣ ………………………..………………………………………………………..…….….3

# ΣΚΟΡ, ΝΙΚΗ & ΗΤΤΑ …………………………………………………………………….….….3

# ΙΔΙΑΙΤΕΡΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ …………………………………………………………….….3

**ΕΝΟΤΗΤΑ 2η**

# ΤΕΧΝΙΚΕΣ ΛΕΠΤΟΜΕΡΕΙΕΣ …………………………………………….……………….….3

# ΑΡΧΙΤΕΚΤΟΝΙΚΗ ………………………………………………………………………….….….3

# CLIENT & SERVER …………………………………………………………………….…….….3

# ΕΠΕΞΗΓΗΣΗ ΤΟΥ ΚΩΔΙΚΑ & ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ …………….………………….….3

# ΠΡΟΚΛΗΣΕΙΣ & BUGS ……………………………………………………………….…….….3

**ΕΝΟΤΗΤΑ 3η**

# ΙΣΤΟΡΙΚΟ ΥΛΟΠΟΙΗΣΗΣ …………………………………………………………….…….….3

# ΠΕΡΑΙΤΕΡΩ ΕΞΕΛΙΞΗ …………………………………………………….…………………….3

# ΣΥΜΠΕΡΑΣΜΑΤΑ ………………………………………………….…..…………………….….3

# ΕΥΡΕΤΗΡΙΟ ΟΡΩΝ ……..……………………………………..…………………………….….3

**ΕΝΟΤΗΤΑ 1η**

# ΕΙΣΑΓΩΓΗ

Με αφορμή, την εκπόνηση της εργασίας στο μάθημα Αλγοριθμική Σκέψη του ΠΜΣ Ανάπτυξη Ψηφιακών Παιχνιδιών και Πολυμεσικών Εφαρμογών του Πανεπιστημίου Δυτικής Μακεδονίας, υλοποιήθηκε ένα multiplayer network παιχνίδι, στη γλώσσα Python. Το παιχνίδι ονομάζεται ‘’Tanks!’’. Επιλέχθηκε το multiplayer χαρακτηριστικό, καθώς είναι πιο δημοφιλές και στα σύγχρονα παιχνίδια. Επιπλέον, δόθηκε ιδιαίτερη έμφαση στην κατασκευή βάσεων δεδομένων και ανταλλαγή (σε πραγματικό χρόνο) δεδομένων μεταξύ ενός Server και ενός Client, καθώς αποτελεί επίσης πολύ δημοφιλές χαρακτηριστικό στα σημερινά παιχνίδια. Σε αυτήν εδώ την αναφορά, θα εμβαθύνουμε περισσότερο στα χαρακτηριστικά του gameplay του παιχνιδιού, στις τεχνικές πληροφορίες, στη διαδικασία υλοποίησης του και γενικά σε πληροφορίες σχετικά με την ανάπτυξη του.

# ΓΕΝΙΚΗ ΑΝΑΣΚΟΠΗΣΗ

**Αφήγηση**

Βρισκόμαστε στη μέση ενός νησιού (όπως διακρίνεται και από τη οθόνη έναρξης), προστατεύοντας ένα κάστρο, το οποίο είναι το τελευταίο καταφύγιο της χώρας. Όλα τα υπόλοιπα καταφύγια έχουν πέσει στα χέρια των εισβολέων. Προσπαθούμε να πετύχουμε τα εχθρικά τανκς ενώ κινούμαστε γύρω από το κάστρο με το δικό μας τανκ.

Gameplay

# ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ

Πρόκειται για ένα Multiplayer παιχνίδι, όπου δύο διαφορετικοί παίκτες εφόσον βρίσκονται στο ίδιο δίκτυο Wifi, συνδέονται από δύο διαφορετικούς υπολογιστές. Επιλέγει ο καθένας ξεχωριστά το βαθμό δυσκολίας (Easy, Medium, Hard), το tank που θα έχει και συνδέεται στο παιχνίδι.

Ένα ενδιαφέρον χαρακτηριστικό που υλοποιείται στην εφαρμογή, είναι η αυτόματη επιλογή των επιπέδων και του εναπομείναντος tank, για τον παίκτη που θα επιλέξει δεύτερος, καθώς ο παίκτης που επέλεξε πρώτος χρονικά έχει ήδη αποκλείσει τον δεύτερο από κάποιες επιλογές. Για παράδειγμα, αν ο ένας επιλέξει το μπλε τανκ, τότε αναγκαστικά, η μόνη επιλογή που δίνεται στον άλλο παίκτη, είναι το κίτρινο.

Η αρχική ιδέα του παιχνιδιού ήταν η υποστήριξη πολλών modes. Θα υπήρχε η δυνατότητα τόσο τα tanks να πολεμούν μεταξύ τους, όσο και ύπαρξη ενός δεύτερου mode. Εκεί, τα 2 tanks συνεργάζονται στην ίδια ομάδα και αντιμετωπίζουν εχθρούς που κάνουν spawn, με σκοπό να επιτεθούν στο κάστρο. Η υλοποίηση του δεύτερου mode με Client-Server αρχιτεκτονική, ήταν αρκετά πολύπλοκη, οπότε η ιδέα αυτή αφέθηκε. Παρόλα αυτά, οι κώδικες για το spawn τέτοιων sprites υφίστανται στον κώδικα, χωρίς όμως να δημιουργούν κάτι, όπως φαίνεται και στο παρακάτω στιγμιότυπο.

# ΧΡΗΣΗ & MECHANICS ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ

Παίκτης

Κίνηση παικτών:

Σφαίρες

Κάστρο

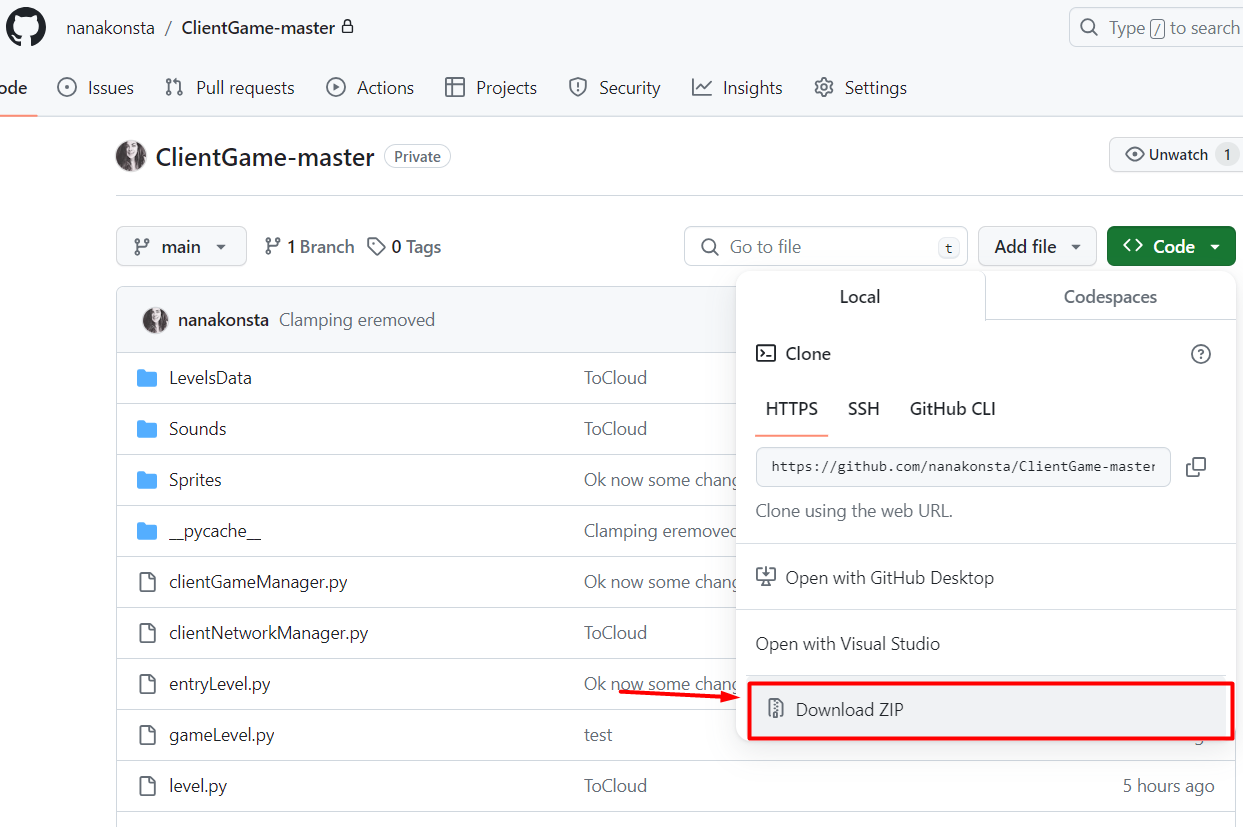
Έδαφος

# ΟΔΗΓΟΣ

Απαραίτητες προϋποθέσεις για την εγκατάσταση και την εύρυθμη λειτουργία της εφαρμογής:

# Αρχικά, γίνεται η λήψη του συμπιεσμένου αρχείου zip , τόσο από το [ClientGame Repository](https://github.com/nanakonsta/ClientGame-master), όσο και από το [ServerGame Repository](https://github.com/nanakonsta/ServerGame-master), μέσω github, όπως δείχνουν οι εικόνες:

# 



# Γίνεται

# λήψη

# ΣΚΟΡ, ΝΙΚΗ & ΗΤΤΑ

Στην αρχή, οι παίκτες ξεκινάνε με ένα συγκεκριμένο νούμερο στο health bar τους, το οποίο μπορεί να κατέβει αν τους πετύχει ο αντίπαλος. Όταν το health μηδενίσει, οι παίκτες χάνουν. Ενώ αν μηδενίσει το health του αντιπάλου, τότε νικάνε. Παράλληλα ενσωματώσαμε και καταμέτρηση του σκορ και των πόντων που πετυχαίνουν, παρότι δεν έχει ρυθμιστεί να εκφράζεται ακόμη.

Υπάρχει δυνατότητα επέκτασης του κώδικα με το health του κάστρου, για μετέπειτα εξέλιξη του παιχνιδιού στο mode, όπου τα δύο τανκς είναι συμπαίκτες και προστατεύουν το κάστρο από npc αντιπάλους sprite που κάνουν spawn σε διάφορα σημεία του ταμπλό.

# ΙΔΙΑΙΤΕΡΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

Κάτι πολύ ιδιαίτερο που αναπτύχθηκε σε αυτό το παιχνίδι είναι η οργανωμένη αρχιτεκτονική του, πρώτον οσον αφορά το διαχωρισμό των κλάσεων που περιγράφουν συγκεκριμένα αντικείμενα (

**ΕΝΟΤΗΤΑ 2η**

# ΤΕΧΝΙΚΕΣ ΛΕΠΤΟΜΕΡΕΙΕΣ

ΕΚΚΙΝΗΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

ΓΡΑΦΙΚΑ:

Σε πρώτο στάδιο, υλοποιήθηκε η κατασκευή απλών γραφικών και χρωμάτων, βασικών για τη λειτουργία της εφαρμογής.

# ΑΡΧΙΤΕΚΤΟΝΙΚΗ

# CLIENT & SERVER

# ΕΠΕΞΗΓΗΣΗ ΤΟΥ ΚΩΔΙΚΑ & ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ

Όπως φαίνεται και από το διαχωρισμό των αρχείων .py σε δύο φακέλους (Client Game master και Server Game master), στην ουσία η εφαρμογή αποτελείται από δύο porjects, τον client και τον server. Αν και αρκετές κλάσεις είναι κοινές και στα δύο, υπάρχουν κάποιες σημαντικές διαφορές που καλό είναι να τονιστούν.

* Η υλοποίηση βασικών αντικειμένων που βλέπει ο παίκτης στον υπολογιστή του και η γενικότερη δημιουργία του παραθύρου, γίνεται από τον Client. Δηλαδή, τα αντικείμενα δημιουργούνται τοπικά.

Φυσικά, έχουμε προβλέψει την αποστολή των δεδομένων από τον Client στον Server, αλλά εν γένει, τα στοιχεία δημιουργούνται στον Client και αποστέλλονται στον Server, με χρήση αρχείων json.

* Όλα τα δεδομένα, όπως όνομα username, password, ταυτόχρονη αντίληψη και εμφάνιση του αντιπάλου στέλνονται και αποθηκεύονται στον server.
* Για τη λειτουργία του παιχνιδιού, πρώτα ξεκινά να ‘’τρέχει’’ ο server και μετά ο client.
* Αν ο κάθε παίκτης συνδεθεί από διαφορετικό υπολογιστή, απαιτείται μόνο ένα run του project client game. Για να εμφανιστούν και τα δύο παράθυρα παικτών στον ίδιο υπολογιστή, όμως, απαιτείται ένα run και μετά ένα debug python file.

Ακολουθεί συνοπτική περιγραφή της λειτουργικότητας των διαφορετικών αρχείων, κλάσεων, μεθόδων και συναρτήσεων που περιέχονται στα server και client projects.

Α) SERVER

Η γενική λειτουργία του server είναι να λαμβάνει δεδομένα από τους χρήστες και να τα στέλνει πίσω. Τέτοια δεδομένα μπορεί να είναι, ποιο χρώμα tank διάλεξε ο παίκτης, που βρίσκεται ο αντίπαλος, η κατάσταση του health bar κλπ. Ας εμβαθύνουμε στη λειτουργικότητα όλων των αρχείων της μορφής .py που βρίσκονται στον Server.

1. Κλάση databaseHandler
2. dataHandler
3. difficultyEnum
4. gamedatabase
5. gameManager
6. gameMode
7. lobby
8. lobbyMode
9. main
10. manager
11. mode
12. player
13. serverGameRunner
14. serverManager
15. tank
16. tankEnum
17. tankFactory
18. test

LEitourgeia server o server ksekinaei kai exei ena parathuro

pou trexei to kentriko nhma kai 2 nhmata ena gia kathe paixth(nhma container epeksergasias tston upologisth)

auata ta duo nhmata einai upeuthina gia na paroun kai na dwsoun data ston paikth

>> h klash database handler einai gia na pairnei kai na dinei erwtrhmata sthn vash stelnei dedomena(requests sthn vash)

>> h klash lobbymode sunchronizei ta dedomena sto

>>lobby twn paixtwn thn epilegmenh duskolia to epilegmeno tank kai to ready

>> to difficulty enum einai ousiastika ena enumeration gia thn duskolia tou paixnidiou exei tria value easy medium kai hard

analoga me thn duskolia tou paixnidiou o paixths exei mikrotero h megalitero firerate kai zwh

>> gamedatabase einai t arxeio thn vcaseis dedomenwn to aarxeia stelnontai

>> to lobby einai voithitikh klash to lobby mode kaim krataei dedomena json estw na mporei na ta stielei to lobby mode stous client

>> GameManager einai mi aklash voithitikhjh opoioa krataei kata thn diarkei tou paixnidiou dedomena ta opoia Xrhsimopoiountai

se polles klaseis opws poio paixths exei pio tank me poio username poso firerate kai oposo zwh exei jo kathe paixths

poso zwh exei epishs kapoia instance ths klaseis paixtrhs h opoio krataei kapoia alla dedomena

>> h klash player exei username vehicle an einai o prwtos h deuteros paixths pou sundeetai poso zwh exei kai to firerate

>> mode einai h base kalsh gia ola ta mode krataei dedomena opws megethos parathurou xrwma data paixth opws einai to json des datap1

sthn periptwsh auths ths klashs auto to arxeio kanei anathsesh timhs sto runtime (otan ksekinas na pazizeis to paixnidi)

>> to gamemode einai h klash pou poaizei to paixnidi dhladh ekei pou stelnontai data gia thn topothesia ton paixtwn tou kastrou to exthwn

to ana kapoios exei varesei me to kanoni to ana kapoios exei xasei zwh kai outw katheksh

>> h main einaiaa po ekei pou kszekinaei na trexei to programmat tou server tou paixnidiou

kai kaloi tiw alles ServerGameRunner

>>tanks uparxoun 3 kloaseis sto arxeio tank.py h mia klash

einai h base kai exei kapoia dedomena genika gia to tank opws xy width kai heigh uparxei meta mia klkasei to tank base

to opio krataei thn eikona ths vvashs tou tank kai kinei tyo tank to tank turret einai mia kllash h opoia kinei to turret

kai to kanei rotate analoga me to pou einai to pontiki sthn othonh epeishs exei mia method firfe cannon h opoioa methodos

einai upoeuthunei gia na riksei vlhmata

epeish stis treis autes klaseis uparxei o sunxronsimos me to diktuo sthn ousia kataskeuazetai ena antikeimeno to opoio stelnetai sto diktuo

>> ServerGameRunner trexei to vasiko parathuro tou paixidiou einai einai upeuthiuo gia na parie kai na dwsei dedomena

opws einai ta events sthn periptwsh tou server den xrhsimopoiuntai kapou dioti den uparxei input apo to paikth

to

>> h klash manager sketo einai mh hlopoihmenh skopos htane na afairei kai na prosthetei paiktes sto paixnidi px pethane o enas vgalton teleios apo to paixnidi

>> Sevrer manager base gia na pairnei kai stelnei json polu vasikh methodos

receiveDataAndTakeAction

pairnei to prwto node tou json kai analogfa me thn odhgia kanei ena action pc sign in pairnei to usernbma ekai to

password kjaneio ena erwthma sthn meswq databnse handler kai epistrefei se mopfh jsonm analoga to ana exei paei kala to sign

in h oxi mhnhma paradeigma allo fethllobby pairni ta x y olwn twn ui entikeimenwn kai ta dinei stron client

opws epeish kai plhroforia gia to ana as pou h duskolia exie kleidwsei sthn peritwsh

pou h duskolia einai kleidwmenh stou clients emfanizetia me diaforetimo xrwma to ui th s duskolia easy hmediuy

hard

=>tankFactory.py Auth h klash einai upeuthunh gia na kataskeuazei objects tupou tasnk pc oi extroi exoun allaxarhskthtistk aopws

diafopretiko sprite allh zwh allh firetate klp

uparxei h methodos contrusct tanlk h opoia auto pou kanei einai analoga me to th dexetai os parametro se tank enum na

dhmiourtgei antikeimeno tetoio tupou

>> class TankEnum(Enum):

P1 = 1

P2 = 2

E1 = 3

E2 = 4

enumeration gia to tank P1 = player = 1 P2 = player = 2 E1 = enemy1

Β) CLIENT

CLIENT:

>>clientGameManager: ελέγχει το level του παίκτη, υπεύθυνη για να αλλάξει το level

>>clientNetworkManager: δίνει και παίρνει πληροφορίες του παίκτη και τ απαει στο δικτυο, αυτή κάνει τη σύνδεση με το server

>>entrylevel: ειναι το 1ο level με το οποιο συνδεεται, εμφανιζει το παραθυρο, πατας το connect και σε συνδεει

>>Gamelevel: ειναι η κεντρικη πιστα του παιχνιδιου, αυτή ζωγραφιζει το παραθυρο, τους παικτες, το κάστρο, τα παντα...

και στελνει και παιρενι δεδομενα αναλογα με την τοποθεσια των παικτων

(εδω τροποποιουμε το μεγεθος παραθυρου, θεση αρχικη παικτη)

>> level: ιδιο περιπου με αυτην του server, είναι h base κλαση gia ta υπολοιπα level

>> levelManager: δεν υλοποιηθηκε

>> lobbylevel: η κλαση που εχει το παραθυρο που ζωγραφιζει το lobby οπου επιλεγεις επιπεδα κ ready

και στελνει τα αντιστοιχα δεδομενα στο server

>> main: τρεχει ολο το παιχνιδι από εδωιδια με του server παρομοια λειτουργια απλα γ τον client,

>> manager: half

>> signInLevelSignup: εχει 2 κλασεις- ειναι αρχειο- εδω σχηματιζεται το παραθυρο που κανεις

sign in, sign up και στελνει τα αντιστοιχα δεδομενα στο server

>> player: ιδια με του server, αρχιτεκτονικη τυπου client server & client authorotive

>>tank ομοια με server

>> tankFactory: dhmioyrgei to τανκ (bazv κ url)

# ΠΡΟΚΛΗΣΕΙΣ & BUGS

Κατά τη διάρκεια υλοποίησης του application, όπως και σε κάθε δημιουργική διαδικασία, εμφανίστηκαν κάποιες δυσκολίες, κυρίως όσον αφορά το συντονισμό Server και Client. Ειδικότερα, λόγω της αρχιτεκτονικής του παχνιδιού

**ΕΝΟΤΗΤΑ 3η**

# ΙΣΤΟΡΙΚΟ ΥΛΟΠΟΙΗΣΗΣ

Με την αρχική ημερομηνία ανάθεσης, 17 Οκτωβρίου

# ΠΕΡΑΙΤΕΡΩ ΕΞΕΛΙΞΗ

# ΣΥΜΠΕΡΑΣΜΑΤΑ

Ολοκληρώνοντας την παρουσίαση και περιγραφή του παιχνιδιού που υλοποιήσαμε, προκύπτουν κάποιες συμπερασματικές σκέψεις.

Η Pygame, αλλά και η ίδια η Python, αποτελεί ένα πολύτιμο εργαλείο για τον εν δυνάμει game developer, λόγω της ευκολίας που παρέχεται στην εκμάθηση και κατανόηση του κώδικα, των βιβλιοθηκών και των εντολών. Παρόλα αυτά, ενδείκνυται περισσότερο για 2D παιχνίδια. Κρίνοντας επίσης από την πολυπλοκότητα και το χρόνο που απαιτήθηκε για την αρχιτεκτονική Client Server

Η ενασχόληση με το project αυτό, αποτέλεσε ένα καλό εναρκτήριο λάκτισμα, για τη βαθύτερη ενασχόληση με τον προγραμματισμό, ειδικότερα τον κλάδο κατασκευής παιχνιδιών. Σίγουρα έχουμε πολλά ακόμη να μάθουμε και, όπως αναμένεται με κάθε δημιουργική διαδικασία, τα λάθη, τα bugs, οι δυσκολίες, είναι κομμάτι της διαδρομής αυτής

Επιπλέον, η

Παιδαγωγική σημασία

Καλύτερη κατανόηση του προγραμματιστικού τρόπου σκέψης

Εν κατακλείδι, ακόμη και αν το παιχνίδι έμεινε εν μέρει ελλιπές και κάποιοι αρχικοί στόχοι εν τέλει δεν υλοποιήθηκαν (με τους κώδικες που αναφέρθηκαν πιο πάνω), τα ακόλουθα απαιτούμενα έχουν υλοποιηθεί ως εξής:

* Ύπαρξη τουλάχιστον ενός παίκτη, για την ακρίβεια δύο παικτών με διαφορετικά χαρακτηριστικά
* Ύπαρξη ταμπλό – πίστας, όπου βρίσκονται ένα κάστρο και 2 τανκς.
* Ύπαρξη τριών επιπέδων δυσκολίας (Easy, Medium, Hard)
* Αποθήκευση σκορ για τον παίκτη και «ανέβασμα» στον server
* Ύπαρξη γραφικών, όπως είναι τα τανκς που

# ΕΥΡΕΤΗΡΙΟ ΟΡΩΝ

Σε αυτό εδώ το παράρτημα, αναφέρονται εξειδικευμένες ορολογίες και έννοιες.

* Αρχείο [JSON](https://realpython.com/lessons/what-is-json/) (JavaScript Object Notation): είναι μία δημοφιλής μορφή αρχείου, που χρησιμοποιείται για την αποθήκευση προσωρινών δεδομένων, όπως δεδομένα ιστοσελίδων ή διαδικτυακών εφαρμογών. Αποτελεί συνήθης τακτική, η χρήση αρχείων json για τη μετάδοση και παραλαβή δεδομένων (σε μορφή κειμένου) μεταξύ ενός server και μιας δικτυακής εφαρμογής. Τα αντικείμενα σε ένα αρχείο json, αντιπροσωπεύονται ως ζευγάρια name/value. Με την εντολή *import json* στην Python, μπορούμε να αρχίσουμε να χρησιμοποιούμε αρχεία json.
* [Factory Pattern/Method](https://medium.com/@digit.sensitivee/game-programming-patterns-the-factory-method-87d7f8c12081): είναι ένα σχεδιαστικό πρότυπο, που χρησιμοποιείται κυρίως για τη δημιουργία αντικειμένων (objects), χωρίς να καθορίζουμε ακριβώς την κλάση του αντικειμένου που θα δημιουργηθεί. Συνήθως, δημιουργούμε μία γενική αφαιρετική κλάση, η οποία περιέχει τις μεθόδους που δημιουργούν αντικείμενα, συγκεκριμένες κλάσεις που την επεκτείνουν και αντιπροσωπεύουν τους τύπους των αντικειμένων, μια διεπαφή ή κλάση τύπου Factory που δηλώνει τη μέθοδο για τη δημιουργία αντικειμένων της διεπαφής και είναι υπεύθυνη για δημιουργία περιπτώσεων, συγκεκριμένες κλάσεις τύπου Factory που είναι υπεύθυνες για τη δημιουργία συγκεκριμένου τύπου αντικειμένου και ένα αρχείο τύπου client code, που καλεί τη μέθοδο factory για να δημιουργήσει objects instances, χωρίς να χρειάζεται να γνωρίζει μια συγκεκριμένη κλάση ή συγκεκριμένο αντικείμνενο.
* [Gameplay](https://en.wikipedia.org/wiki/Gameplay): η γενικότερη εμπειρία που βιώνει ένας παίκτης παίζοντας ένα παιχνίδι. Συνδέεται με στοιχεία όπως: τα mechanics του ίδιου του παιχνιδιού, τους κανόνες, τις προκλήσεις, τους στόχους, το flow και τη γενικότερη διαδικασία απορρόφησης και διάδρασης του παίκτη στον κόσμο του παιχνιδιού.
* [Sprite](https://en.wikipedia.org/wiki/Sprite_(computer_graphics)): 2D εικόνα ή animation που ενσωματώνεται σε μια σκηνή. Συχνά τα sprites χρησιμοποιούνται για να απεικονίσουν χαρακτήρες, παίκτες, npc, αντικείμενα και γενικά οποιοδήποτε είδους γραφικό.
* [Pygame](https://www.pygame.org/wiki/about): είναι ένα γκρουπ από modules της Python, το οποίο περιέχει ειδικές συναρτήσεις, μεταβλητές και κλάσεις, σχετικές με την ανάπτυξη παιχνιδιών χρησιμοποιώντας την προγραμματιστική γλώσσα Python