



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
UNIVERSITY OF WEST ATTICA

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΤΙΤΛΟΣ:

**ΑΝΑΠΤΥΞΗ ΠΑΙΧΝΙΔΙΟΥ ΜΑΘΗΣΗΣ ΓΙΑ ΕΠΙΤΡΑΠΕΖΙΟΥΣ ΥΠΟΛΟΓΙΣΤΕΣ
ΚΡΕΜΑΛΑ**

ΜΠΟΥΛΛΟ ΝΤΕΝΙ, CSE/43353

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ:

ΔΗΜΗΤΡΗΣ ΚΕΧΑΓΙΑΣ

ΑΘΗΝΑ , ΜΑΡΤΙΟΣ 2020

Επιτελική Σύνοψη

Τα τελευταία χρόνια γίνεται όλο και πιο αισθητή η ανάγκη για τη προσαρμογή και την αφομοίωση των καινούργιων μέσων τεχνολογίας στην καθημερινή ζωή των ανθρώπων στον τομέα της εκπαίδευσης. Η εκπαίδευση και ο τομέας της παιδείας καλείται να ενημερωθεί, να προσαρμοστεί σε αυτές τις τεχνολογίες και μέσα από αυτές να αναπτυχθεί. Αυτό το κάνει χρησιμοποιώντας κατά κύριο μέσω το Διαδίκτυο(WWW: World Wide Web), τους Υπολογιστές και τα κινητά τηλέφωνα. Οι εκπαιδευτικοί καλούνται εξίσου να παρουσιάσουν αυτά τα μέσα τεχνολογίας στα παιδιά με σκοπό να τα βοηθήσουν και να αυξηθεί ο αριθμός της απόδοσης και της αναπτυξιακής τους μάθησης. Αυτό μπορεί πολύ εύκολα να γίνει , για παράδειγμα, μέσω των εφαρμογών , άλλοτε στους Υπολογιστές και άλλοτε στα κινητά τηλέφωνα, σε μορφή παιχνιδιού. Στην παρακάτω εργασία θα ασχοληθούμε κυρίως με μια εφαρμογή παιχνιδιού που είναι διαθέσιμη για Υπολογιστές.

Λέξεις-Κλειδιά:

Υπολογιστές, Εκπαίδευση, Τεχνολογία, Εφαρμογές, Παιχνίδι, Παιδιά, Μαθησιακή Ανάπτυξη
Computers, Education, Technology, Applications, Games, Children, Learning Development

ΠΕΡΙΕΧΟΜΕΝΑ

Κεφάλαιο 1^ο

1.1 Πρόλογος.....	σελ.4
1.2 Περίληψη.....	σελ.4-5
1.3 Βασικοί Ορισμοί.....	σελ.5-6
1.4 Παραδοτέα της Εργασίας.....	σελ.6

Κεφάλαιο 2^ο

2.1 Ανάλυση Δεδομένων Ερευνών.....	σελ.6-7
2.2 Πλεονεκτήματα Μάθησης Μέσω Παιχνιδιού.....	σελ.7
2.3 Γιατί ένα παιχνίδι βοηθάει τα παιδιά να μαθαίνουν πιο εύκολα.....	σελ.7-8
2.4 Προσαρμογή στις νέες τεχνολογίες.....	σελ.8
2.5 Αρνητικές Βλέψεις πάνω στην εισαγωγή παιχνιδιού στην ζωή ενός παιδιού.....	σελ.8
2.6 Συμπέρασμα.....	σελ.9

Κεφάλαιο 3^ο

3.1 Τι είναι η κρεμάλα.....	σελ.9
-----------------------------	-------

Κεφάλαιο 4^ο

4.1 Περιγραφή διεπαφής χρήστη (user interface).....	σελ.9-16
4.1.1 Αρχικό Μενού.....	σελ.9-10
4.1.2 Έναρξη Μενού.....	σελ.11-12
4.1.3 Παράθυρο Παιχνιδιού.....	σελ.12-13
4.1.4 Παράθυρο Νίκης.....	σελ.14
4.1.5 Παράθυρο Ήττας.....	σελ.15
4.1.6 Παράθυρο Ρεκόρ.....	σελ.16

Κεφάλαιο 5^ο

5.1 Εγχειρίδιο Κώδικα Παιχνιδιού(documentation).....	σελ.16-17
5.2 Δομή του παιχνιδιού.....	σελ.17-21
5.2.1 Δομή Core	σελ.17-18
5.2.2 Δομή Platform.....	σελ.19
5.2.3 Δομή Opendl.....	σελ.19

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

5.2.4 Δομή AssetFrameWork.....σελ.20	σελ.20
5.2.5 Δομή Renderer.....σελ.20	σελ.20
5.2.6 Δομή GameFrameWork.....σελ.20-21	σελ.20-21
5.3 Ανάλυση Δομών.....σελ.21-125	σελ.21-125
5.3.1 Δομή Coreσελ.21-49	σελ.21-49
5.3.2 Δομή Platform.....σελ.49-58	σελ.49-58
5.3.3 Δομή Opengl.....σελ.59-73	σελ.59-73
5.3.4 Δομή Memory.....σελ.73-78	σελ.73-78
5.3.5 Δομή GameFrameWork.....σελ.78-94	σελ.78-94
5.3.6 Δομή Renderer.....σελ.94-125	σελ.94-125

Κεφάλαιο 6^ο

6.1 Βιβλιογραφία-Πηγές-Αναφορές.....σελ.126-127	σελ.126-127
---	-------------

ΚΕΦΑΛΑΙΟ 1^ο : ΕΙΣΑΓΩΓΗ

1.1 Πρόλογος

Η παρούσα εργασία αποτελεί το έντυπο παραδοτέο μέρος της Πτυχιακής Εργασίας με θέμα την ανάπτυξη εφαρμογής παιχνιδιού μάθησης για υπολογιστές για μικρά παιδιά. Η εργασία αυτή πραγματοποιήθηκε στα πλαίσια της εκπόνησης της Πτυχιακής εργασίας του φοιτητή του τμήματος Μηχανικών Πληροφορικής και Υπολογιστών του Πανεπιστημίου Πειραιά, Μπούλλο Ντένι, υπό την επίβλεψη του Καθηγητή του τμήματος, κου Δημήτρη Κεχαγιά.

1.2 Περίληψη

Σκοπός αυτής της εργασίας είναι ο σχεδιασμός, η ανάπτυξη και η υλοποίηση ενός παιχνιδιού μάθησης για μικρά παιδιά. Το παιχνίδι θα έχει τα βασικά χαρακτηριστικά της «Κρεμάλας» με κάποιες επιπλέον λειτουργίες και θα προσφέρει τη δυνατότητα σε μικρά παιδιά να παίζουν και παράλληλα να μαθαίνουν,σε θέματα όπως, ιστορία, γεωγραφία, μουσική κλπ. Η εφαρμογή θα αναπτυχθεί σε περιβάλλον C και C++ μέσω του API της Opengl.

The purpose of this thesis is the design, the development and the implementation of a learning game for young children. The game will have the basic features of “HangMan” with some additional functions and will offer the opportunity to young children to play and learn at the same

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

time like on topics such as history, geography, music etc. The application will be developed in C and C ++ environment through the OpenGL API.

1.3 Βασικοί Ορισμοί

- **Κρεμάλα:** Κρεμάλα είναι παιχνίδι του οποίου σκοπός είναι η αναζήτηση κρυφών λέξεων, γράμμα προς γράμμα από τους παίκτες. Είναι διαχρονικό και παίζεται από όλες τις ηλικίες.
- **Εφαρμογή:** Ένα πρόγραμμα εφαρμογής (app ή application για συντομία) είναι ένα πρόγραμμα υπολογιστή, tablet ή smartphone σχεδιασμένο για να εκτελεί μία ομάδα συντονισμένων λειτουργιών, έργων ή δραστηριοτήτων προς όφελος του χρήστη.
- **C++:** Είναι μία γενικού σκοπού γλώσσα προγραμματισμού Η/Υ. Θεωρείται μέσου επιπέδου γλώσσα, καθώς περιλαμβάνει έναν συνδυασμό χαρακτηριστικών από γλώσσες υψηλού και χαμηλού επιπέδου. Είναι μια μεταγλωττιζόμενη γλώσσα πολλαπλών παραδειγμάτων, με τύπους. Υποστηρίζει δομημένο, αντικειμενοστρεφή και γενικό προγραμματισμό. Η γλώσσα αναπτύχθηκε από τον Μπιάρνε Στρούστρουπ το 1979.
- **C:** Είναι μια διαδικαστική γλώσσα προγραμματισμού γενικής χρήσης, η οποία αναπτύχθηκε αρχικά, μεταξύ του 1969 και του 1973, από τον Ντένις Ρίτσι στα εργαστήρια AT&T Bell Labs για να χρησιμοποιηθεί για την ανάπτυξη του λειτουργικού συστήματος UNIX.
- **API:** Η Διεπαφή Προγραμματισμού Εφαρμογών (αγγλ. API, από το Application Programming Interface), γνωστή και ως Διασύνδεση Προγραμματισμού Εφαρμογών (για συντομία διεπαφή ή διασύνδεση), είναι η διεπαφή των προγραμματιστικών διαδικασιών που παρέχει ένα λειτουργικό σύστημα, βιβλιοθήκη ή εφαρμογή προκειμένου να επιτρέπει να γίνονται προς αυτά αιτήσεις από άλλα προγράμματα ή/και ανταλλαγή δεδομένων.
- **OpenGL:** Open Graphics Library (OpenGL) είναι μια Διεπαφή Προγραμματισμού Εφαρμογών για την απόδοση 2D και 3D Γραφικών. Είναι ανεξάρτητο λειτουργικού

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

συστήματος και συνήθως αλληλεπιδρά με την κάρτα γραφικών ώστε να πετύχει ταχύτερη απόδοση γραφικών. Χρησιμοποιείται εκτενώς στα εργαλεία σχεδίασης μέσω υπολογιστή (αγγλ. CAD), στα βιντεοπαιχνίδια, στην εικονική και επαυξημένη πραγματικότητα, σε εξομοιωτές πτήσης, και στις επιστημονικές οπτικοποιήσεις. Το OpenGL αναπτύχθηκε το 1991 από την Silicon Graphics (SGI) και από τον Ιούλιο του 2006 αναπτύσσεται από την μη-κερδοσκοπική κοινοπραξία Khronos Group.

1.3 Παραδοτέα της Εργασίας

Στην ενότητα αυτή αναφέρουμε επιγραμματικά τα παραδοτέα της εργασίας. Με τον όρο παραδοτέα εννοούμε οτιδήποτε συμπεριλαμβάνεται στην εργασία. Τα παραδοτέα της συγκεκριμένης εργασίας είναι:

- Το παρόν έντυπο κείμενο της πτυχιακής εργασίας, το οποίο περιλαμβάνει την επισκόπηση της σχετικής βιβλιογραφίας, τη περιγραφή του σχεδιασμού και της υλοποίησης της εφαρμογής.
- Το εγχειρίδιο με τις ανάλογες μεθόδους και τον πηγαίο κώδικα που χρησιμοποιήθηκαν για τον προγραμματισμό της εφαρμογής παιχνιδιού .
- Η συλλογή πηγών και σχετικής βιβλιογραφίας για τη δημιουργία μίας βάσης γνώσης σχετικά με το θέμα.

ΚΕΦΑΛΑΙΟ 2^ο

2.1 Ανάλυση Δεδομένων Ερευνών

Με τον όρο Ανάλυση Δεδομένων Ερευνών αναφερόμαστε στα αποτελέσματα ερευνών πάνω στην Αναπτυξιακή Μάθηση των παιδιών στα πλαίσια χρησιμοποίησης εφαρμογών παιχνιδιού στους Υπολογιστές.

Πανεπιστήμιο Δυτικής Αττικής Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Σύμφωνα με διάφορες έρευνες που έχουν πραγματοποιηθεί, έχουν συλλεχθεί τα παρακάτω δεδομένα:

Τα ηλεκτρονικά παιχνίδια έχουν παρατηρηθεί ότι βοηθάνε τη μάθηση με τρόπους που διατηρούν τους παίκτες στην άκρη των καθισμάτων τους, προωθώντας το συνεχές ενδιαφέρον για το παιχνίδι για ώρες, εβδομάδες, ακόμη και χρόνια. Οι παίκτες ακονίζουν τις δεξιότητές τους και χτίζουν γνώσεις όσο συνεχίζουν να παίζουν. Σε μερικές σπάνιες περιπτώσεις, οι προγραμματιστές παιχνιδιών, όπως η Valve (βλ. Valve Corporation, 2007, Valve Corporation, 2011), έχουν περιγράψει το αποτελεσματικό σχεδιασμό του πλαισίου της «πολυεπίπεδης μάθησης» που επιχειρεί να βελτιστοποιήσει τα μαθησιακά στοιχεία σύμφωνα με τις αλληλένδετες αρχές της πρόκλησης, των δεξιοτήτων, της εμπλοκής και βύθιση. Σε αυτό το πλαίσιο, η δέσμευση και η μάθηση είναι απαραίτητες για να διατηρήσουν τους παίκτες την πρόοδο στο παιχνίδι, και το αντίστροφο.

2.2 Πλεονεκτήματα Μάθησης Μέσω Παιχνιδιού

Η Μάθηση βάσει παιχνιδιού είναι η διαδικασία της σκέψης και της μηχανικής του παιχνιδιού για την προσέλευση χρηστών και την επίλυση προβλημάτων. Τα παιχνίδια μπορούν να βοηθήσουν στη μάθηση επειδή οι άνθρωποι είναι περισσότερο :

- Παρακινημένοι
- Δεσμευμένοι
- Και έχουν την ψευδαίσθηση ότι μπορούν να πετύχουν περισσότερα στα παιχνίδια από ότι στον πραγματικό κόσμο

Για το λόγο αυτό τα παιχνίδια προγραμματίζονται έτσι ώστε να καταφέρουν να αυξήσουν το επίπεδο αφομοίωσης των χρηστών με πηγές βιβλιοθηκών και να βοηθήσουν τους χρήστες να λύσουν προβλήματα πιο αποτελεσματικά και γρήγορα.

Η αρχιτεκτονική ενός παιχνιδιού είναι σχεδιασμένη έτσι ώστε να επιβραβεύει τους χρήστες του με αποτέλεσμα οι χρήστες να παραμένουν συνεχώς κινητοποιημένοι. Καθώς τα στάδια του παιχνιδιού ολοένα και μεγαλώνουν αυξάνεται και ο βαθμός δυσκολίας. Αυτό έχει σαν αποτέλεσμα να συνεχίζει να αυξάνει το ενδιαφέρον των χρηστών και να αποδίδουν περισσότερο. Με αυτό τον τρόπο αυξάνεται και το επίπεδο μάθησης.

2.3 Γιατί ένα παιχνίδι βοηθάει τα παιδιά να μαθαίνουν πιο εύκολα

Τα περισσότερα παιδιά δυσκολεύονται να κρατήσουν το ενδιαφέρον τους για ένα θέμα που τους παρουσιάζεται από τους εκπαιδευτικούς, αν αυτό δεν το θεωρούν διασκεδαστικό. Στο σχολείο και στο σπίτι δυσκολεύονται να διαβάσουν καθώς το θεωρούν αγγαρεία. Εδώ αναλαμβάνουν την θέση στο προσκήνιο και κάνουν την εμφάνιση τους τα παιχνίδια. Ένα παιδί

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

δεν μπορεί να πει όχι όταν του τίθεται η ερώτηση «Θέλεις να παίξουμε». Η απάντηση θα είναι σχεδόν πάντα «Ναι». Πόσο μάλλον αν αυτό το παιχνίδι είναι διασκεδαστικό και μέσα από αυτό μπορεί να επιβραβεύεται. Σε αυτό το σημείο καλούνται οι προγραμματιστές παιχνιδιών να αναπτύξουν ένα παιχνίδι αρκετά διασκεδαστικό για τα παιδιά , αλλά και ταυτόχρονα εποικοδομητικό. Πρέπει μέσα από αυτό να αναπτύξει το παιδί τις γνώσεις του και να μάθει πράγματα που πριν δεν ήξερε. Μέσα από τις επιβραβεύσεις το παιδί νιώθει περισσότερο σπουδαίο και έτσι συνεχίζει να θέλει να αναπτυχθεί. Αυτό είναι θετικό γιατί στο χώρο της διδασκαλίας , οι εκπαιδευτικοί χάνουν τον ρόλο τους της περισσότερες μέσα στην καθημερινότητα και ξεχνάνε να επιβραβεύσουν την απόδοση των παιδιών , με αποτέλεσμα τα παιδιά να νιώθουν ασήμαντα και έτσι να χάνουν το ενδιαφέρον τους για μάθηση.

2.4 Προσαρμογή στις νέες τεχνολογίες

Την εποχή αυτή που ζούμε καλούμαστε να συνυπάρχουμε και να διαχειριζόμαστε τις νέες τεχνολογίες που αναπτύσσονται λόγω του ότι διευκολύνουν την καθημερινότητά μας. Δεδομένου αυτού , είναι επιθυμητό και λογικό να μπορεί ο καθένας από εμάς να κατέχει αυτές τις γνώσεις. Το καλύτερο επομένως που θα μπορούσε να συμβεί , θα ήταν να μπορούν και τα παιδιά από μικρή ηλικία να κατέχουν και να διαχειρίζονται αυτά τα μέσα τεχνολογίας.

Η διατριβή ενός παιδιού με τους υπολογιστές μπορεί να επιφέρει θετικά αποτελέσματα και να αναπτύξει διάφορες δεξιότητες και ικανότητες που αργότερα μπορεί να αποδειχθούν χρήσιμες στην ζωή τους.

2.5 Αρνητικές Βλέψεις πάνω στην εισαγωγή παιχνιδιού στην ζωή ενός παιδιού

Πολλοί άνθρωποι θεωρούν ότι τα παιχνίδια και οι νέες τεχνολογίες είναι περισσότερο επιβλαβής στην ζωή των παιδιών. Κάποιοι ισχυρίζονται ότι ένα παιδί δεν θα έπρεπε να κοιτάζει συνέχεια την οθόνη ενός υπολογιστή ή ενός κινητού καθώς μπορεί αυτό να έχει συνέπειες στην υγεία του. Άλλοι πιστεύουν ότι ένα παιδί θα έπρεπε περισσότερο να διατηρήσει την παιδικότητά του και να μάθει να συναναστρέφεται με άλλα παιδιά και όχι με άψυχα αντικείμενα(όπως υπολογιστές, κινητά τηλέφωνα κ.λπ).

2.6 Συμπέρασμα

Οι απόψεις δίστανται και για αυτό δεν μπορεί κάποιος να συμπεράνει αν είναι θετικό ή όχι το να εισηγηθούν οι εφαρμογές παιχνιδιού στην ζωή ενός παιδιού. Το καλύτερο λοιπόν που θα μπορούσε θεωρητικά να συμβεί είναι να υπάρχει μια ισορροπία στο πώς θα συμβεί αυτό.

Κεφάλαιο 3^ο

3.1 Τι είναι η Κρεμάλα;

Ο αριθμός των παικτών της ποικίλλει από 2 έως 4. Τα βήματα που ακολουθούνται είναι σχετικά εύκολα. Αρχικά, σε μία κόλλα χαρτί ζωγραφίζεται μία κρεμάλα και δίπλα τοποθετούνται παύλες, όσες και τα γράμματα της λέξης που ψάχνει ο αντίπαλος. Ύστερα, ο αντίπαλος λέει ένα γράμμα από την αλφαβήτα· αν το γράμμα που είπε ο αντίπαλος υπάρχει μέσα στη λέξη, τότε τοποθετείται το γράμμα στη παύλα που αντιστοιχεί. Αν όμως είναι λάθος, τότε σχεδιάζεται το κεφάλι του πάνω στη κρεμάλα και σημειώνεται το γράμμα σε μία γωνία του χαρτιού, ώστε να ξέρει ποια γράμματα έχει πει. Με αυτόν τον τρόπο λοιπόν συνεχίζεται το παιχνίδι, είτε μέχρι να βρει ο αντίπαλος τη λέξη ή μέχρι να χάσει, αν σχεδιαστούν όλα τα μέλη του σώματος και δεν έχει βρει ακόμη τη λέξη.

Σκοπός της Κρεμάλας

Ο σκοπός του παιχνιδιού της Κρεμάλας είναι να αναπτύξει το παιδί γενικές γνώσεις πάνω σε διάφορα θέματα, όπως για παράδειγμα στην γεωγραφία, ιστορία, μουσική κ.λπ.

Κεφάλαιο 4^ο

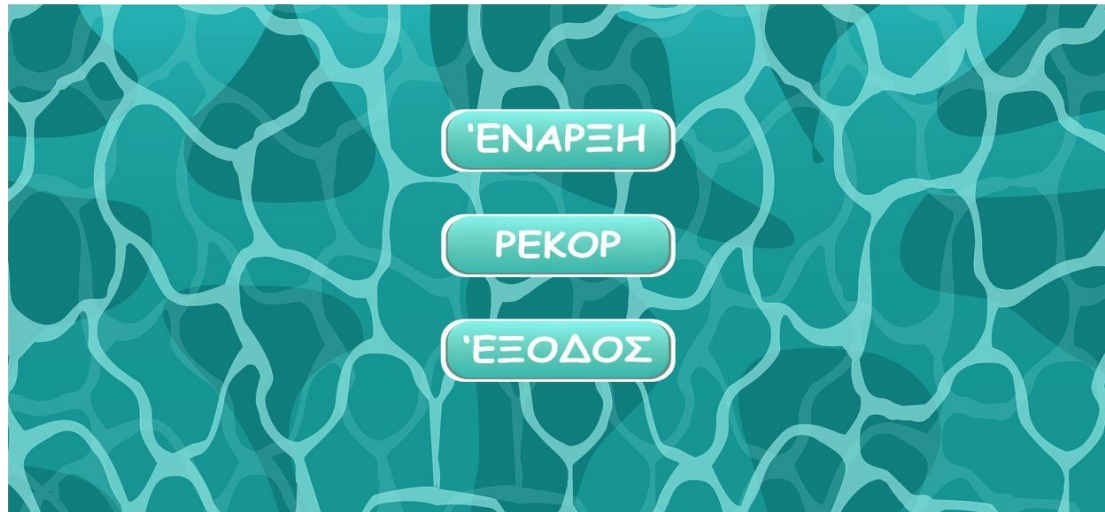
4.1 Περιγραφή διεπαφής χρήστη (user interface)

4.1.1 Αρχικό Μενού

Ο χρήστης όταν ανοίγει την εφαρμογή κατευθύνεται στο παρακάτω μενού που αποτελείται από τα παρακάτω επιλογές (κουμπιά):

- Έναρξη
- Ρεκόρ
- Έξοδος

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών



Τι συμβαίνει αν ο Χρήστης πατήσει το κουμπί Έναρξη :

Ο χρήστης κατευθύνεται στο επόμενο παράθυρο. Εκεί καλείται να επιλέξει τα παρακάτω:

- Φύλο: Αγόρι / Κορίτσι.
- Όνομα: Να συμπληρώσει το όνομά του (πρέπει να ξεκινάει με χαρακτήρα).
- Κατηγορία: Να επιλέξει κατηγορία του παιχνιδιού (π.χ ιστορία).
- Δυσκολία: Να επιλέξει τον βαθμό δυσκολίας της λέξης.

Τα παρακάτω κουμπιά :



: κατευθύνει τον χρήστη στο αρχικό μενού.

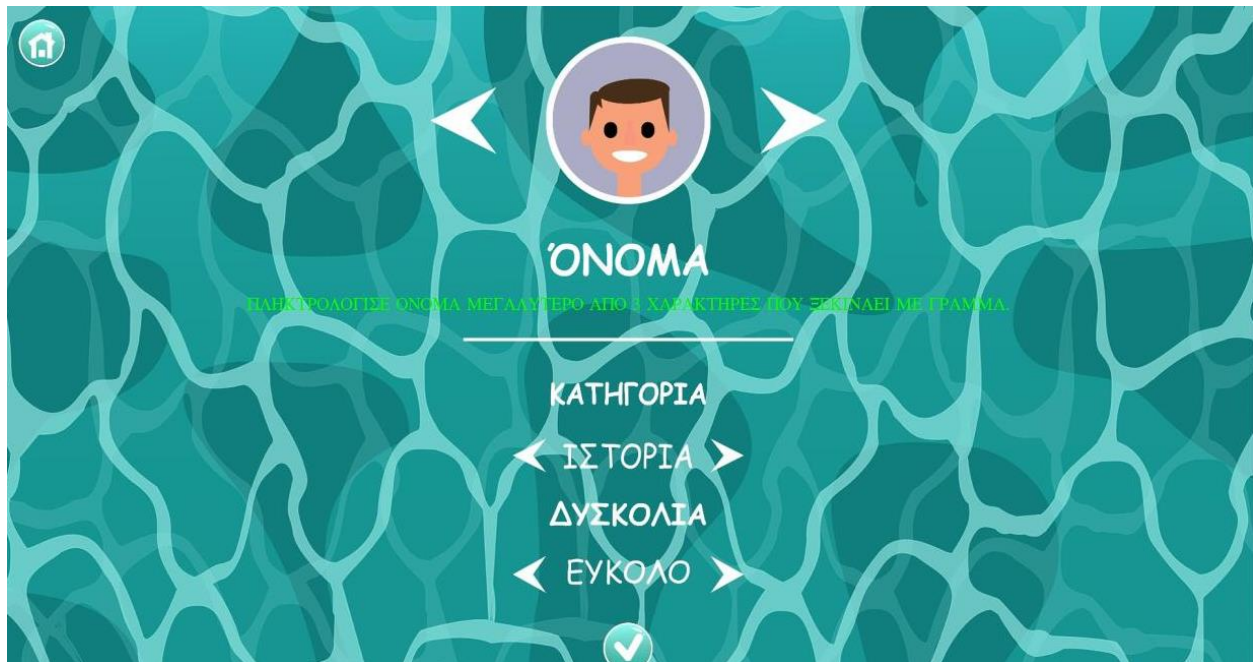


: κατευθύνει τον χρήστη στο μενού με τα ρεκόρ.



: τερματίζει την εφαρμογή.

4.1.2 Έναρξη Μενού



: ο χρήστης διαλέγει φύλο.



: ο χρήστης πληκτρολογεί όνομα

ανάλογα με τους παρακάτω κανόνες.

- Πρέπει το όνομα να ξεκινάει με γράμμα.
- Πρέπει το όνομα να περιέχει πάνω από 3 χαρακτήρες.



: ο χρήστης διαλέγει κατηγορία παιχνιδιού.



: ο χρήστης διαλέγει βαθμό δυσκολίας παιχνιδιού.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Κουμπιά :



: κατευθύνει τον χρήστη στο επόμενο παράθυρο μόνο αν ο χρήστης έχει πληκτρολογήσει έγκυρο όνομα.

4.1.3 Παράθυρο παιχνιδιού



Στο παραπάνω παράθυρο εμφανίζεται και το παιχνίδι.



: σκορ του παιχνιδιού.



: γύρος του παιχνιδιού.



: προσπάθειες που απομένουν στον χρήστη.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

- Εύκολο: δεν χάνει προσπάθειες.
- Μέτριο: χάνει μία προσπάθεια στην περίπτωση επιλογής λάθος γράμματος.
- Δύσκολο: το παιχνίδι ξεκινάει μόνο με μία μόνο διαθέσιμη προσπάθεια , η οποία χάνεται όταν ο χρήστης επιλέγει λάθος γράμμα.



: άγνωστη λέξη της

κρεμάλας



: κουμπί βοήθειας

Ανάλογα με τον βαθμό δυσκολίας , ο χρήστης μπορεί να ζητήσει βοήθεια. Μόνο αν το επίπεδο είναι δύσκολο , τότε η βοήθεια δεν είναι διαθέσιμη προς τον χρήστη.



Διαθέσιμα γράμματα της αλφαβήτου.

- Αν ο χρήστης επιλέξει σωστό γράμμα, το γράμμα αυτό πρασινίζει.
- Αν ο χρήστης επιλέξει λάθος γράμμα, το γράμμα αυτό κοκκινίζει.

Το κάθε γράμμα μπορεί να επιλεγθεί μόνο μία φορά.



: κατευθύνει τον χρήστη στο αρχικό μενού

4.1.4 Παράθυρο νίκης



Το παράθυρο αυτό εμφανίζεται στον χρήστη όταν βρει την λέξη της κρεμάλας.

Εκεί εμφανίζονται τα παρακάτω:

- Σκορ: το σκορ του παιχνιδιού.
- Ακολουθία νικών: πόσες σωστές λέξεις έχει βρει ο χρήστης χωρίς να κάνει λάθος.

Κουμπιά:

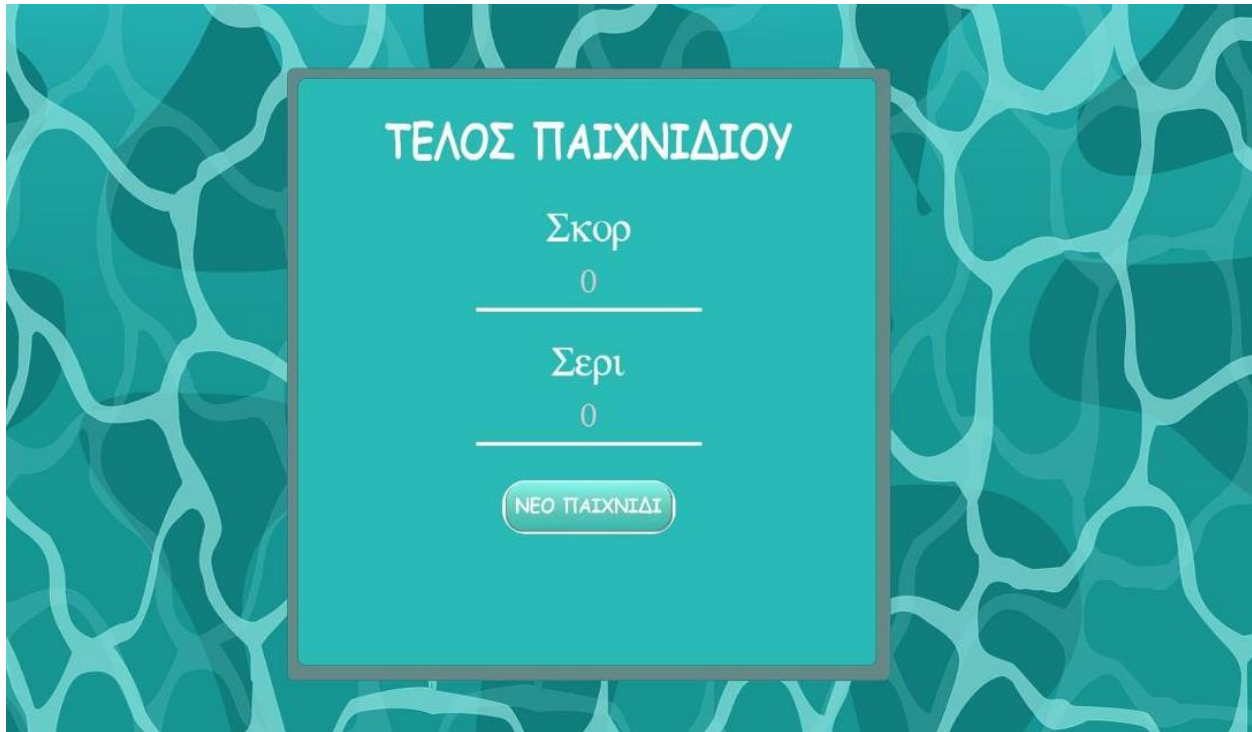


: ο χρήστης μπορεί να προχωρήσει στην επόμενη λέξη.



: κατευθύνει τον χρήστη στο αρχικό μενού.

4.1.5 Παράθυρο ήττας



Το παράθυρο αυτό εμφανίζεται στον χρήστη όταν δεν μπορέσει να βρει την λέξη της κρεμάλας.

Εκεί εμφανίζονται τα παρακάτω:

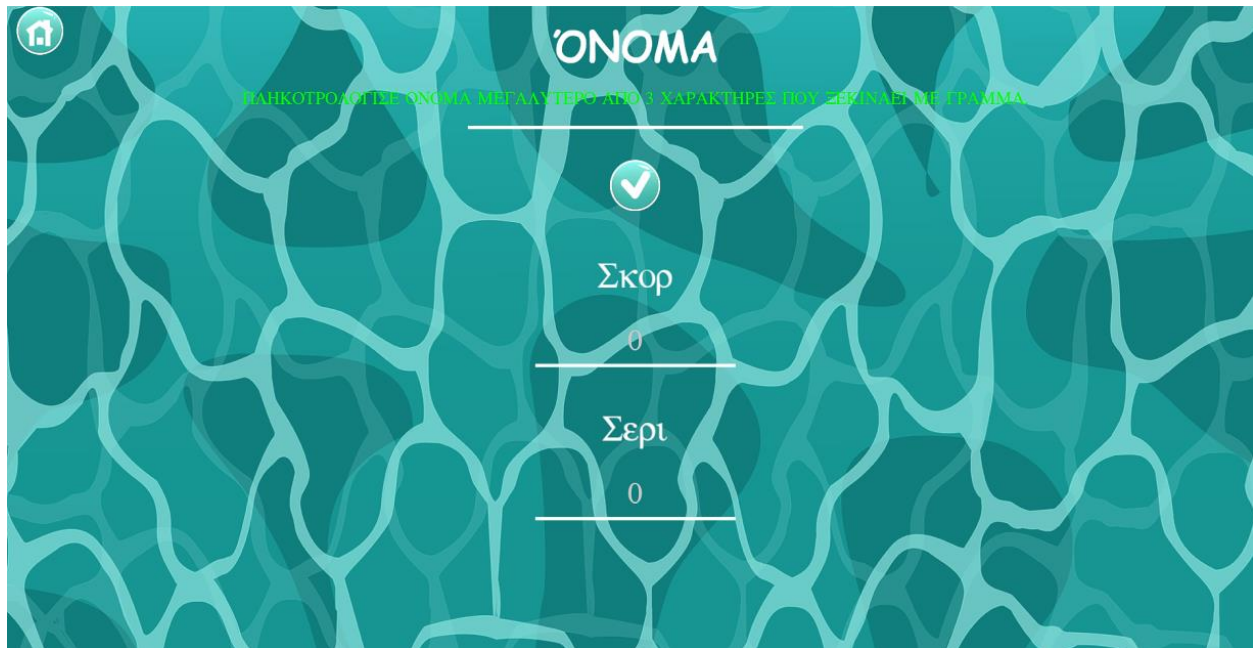
- Σκορ: το σκορ του παιχνιδιού.
- Ακολουθία νικών: πόσες σωστές λέξεις έχει βρει ο χρήστης χωρίς να κάνει λάθος.

Κουμπιά:



: ο χρήστης καλείται να φτιάξει καινούργιο παιχνίδι (εικόνα 1.2)

4.1.6 Παράθυρο Ρεκόρ



Ο χρήστης πρέπει να πληκτρολογήσει το όνομα του και να πατήσει το παρακάτω κουμπί



- Αν το όνομα του χρήστη είναι αποθηκευμένο, τότε εμφανίζονται τα αποτελέσματα.
- Αλλιώς εμφανίζεται μήνυμα ότι το όνομα είναι λάθος.

Εκεί εμφανίζονται τα παρακάτω:

- Σκορ: το σκορ του παιχνιδιού.
- Ακολουθία νικών: πόσες σωστές λέξεις έχει βρει ο χρήστης χωρίς να κάνει λάθος.

Κεφάλαιο 5°

Εγχειρίδιο Κώδικα Παιχνιδιού

Η δομή του παιχνιδιού αποτελείται από πέντε μέρη κώδικα.

Το **core**(σχετίζεται με συναρτήσεις τύπου αλφαριθμητικών(std::string), πινάκων(std::vector) μαθηματικών(Matrix), μακροεντολές(macros) κ.α εμπλουτισμένες με περισσότερες και πιο

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

αποδοτικές συναρτήσεις από αυτές που δίνονται από το σύστημα και εφόσον στην εργασία αυτή δεν θα χρησιμοποιηθεί σχεδόν καμία εξωτερική βιβλιοθήκη.

Το **Platform** σχετίζεται με την πλατφόρμα της εφαρμογής(στην συγκεκριμένη περίπτωση τα Windows), την δημιουργία παραθύρου και την αντιμετώπιση και λειτουργία των σχετικών συμβάντων που μπορούν να συμβούν σε αυτό.

Η **OpenGL** η οποία είναι η καρδιά της εφαρμογής παρέχει τα απαραίτητα εργαλεία και μεθόδους που θα χρειαστούν για την απεικόνιση του παιχνιδιού.

Ο **Renderer** είναι ουσιαστικά το μυαλό της εφαρμογής και το πιο σημαντικό κομμάτι κάθε παιχνιδιού (και γι' αυτό τον λόγο έχει δοθεί και μεγαλύτερη σημασία) σχετίζεται με τη μεθοδολογία λήψης και επεξεργασίας δεδομένων μεταξύ της κάρτας γραφικών και του πυρήνα του υπολογιστή και την απεικόνιση των αποτελεσμάτων στην οθόνη.

Τέλος το **GameFramework** σχετίζεται με την υλοποίηση της λογικής του παιχνιδιού Κρεμάλα.

Παρακάτω θα εξηγηθούν περιληπτικά τι κάνουν οι συναρτήσεις της δομής.

Καταρχάς υπάρχουν δυο τύποι αρχείων κώδικα στην εργασία τα οποία είναι:

- Τα αρχεία Header(Header.h) τα οποία αφορούν τις δηλώσεις των μεταβλητών συναρτήσεων, κλάσεων κ.λπ.
- Τα εκτελέσιμα(SourceCode.cpp) τα οποία αφορούν τους ορισμούς και την υλοποίηση των παραπάνω μεθόδων και μεταβλητών.

5.2 Δομή του παιχνιδιού

5.2.1 Δομή Core

Διακρίνεται σε

- Array.h
- Array.cpp
- CString.h
- CString.inl
- String.h
- String.inl
- Color32.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

- Color32.inl
- GenericMath.h
- IntPoint.h
- IntPoint.inl
- IntRect.h
- IntRect.inl
- Math.h
- Math.cpp
- Matrix.h
- Matrix.inl
- NumericLimits.h
- Transform.h
- Transform.inl
- UtilityMath.h
- Vector2D.h
- Vector2D.inl
- Vector3D.h
- Vector3D.inl
- Vector4D.h
- Vector4D.inl
- AllocatorInterface.h
- AllocatorInterface.inl
- FixedAllocator.h
- FixedAllocator.inl
- GenericMemory.h
- HeapAllocator.h
- HeapAllocator.inl
- InlineAllocator.h
- InlineAllocator.inl
- Memory.h
- StackAllocator.h
- StackAllocator.inl
- UtilityMemory.h
- CommonTypes.h
- CoreMinimal.h
- Definitions.h

5.2.2 Δομή Platform

Διακρίνεται σε

- Platform.h
- WindowsApplication.h
- WindowsApplication.cpp
- WindowsWindow.h
- WindowsWindow.cpp

5.2.3 Δομή OpenGL

Διακρίνεται σε

- EntryPoint.h
- EntryPoint.cpp
- OpenGLWindows.h
- OpenGLWindows.cpp
- OpenGL.h
- VertexBuffer.h
- VertexBuffer.cpp
- IndexBuffer.h
- IndexBuffer.cpp
- VertexShader.h
- VertexShader.cpp
- PixelShader.h
- PixelShader.cpp
- PipelineState.h
- PipelineState.cpp
- Texture2D.h
- Texture2D.cpp

5.2.4 Δομή AssetFrameWork

Διακρίνεται σε

- AssetFramework.h
- AssetFramework.cpp

5.2.5 Δομή Renderer

Διακρίνεται σε

- RenderResourceAllocator.h
- RenderResourceAllocator.cpp
- Renderer.h
- TileRenderer.h
- TileRenderer.cpp
- SpriteRenderer.h
- SpriteRenderer.cpp
- FontCashe.h
- FontCashe.cpp
- TextRenderer.h
- TextRenderer.cpp
- CanvasRenderer.h
- CanvasRenderer.cpp
- CanvasButton.h
- CanvasButton.cpp
- CanvasMenu.h
- CanvasMenu.cpp
- CanvasMenuState.h
- CanvasMenuState.cpp

5.2.6 Δομή GameFramework

Διακρίνεται σε

- GameCharCodeMap.h
- GameCharCodeMap.cpp
- GameCore.h
- GameFramework.h

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

- GameMainMenu.h
- GameMainMenu.cpp
- GamePlayMenu.h
- GamePlayMenu.cpp
- GameState.h
- GameState.cpp
- GameMenu.h
- GameMenu.cpp
- GameWinMenu.h
- GameWinMenu.cpp
- GameLoseMenu.h
- GameLoseMenu.cpp
- GameRecordsMenu.h
- GameRecordsMenu.cpp

5.3 Ανάλυση Δομών

5.3.1 Δομή Core

Προσδιορισμός

FORCEINLINE int32 Atoi(const TCHAR *InString)

Παράμετροι

*const TCHAR *InString* : Αλφαριθμητικό προς μετατροπή.

Header file

CString.h

Περιγραφή

Μετατρέπει την αλφαριθμητική παράμετρο σε έναν ακέραιο αριθμό.

Προσδιορισμός

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

FORCEINLINE float Atof(const TCHAR *InString)

Παράμετροι

*const TCHAR *InString* : Αλφαριθμητικό προς μετατροπή.

Header file

CString.h

Περιγραφή

Μετατρέπει την αλφαριθμητική παράμετρο σε έναν δεκαδικό αριθμό.

Προσδιορισμός

FORCEINLINE TCHAR *Copy(TCHAR *OutDestination, const TCHAR *InSource)

Παράμετροι

*TCHAR *OutDestination* : Αλφαριθμητικό προς αποθήκευση.

*const TCHAR *InSource* : Αλφαριθμητικό προς αντιγραφή.

Header file

CString.h

Περιγραφή

Αντιγράφει το αλφαριθμητικό της παραμέτρου *InSource* στην παράμετρο *OutDestination*.

Προσδιορισμός

FORCEINLINE TCHAR *Concat(TCHAR *OutDestination, const TCHAR *InSource)

Παράμετροι

*TCHAR *OutDestination* : Αλφαριθμητικό προς επέκταση.

*const TCHAR *InSource* : Αλφαριθμητικό προς αντιγραφή.

Header file

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

CString.h

Περιγραφή

Επεκτείνει το αλφαριθμητικό της παραμέτρου *OutDestination* με την παράμετρο *Insource*. Στο τέλος εκχωρείται και ο τερματικός χαρακτήρας.

Προσδιορισμός

```
FORCEINLINE TCHAR *SubString(const TCHAR *InString,  
                             const TCHAR *InSubString)
```

Παράμετροι

*const TCHAR *InString* : Αλφαριθμητικό προς έλεγχο.

*const TCHAR *InSubString* : Αλφαριθμητικό ελέγχου.

Header file

CString.h

Περιγραφή

Ελέγχει αν το αλφαριθμητικό της παραμέτρου *InSubString* υπάρχει στην παράμετρο *InString*. Σε περίπτωση επιτυχίας επιστρέφει την διεύθυνση που βρέθηκε αλλιώς 0.

Προσδιορισμός

```
FORCEINLINE int32 Compare(const TCHAR *InFirstString,  
                          const TCHAR *InSecondString)
```

Παράμετροι

*const TCHAR *InFirstString* : Αλφαριθμητικό προς έλεγχο.

*const TCHAR *InSecondString* : Αλφαριθμητικό ελέγχου.

Header file

CString.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Ελέγχει αν το αλφαριθμητικό της παραμέτρου *InFirstString* είναι ίδιο με αυτό της παραμέτρου *InSecondString*. Σε περίπτωση επιτυχίας επιστρέφει 1 αλλιώς 0.

Προσδιορισμός

```
FORCEINLINE int32 GetLength(const TCHAR *InString)
```

Παράμετροι

*const TCHAR *InString* : Αλφαριθμητικό προς εξέταση.

Header file

CString.h

Περιγραφή

Επιστρέφει το μέγεθος σε χαρακτήρες του αλφαριθμητικού της παραμέτρου *InString*.

Προσδιορισμός

```
FORCEINLINE bool IsNumeric(const TCHAR *InString)
```

Παράμετροι

*const TCHAR *InString* : Αλφαριθμητικό προς εξέταση.

Header file

CString.h

Περιγραφή

Ελέγχει αν το αλφαριθμητικό της παραμέτρου *InString* είναι αριθμός.

Προσδιορισμός

```
FORCEINLINE FString()
```

```
FORCEINLINE FString(const FString &InString)
```

```
FORCEINLINE FString(const TCHAR *InString)
```


Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

FORCEINLINE FString(const char &InChar)

Παράμετροι

const FString &InString : Δημιουργία αλφαριθμητικού.

*const TCHAR *InString* : Δημιουργία αλφαριθμητικού.

const char &InChar : Δημιουργία χαρακτήρα αλφαριθμητικού.

Header file

String.h

Περιγραφή

Δημιουργεί ένα αλφαριθμητικό αντικείμενο *Fstring* το οποίο μπορεί να είναι κενό, ενός αλφαριθμητικού ή χαρακτήρα που εκχωρείται από την παράμετρο.

Προσδιορισμός

FORCEINLINE ~FString()

Παράμετροι

Δεν έχει.

Header file

String.h

Περιγραφή

Αποδεσμέυει από τη μνήμη το αλφαριθμητικό αντικείμενο.

Προσδιορισμός

FORCEINLINE FString &Resize(uint32 InLength)

Παράμετροι

uint32 InLength : Μέγεθος του αλφαριθμητικού.

Header file

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

String.h

Περιγραφή

Αλλάζει το μέγεθος σε χαρακτήρες του αλφαριθμητικού με βάση την παράμετρο *InLength*.

Προσδιορισμός

```
FORCEINLINE FString &SubString(uint32 Start, uint32 End)
```

Παράμετροι

uint32 Start : Αρχική θέση προς διαγραφή.

uint32 End : Τελική θέση προς διαγραφή.

Header file

String.h

Περιγραφή

Διαγράφει μια σειρά χαρακτήρων που ορίζεται από τις παραμέτρους *Start*(αρχή) και *End*(τέλος) του αλφαριθμητικού.

Προσδιορισμός

```
FORCEINLINE FString &Copy(const FString &InString)
```

Παράμετροι

const FString &InString : Αλφαριθμητικό προς αντιγραφή.

Header file

String.h

Περιγραφή

Επιστρέφει ένα αλφαριθμητικό το οποίο έχει τιμή ίση με την παράμετρο *InString*.

Προσδιορισμός

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

FORCEINLINE FString &Append(const FString &InString)

Παράμετροι

const FString &InString : Αλφαριθμητικό προς αντιγραφή.

Header file

String.h

Περιγραφή

Επεκτείνει το αλφαριθμητικό με την τιμή της παραμέτρου *InString*.

Προσδιορισμός

FORCEINLINE FString &Insert(const FString &InString, uint32 Index)

Παράμετροι

const FString &InString : Αλφαριθμητικό προς εισαγωγή.

uint32 Index : Θέση εισαγωγής αλφαριθμητικού.

Header file

String.h

Περιγραφή

Εισάγει ένα αλφαριθμητικό στο ήδη υπάρχων ίσο την τιμή της παραμέτρου *InString* στην θέση *Index*.

Προσδιορισμός

FORCEINLINE FString &Insert(const FString &InString, uint32 Index)

Παράμετροι

const FString &InString : Αλφαριθμητικό προς εισαγωγή.

uint32 Index : Θέση εισαγωγής αλφαριθμητικού.

Header file

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

String.h

Περιγραφή

Εισάγει ένα αλφαριθμητικό ίσο την τιμή της παραμέτρου *InString* στην θέση *Index*.

Προσδιορισμός

```
FORCEINLINE FString &Remove(uint32 Index, uint32 Count)
```

Παράμετροι

uint32 Index : Θέση προς διαγραφή.

uint32 Count : Μέγεθος προς διαγραφή.

Header file

String.h

Περιγραφή

Διαγράφει μια σειρά χαρακτήρων *Count*(προκαθορισμένο στο 1) στο σημείο *Index* του αλφαριθμητικού.

Προσδιορισμός

```
FORCEINLINE FString &Replace(const FString &InDestination,  
                             const FString &InSource)
```

Παράμετροι

const FString &InDestination : Αλφαριθμητικό προς αντικατάσταση.

const FString &InSource : Αλφαριθμητικό αντικατάστασης.

Header file

String.h

Περιγραφή

Αντικαθιστά το αλφαριθμητικό *InDestination* με την τιμή του *InSource*.

Προσδιορισμός

```
FORCEINLINE bool Compare(const FString &InString) const
```

Παράμετροι

const FString &InString : Αλφαριθμητικό προς σύγκριση.

Header file

String.h

Περιγραφή

Συγρίνει την τιμή του αλφαριθμητικού *InString* με αυτήν του αλφαριθμητικού αντικειμένου. Αν είναι ίσες, επιστρέφει 1 αλλιώς 0.

Προσδιορισμός

```
FORCEINLINE bool Contains(const FString &InString) const
```

Παράμετροι

const FString &InString : Αλφαριθμητικό προς σύγκριση.

Header file

String.h

Περιγραφή

Ελέγχει αν υπάρχει η τιμή του αλφαριθμητικού *InString* στο αλφαριθμητικό του αντικειμένου. Αν υπάρχει, επιστρέφει 1 αλλιώς 0.

Προσδιορισμός

```
FORCEINLINE void Empty()
```

Παράμετροι

Δεν έχει.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Header file

String.h

Περιγραφή

Μηδενίζει το αλφαριθμητικό του αντικειμένου.

Προσδιορισμός

```
FORCEINLINE int32 SearchLast(const FString &InString) const
```

Παράμετροι

const FString &InString : Αλφαριθμητικό προς αναζήτηση.

Header file

String.h

Περιγραφή

Ελέγχει αν υπάρχει το αλφαριθμητικό του αντικειμένου με την τιμή του *InString* και επιστρέφει την τελευταία θέση που βρέθηκε.

Προσδιορισμός

```
FORCEINLINE int32 Search(const FString &InString,  
                        uint32 Offset) const
```

Παράμετροι

const FString &InString : Αλφαριθμητικό προς αναζήτηση.

uint32 Offset : Θέση αναζήτησης.

Header file

String.h

Περιγραφή

Ελέγχει αν υπάρχει το αλφαριθμητικό του αντικειμένου με την τιμή του *InString* στην θέση που ορίζεται από την παράμετρο *Offset*(0 εξ'ορισμού) και επιστρέφει 1 αν βρέθηκε αλλιώς -1.

Προσδιορισμός

FORCEINLINE uint32 GetLength() **const**

Παράμετροι

Δεν έχει.

Header file

String.h

Περιγραφή

Επιστρέφει το μέγεθος σε χαρακτήρες(bytes) του αλφαριθμητικού αντικειμένου.

Προσδιορισμός

FORCEINLINE bool IsEmpty() **const**

Παράμετροι

Δεν έχει.

Header file

String.h

Περιγραφή

Ελέγχει αν το αλφαριθμητικού αντικειμένου είναι κενό.

Προσδιορισμός

FORCEINLINE bool IsNumeric() **const**

Παράμετροι

Δεν έχει.

Header file

String.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Περιγραφή

Ελέγχει αν το αλφαριθμητικό αντικειμένου είναι αριθμός.

Προσδιορισμός

```
FORCEINLINE bool IsNumeric() const
```

Παράμετροι

Δεν έχει.

Header file

String.h

Περιγραφή

Ελέγχει αν το αλφαριθμητικό αντικειμένου είναι αριθμός.

Προσδιορισμός

```
FORCEINLINE TCHAR *GetData()
```

Παράμετροι

Δεν έχει.

Header file

String.h

Περιγραφή

Επιστρέφει το αλφαριθμητικό αντικειμένου.

Προσδιορισμός

```
FORCEINLINE const TCHAR *GetData() const
```

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δεν έχει.

Header file

String.h

Περιγραφή

Επιστρέφει το αλφαριθμητικό αντικειμένου.

Προσδιορισμός

```
template <class Type>  
static FORCEINLINE void Swap(Type &A, Type &B)
```

Παράμετροι

Type &A : Μεταβλητή A τύπου *Type*.

Type &B : Μεταβλητή B τύπου *Type*.

Header file

Math.h

Περιγραφή

Ανταλλάζει τις τιμές των A,B.

Προσδιορισμός

```
template <class Type>  
static FORCEINLINE Type Clamp(Type Value, Type Min, Type Max)
```

Παράμετροι

Type Value : Δοσμένη τιμή.

Type Min : Ελάχιστη επιτρεπόμενη τιμή.

Type Max : Μέγιστη επιτρεπόμενη τιμή.

Header file

Πανεπιστήμιο Δυτικής Αττικής Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Math.h

Περιγραφή

Ελέγχει αν η τιμή Value είναι μεγαλύτερη από την Max και μικρότερη από την Min. Αν είναι μεγαλύτερη από την Max επιστρέφει Max, αν μικρότερη από την Min επιστρέφει Min αλλιώς επιστρέφει Value.

Προσδιορισμός

```
template <class Type>  
static FORCEINLINE Type CircularClamp(Type Value, Type Min, Type Max)
```

Παράμετροι

Type Value : Δοσμένη τιμή.

Type Min : Ελάχιστη επιτρεπόμενη τιμή.

Type Max : Μέγιστη επιτρεπόμενη τιμή.

Header file

Math.h

Περιγραφή

Ελέγχει αν η τιμή Value είναι μεγαλύτερη από την Max και μικρότερη από την Min. Αν είναι μεγαλύτερη από την Max επιστρέφει Min, αν μικρότερη από την Min επιστρέφει Max αλλιώς επιστρέφει Value.

Προσδιορισμός

```
FORCEINLINE FColor32()
```

Παράμετροι

Δεν έχει.

Header file

Color32.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Περιγραφή

Δημιουργεί και εκχωρεί σε ένα αντικείμενο *Color32* το μαύρο χρώμα.

Προσδιορισμός

```
FORCEINLINE FColor32(uint8 R, uint8 G, uint8 B)
```

Παράμετροι

uint8 R : Μεταβλητή για το κόκκινο χρώμα.

uint8 G : Μεταβλητή για το πράσινο χρώμα.

uint8 B : Μεταβλητή για το μπλε χρώμα.

Header file

Color32.h

Περιγραφή

Δημιουργεί και εκχωρεί σε ένα αντικείμενο *Color32* το χρώμα που προκύπτει από τις παραμέτρους *R,G,B*.

Προσδιορισμός

```
FORCEINLINE FColor32(uint8 R, uint8 G, uint8 B, uint8 A)
```

Παράμετροι

uint8 R : Μεταβλητή για το κόκκινο χρώμα.

uint8 G : Μεταβλητή για το πράσινο χρώμα.

uint8 B : Μεταβλητή για το μπλε χρώμα.

uint8 A : Μεταβλητή για το μπλε χρώμα.

Header file

Color32.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δημιουργεί και εκχωρεί σε ένα αντικείμενο *Color32* το χρώμα που προκύπτει από τις παραμέτρους *R,G,B*.

Προσδιορισμός

```
FORCEINLINE bool IsOpaqueAlpha() const
```

Παράμετροι

Δεν έχει.

Header file

Color32.h

Περιγραφή

Ελέγχει και επιστρέφει αν το χρώμα είναι αδιαφανής ή όχι.

IntPoint

Προσδιορισμός

```
FORCEINLINE FIntPoint()
```

Παράμετροι

Δεν έχει.

Header file

IntPoint.h

Περιγραφή

Constructor που δημιουργεί ένα αντικείμενο *FIntPoint* το οποίο περιέχει ακαίρες συντεταγμένες *x* και *y*. Τις αρχικοποιεί με μηδενική τιμή.

Προσδιορισμός

```
FORCEINLINE FIntPoint(int32 X, int32 Y)
```

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Παράμετροι

int32 X : Συντεταγμένη Χ.

int32 Y : Συντεταγμένη Υ.

Header file

IntPoint.h

Περιγραφή

Constructor που δημιουργεί ένα αντικείμενο `FIntPoint` το οποίο περιέχει ακαίρες συντεταγμένες *x* και *y* με τιμές *X* και *Y* από τις παραμέτρους.

Προσδιορισμός

```
FORCEINLINE FIntPoint(const FIntPoint &Other)
```

Παράμετροι

const FIntPoint &Other : Αντικείμενο `FIntPoint`.

Header file

IntPoint.h

Περιγραφή

Constructor που δημιουργεί ένα αντικείμενο `FIntPoint` το οποίο περιέχει ακαίρες συντεταγμένες *x* και *y* με τιμές που εκχωρούνται από την παράμετρο *Other*.

Προσδιορισμός

```
FORCEINLINE FIntPoint(const FVector2D &Other)
```

Παράμετροι

const FVector2D &Other : Αντικείμενο `FVector2D`.

Header file

IntPoint.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Περιγραφή

Constructor που δημιουργεί ένα αντικείμενο *FIntPoint* το οποίο περιέχει ακαίρες συντεταγμένες *x* και *y* με τιμές που εκχωρούνται από την παράμετρο *Other*.

Προσδιορισμός

```
FORCEINLINE bool InsideRect(const FIntRect &InRect) const
```

Παράμετροι

const FIntRect &InRect : Αντικείμενο *InRect* προς έλεγχο.

Header file

IntPoint.h

Περιγραφή

Ελέγχει και επιστρέφει αν οι συντεταγμένες της κλάσης *IntPoint* βρίσκονται εντός του ορθογωνίου της παραμέτρου *InRect*.

IntRect

Προσδιορισμός

```
FORCEINLINE FIntRect(int32 MinX, int32 MinY, int32 MaxX, int32 MaxY);
```

Παράμετροι

int32 MinX : Αρχικό σημείο *x* του ορθογώνιου στον οριζόντιο άξονα.

int32 MinY : Αρχικό σημείο *y* του ορθογώνιου στον κατακόρυφο άξονα.

int32 MaxX : Τελικό σημείο *x* του ορθογώνιου στον οριζόντιο άξονα.

int32 MaxY : Τελικό σημείο *y* του ορθογώνιου στον κατακόρυφο άξονα.

Header file

IntRect.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Constructor που δημιουργεί ένα ορθογώνιο με βάση τις συντεταγμένες των παραμέτρων. Το μήκος και το πλάτος προκύπτει από την αφαίρεση των τελικών και των αρχικών σημείων x και y αντίστοιχα.

Προσδιορισμός

```
FORCEINLINE FIntRect(const FIntRect &Other)
```

Παράμετροι

const FIntRect &Other : Αντικείμενο προς αποθήκευση.

Header file

IntRect.h

Περιγραφή

Constructor που δημιουργεί ένα αντικείμενο *FIntRect* στο οποίο αποθηκεύεται το περιεχόμενο του αντικειμένου *Other* (Συντεταγμένες ορθογωνίου).

Προσδιορισμός

```
FORCEINLINE FIntRect(float Value)
```

Παράμετροι

float Value : Αντικείμενο τύπου *FVector4D* προς αποθήκευση.

Header file

IntRect.h

Περιγραφή

Constructor που δημιουργεί ένα αντικείμενο *FIntRect* στο οποίο εκχωρείται το περιεχόμενο της μεταβλητής *Value* (Συντεταγμένες ορθογωνίου).

Προσδιορισμός

```
FORCEINLINE FIntRect(const FVector2D &Min, const FVector2D &Max)
```

Παράμετροι

const FVector2D &Min : Αρχικές συντεταγμένες x,y.

const FVector2D &Max : Τελικές συντεταγμένες x,y.

Header file

IntRect.h

Περιγραφή

Constructor που δημιουργεί ένα ορθογώνιο με βάση τις συντεταγμένες των παραμέτρων.

Προσδιορισμός

FORCEINLINE FIntRect(**const** FIntPoint &Min, **const** FIntPoint &Max)

Παράμετροι

const FIntPoint &Min : Αρχικές συντεταγμένες x,y.

const FIntPoint &Max : Τελικές συντεταγμένες x,y.

Header file

IntRect.h

Περιγραφή

Constructor που δημιουργεί ένα ορθογώνιο με βάση τις συντεταγμένες των παραμέτρων.

Προσδιορισμός

FORCEINLINE bool InsideRect(**const** FIntRect &InRect) **const**

Παράμετροι

const FIntRect &InRect : Αντικείμενο ορθογωνίου.

Header file

IntRect.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Περιγραφή

Ελέγχει αν το ορθογώνιο της παραμέτρου *InRect* βρίσκεται εντός του ορθογωνίου του αντικειμένου της κλάσης. Επιστρέφει 1 σε επιτυχία αλλιώς 0.

Προσδιορισμός

FORCEINLINE *FVector4D* ToVector4D() **const**

Παράμετροι

Δεν έχει.

Header file

IntRect.h

Περιγραφή

Επιστρέφει τις συντεταγμένες του ορθογωνίου σε ένα αντικείμενο τύπου *FVector4D*.

Προσδιορισμός

FORCEINLINE int32 GetOriginX() **const**

Παράμετροι

Δεν έχει.

Header file

IntRect.h

Περιγραφή

Επιστρέφει το αρχικό σημείο x που ξεκινάει το ορθογώνιο.

Προσδιορισμός

FORCEINLINE int32 GetOriginY() **const**

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δεν έχει.

Header file

IntRect.h

Περιγραφή

Επιστρέφει το αρχικό σημείο γ που ξεκινάει το ορθογώνιο.

Προσδιορισμός

```
FORCEINLINE FIntPoint GetOrigin() const
```

Παράμετροι

Δεν έχει.

Header file

IntRect.h

Περιγραφή

Επιστρέφει τα αρχικά σημεία x, γ που ξεκινάει το ορθογώνιο.

Προσδιορισμός

```
FORCEINLINE int32 GetSizeX() const;
```

Παράμετροι

Δεν έχει.

Header file

IntRect.h

Περιγραφή

Επιστρέφει το μήκος του ορθογώνιου.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

```
FORCEINLINE int32 GetSizeY() const;
```

Παράμετροι

Δεν έχει.

Header file

IntRect.h

Περιγραφή

Επιστρέφει το πλάτος του ορθογώνιου.

Προσδιορισμός

```
FORCEINLINE FIntPoint GetSize() const
```

Παράμετροι

Δεν έχει.

Header file

IntRect.h

Περιγραφή

Επιστρέφει το μήκος και το πλάτος του ορθογώνιου.

Προσδιορισμός

```
FORCEINLINE FMatrix()
```

Παράμετροι

Δεν έχει.

Header file

Matrix.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Περιγραφή

Constructor που δημιουργεί τον μοναδιαίο πίνακα.

Προσδιορισμός

```
FORCEINLINE FMatrix(float Value)
```

Παράμετροι

float Value : Τιμή που πολλαπλασιάζουμε.

Header file

Matrix.h

Περιγραφή

Constructor που πολλαπλασιάζει τον μοναδιαίο πίνακα με την παράμετρο *Value*.

Προσδιορισμός

```
FORCEINLINE FMatrix(const FMatrix &Other)
```

Παράμετροι

const FMatrix &Other : Πίνακας προς αντιγραφή.

Header file

Matrix.h

Περιγραφή

Constructor που εκχωρεί σε έναν πίνακα την τιμή της παραμέτρου(πίνακα) *Other*.

Προσδιορισμός

```
FORCEINLINE FMatrix(const FVector4D &Vector0, const FVector4D &Vector1,  
                    const FVector4D &Vector2, const FVector4D &Vector3)
```

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

const FVector4D &Vector0 : Πρώτο διάνυσμα.

const FVector4D &Vector1 : Δεύτερο διάνυσμα.

const FVector4D &Vector2 : Τρίτο διάνυσμα.

const FVector4D &Vector3 : Τέταρτο διάνυσμα.

Header file

Matrix.h

Περιγραφή

Constructor που δημιουργεί έναν πίνακα ο οποίος προκύπτει από τα διανύσματα των παραμέτρων *Vector0*, *Vector1*, *Vector2*, *Vector3*.

Προσδιορισμός

FORCEINLINE FMatrix Translate(**const** FVector3D &InTranslation)

Παράμετροι

const FVector3D &InTranslation : Πίνακας μεταβολής θέσης.

Header file

Matrix.h

Περιγραφή

Μετασχηματίζει τον πίνακα του αντικειμένου βάση της παραμέτρου *InTranslation*, με τέτοιο τρόπο ώστε να μετακινεί μια «φιγούρα» από ένα σημείο του ορθοκανονικού συστήματος σε ένα άλλο χωρίς να μεταβάλει το σχήμα, μέγεθος ή τον προσανατολισμό της.

Προσδιορισμός

FORCEINLINE FMatrix Scale(**const** FVector3D &InScale)

Παράμετροι

const FVector3D &InScale : Πίνακας μεταβολής μεγέθους.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Header file

Matrix.h

Περιγραφή

Μετασχηματίζει τον πίνακα του αντικειμένου βάση της παραμέτρου *InScale*, με τέτοιο τρόπο ώστε να μετασχηματίζει το μέγεθος μιας «φιγούρας».

Προσδιορισμός

```
FORCEINLINE FMatrix GetInversed() const
```

Παράμετροι

Δεν έχει.

Header file

Matrix.h

Περιγραφή

Αντιστρέφει τον πίνακα.

Προσδιορισμός

```
FORCEINLINE FMatrix GetTransposed() const
```

Παράμετροι

Δεν έχει.

Header file

Matrix.h

Περιγραφή

Αντιστρέφει τις σειρές με τις στήλες του πίνακα.

Vector2D

Προσδιορισμός

```
FORCEINLINE FVector2D();
```

Παράμετροι

Δεν έχει.

Header file

Matrix.h

Περιγραφή

Δημιουργεί ένα μηδενικό διάνυσμα *FVector2D* με x και y ίσο με μηδέν.

Προσδιορισμός

```
FORCEINLINE FVector2D(const FVector2D &Other)
```

Παράμετροι

Δεν έχει.

Header file

Matrix.h

Περιγραφή

Δημιουργεί ένα διάνυσμα *FVector2D* στο οποίο εκχωρείται το περιεχόμενο της παραμέτρου *Other*.

Προσδιορισμός

```
FORCEINLINE FVector2D(const FVector3D &Other)
```

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δεν έχει.

Header file

Matrix.h

Περιγραφή

Δημιουργεί ένα διάνυσμα *FVector2D* στο οποίο το x και το y του ορίζεται απο τις παραμέτρους X και Y.

Προσδιορισμός

```
FORCEINLINE FVector2D(const FVector4D &Other)
```

Παράμετροι

Δεν έχει.

Header file

Matrix.h

Περιγραφή

Δημιουργεί ένα διάνυσμα *FVector2D* στο οποίο το x και το y του ορίζεται απο τις παραμέτρους X και Y.

Προσδιορισμός

```
FORCEINLINE FVector2D(float X, float Y)
```

Παράμετροι

Δεν έχει.

Header file

Matrix.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δημιουργεί ένα διάνυσμα *FVector2D* στο οποίο το *x* και το *y* του ορίζεται από τις παραμέτρους *X* και *Y*.

5.3.2 Δομή Platform

Προσδιορισμός

```
static LRESULT CALLBACK
```

```
WindowMessageProc(HWND Window, uint32 Message, WPARAM wParam, LPARAM lParam)
```

Παράμετροι

Window : Το «χερούλι» για το παράθυρο. Γενικά μπορεί να είναι μια διεύθυνση μνήμης, μια αναφορά σε ένα αντικείμενο μιας κλάσης κλπ που μόνο τα windows ξέρουν και χρησιμοποιούν.

Message : Το μήνυμα που περνάμε στο παράθυρο.

wParam : Επιπλέον πληροφορίες για το μήνυμα.

lParam : Επιπλέον πληροφορίες για το μήνυμα.

Header file

WindowsWindow.h

Περιγραφή

Συνάρτηση των windows που επεξεργάζεται τα μηνύματα που στέλνονται στο παράθυρο. Επιστρέφει το αποτέλεσμα της επεξεργασίας μηνυμάτων και εξαρτάται από το μήνυμα που στάλθηκε.

Προσδιορισμός

```
static void InitializeWindowsClass ()
```

Παράμετροι

Δεν έχει.

Header file

WindowsWindow.h

Περιγραφή

Συνάρτηση που αρχικοποιεί τα χαρακτηριστικά της WindowClass(κλάση παραθύρων)

Η κλάση αυτή καθορίζει ένα σύνολο συμπεριφορών που ενδέχεται να έχουν ως κοινού πολλά παράθυρα. Για παράδειγμα, σε μια ομάδα κουμπιών, κάθε κουμπί έχει παρόμοια συμπεριφορά όταν ο χρήστης κάνει κλικ στο κουμπί. Φυσικά, τα κουμπιά δεν είναι απολύτως πανομοιότυπα. Κάθε κουμπί εμφανίζει τη δική του συμβολοσειρά κειμένου και έχει τις δικές του συντεταγμένες οθόνης.

Κάθε παράθυρο πρέπει να συσχετίζεται με μια κλάση παραθύρων, ακόμα κι αν το πρόγραμμά σας δημιουργεί μόνο μία παρουσία αυτής της κλάσης. Είναι σημαντικό να κατανοήσουμε ότι μια κλάση παραθύρου δεν είναι "κλάση" με την έννοια της C ++. Αντίθετα, είναι μια δομή δεδομένων που χρησιμοποιείται εσωτερικά από το λειτουργικό σύστημα. Οι κλάσεις παραθύρων καταχωρούνται στο σύστημα κατά το χρόνο εκτέλεσης.

Προσδιορισμός

```
FWindow(const FWindowInitializer &Initializer)
```

Παράμετροι

Initializer : Αντικείμενο της κλάσης όπου περιέχει όλα τα χαρακτηριστικά του παραθύρου όπως το μέγεθος τις συντεταγμένες χρώμα κ.α

Header file

WindowsWindow.h

Περιγραφή

Συνάρτηση(constructor) η οποία δημιουργεί το παράθυρο και το εμφανίζει.

Προσδιορισμός

```
~FWindow()
```

Παράμετροι

Δεν έχει.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Header file

WindowsWindow.h

Περιγραφή

Συνάρτηση(destructor) η οποία καταστρέφει το παράθυρο και διαγράφει την σχετική μνήμη που δεσμεύτηκε.

Προσδιορισμός

```
Fapplication *Get()
```

Παράμετροι

Δεν έχει.

Header file

WindowsApplication.h

Περιγραφή

Συνάρτηση που επιστρέφει το αντικείμενο της κλάσης Fapplication.

Προσδιορισμός

```
void RequestClose()
```

Παράμετροι

Δεν έχει.

Header file

WindowsApplication.h

Περιγραφή

Συνάρτηση που αληθεύει το flag για να κλείσει το παράθυρο.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

`void ProcessPendingEvents()`

Παράμετροι

Δεν έχει

Header file

WindowsApplication.h

Περιγραφή

Συνάρτηση που βλέπει, διαβάζει και επεξεργάζεται τα μηνύματα του παραθύρου.

Προσδιορισμός

`void GetApplicationResolution(int32 *OutResolutionSizeX,
int32 *OutResolutionSizeY)`

Παράμετροι

`int32 *OutResolutionSizeX` : Το μήκος του παραθύρου.

`int32 *OutResolutionSizeY` : Το πλάτος του παραθύρου.

Header file

WindowsApplication.h

Περιγραφή

Συνάρτηση που επιστρέφει τις διαστάσεις του παραθύρου.

Προσδιορισμός

`void GetCursorPos(FWindow *Window, int32 *OutCursorX,
int32 *OutCursorY)`

Παράμετροι

`FWindow *Window` : Το αντικείμενο του παραθύρου.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

int32 *OutCursorX : Το μήκος του παραθύρου.

int32 *OutCursorY : Το πλάτος του παραθύρου.

Header file

WindowsApplication.h

Περιγραφή

Συνάρτηση που επιστρέφει τις συντεταγμένες του ποντικιού στο παράθυρο.

Προσδιορισμός

```
void GetMouseState(uint16 *bOutIsClicked)
```

Παράμετροι

uint16 *bOutIsClicked : Το μήκος του παραθύρου.

Header file

WindowsApplication.h

Περιγραφή

Συνάρτηση που ελέγχει και επιστρέφει αν πατήθηκε το αριστερό κλικ του ποντικιού.

Προσδιορισμός

```
void SetTimeElapsed(double TimeElapsed)
```

Παράμετροι

double TimeElapsed : Ο χρόνος που πέρασε.

Header file

WindowsApplication.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Συνάρτηση που θέτει την παραμέτρο στην οποία αποθηκεύεται ο χρόνος που πέρασε. (Προκύπτει από τις συναρτήσεις Tick() και MilliSeconds() στην περίπτωση μας).

Προσδιορισμός

```
double GetTimeElapsed()
```

Παράμετροι

Δεν έχει.

Header file

WindowsApplication.h

Περιγραφή

Συνάρτηση που επιστρέφει τον χρόνο που πέρασε. (Προκύπτει από τις συναρτήσεις Tick() και MilliSeconds() στην περίπτωση μας).

Προσδιορισμός

```
bool IsKeyDown(uint16 KeyCode) const
```

Παράμετροι

uint16 KeyCode : Κωδικός κουμπιού από το πληκτρολόγιο.

Header file

WindowsApplication.h

Περιγραφή

Συνάρτηση που ελέγχει αν πατήθηκε ένα κουμπί από το πληκτρολόγιο. Σε περίπτωση που πατήθηκε το επιστρέφει, αλλιώς επιστρέφει 0.

Προσδιορισμός

```
bool IsKeyUp (uint16 KeyCode) const
```

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Παράμετροι

uint16 KeyCode : Κωδικός κουμπιού από το πληκτρολόγιο.

Header file

WindowsApplication.h

Περιγραφή

Συνάρτηση που ελέγχει αν πατήθηκε και εξακολουθεί να πατίεται ένα κουμπί από το πληκτρολόγιο. Επιστρέφει ένα σε αυτήν την περίπτωση, αλλιώς επιστρέφει 0.

Προσδιορισμός

```
void UpdateKeyState()
```

Παράμετροι

Δεν έχει.

Header file

WindowsApplication.h

Περιγραφή

Αρχικοποιεί την κατάσταση των πλήκτρων σε 0 (Δηλαδή σε μη πατημένη κατάσταση).

Προσδιορισμός

```
FPlatformFileIni(const FString &Filename)
```

Παράμετροι

const FString &Filename : Η διεύθυνση του αρχείου.

Header file

WindowsPlatformFileIni.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Αποθηκεύει τη διεύθυνση του αρχείου σε μια μεταβλητή `private`.

Προσδιορισμός

```
bool Exists(const FString &InField) const
```

Παράμετροι

const FString &InField : Αλφαριθμητικό προς αναζήτηση.

Header file

WindowsPlatformFileIni.h

Περιγραφή

Η συνάρτηση ελέγχει αν το όνομα υπάρχει στο αρχείο. Αν υπάρχει επιστρέφει 1 αλλιώς 0.

Προσδιορισμός

```
void BeginReadback(const FString &InField)
```

Παράμετροι

const FString &InField : Αλφαριθμητικό προς αναζήτηση.

Header file

WindowsPlatformFileIni.h

Περιγραφή

Η συνάρτηση αποθηκεύει το αλφαριθμητικό που αναζητούμε στο αρχείο και θα χρησιμοποιηθεί ως προς διάβασμα.

Προσδιορισμός

```
int32 ReadInteger(const FString &InKey)
```


Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Παράμετροι

const FString &InKey : Κλειδί προς αναζήτηση.

Header file

WindowsPlatformFileIni.h

Περιγραφή

Η συνάρτηση με βάση το αλφαριθμητικό που βρέθηκε στην αναζήτηση, αναζητεί εκ νέου ένα κλειδί που υπάρχει σε αυτό στο αρχείο και αποθηκεύει την τιμή που βρέθηκε.

Προσδιορισμός

void EndReadback()

Παράμετροι

Δεν έχει.

Header file

WindowsPlatformFileIni.h

Περιγραφή

Η συνάρτηση μηδενίζει το αλφαριθμητικό αναζήτησης.

Προσδιορισμός

void BeginWriteStream(*const* *FString* &InField)

Παράμετροι

const FString &InField: Αλφαριθμητικό προς αναζήτηση.

Header file

WindowsPlatformFileIni.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Η συνάρτηση αποθηκεύει το αλφαριθμητικό που αναζητούμε στο αρχείο και θα χρησιμοποιηθεί ως προς επεξεργασία.

Προσδιορισμός

```
void WriteInteger(const FString &InKey, int32 Integer)
```

Παράμετροι

const FString &InKey : Κλειδί προς αναζήτηση.

int32 Integer : Τιμή προς επεξεργασία.

Header file

WindowsPlatformFileIni.h

Περιγραφή

Η συνάρτηση με βάση το αλφαριθμητικό που βρέθηκε στην αναζήτηση, αναζητεί εκ νέου ένα κλειδί που υπάρχει σε αυτό στο αρχείο, τροποποιεί την τιμή με βάση την παράμετρο και την αποθηκεύει στο αρχείο.

Προσδιορισμός

```
void EndWriteStream()
```

Παράμετροι

Δεν έχει.

Header file

WindowsPlatformFileIni.h

Περιγραφή

Η συνάρτηση μηδενίζει το αλφαριθμητικό αναζήτησης.

5.3.3 Δομή Opengl

Προσδιορισμός

```
static void CreateOpenGLContextWindow(FRenderContext *InOutContext,  
HDC DeviceContext)
```

Παράμετροι

*FRenderContext *InOutContext* : Context της Opengl.

HDC DeviceContext : Context των windows.

Header file

OpenGLWindows.h

Περιγραφή

Δημιουργεί ένα Opengl context με βάση το device context του window έτσι ώστε να μπορούμε να καλέσουμε Opengl commands.

Προσδιορισμός

```
void ReleaseRenderContext(FRenderContext *InOutContext)
```

Παράμετροι

*FRenderContext *InOutContext* : Context της Opengl.

Header file

OpenGLWindows.h

Περιγραφή

Διαγράφουμε το context και τη σχετική μνήμη που δεσμεύει.

Προσδιορισμός

```
void MakeRenderContextCurrent(FRenderContext *InOutContext))
```

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

*FRenderContext *InOutContext* : Context της OpenGL.

Header file

OpenGLWindows.h

Περιγραφή

Καθορίζουμε το context το οποίο θα χρησιμοποιήσουμε κατά την διάρκεια της εργασίας.

Προσδιορισμός

```
void InitPlatformOpenGL()
```

Παράμετροι

Δεν έχει.

Header file

OpenGLWindows.h

Περιγραφή

Φορτώνει τις συναρτήσεις της OpenGL από τα dll.

Προσδιορισμός

```
FORCEINLINE static void Present(FWindow *Window)
```

Παράμετροι

*FWindow *Window* : Ενεργό παράθυρο.

Header file

Opengl.h

Περιγραφή

Συνάρτηση που δίνει τη δυνατότητα να εμφανίζει στην οθόνη οτιδήποτε θέλουμε να εμφανίσουμε.

Προσδιορισμός

```
FORCEINLINE static void Clear(const FColor32 &Color)
```

Παράμετροι

const FColor32 &Color : Χρώμα buffer.

Header file

Opengl.h

Περιγραφή

Αντικαθιστά τον backbuffer με το χρώμα της παραμέτρου.

Προσδιορισμός

```
FORCEINLINE static void Draw(uint32 Count)
```

Παράμετροι

uint32 Count : Αριθμός πλευρών.

Header file

Opengl.h

Περιγραφή

DrawCall που εμφανίζει στην οθόνη το αντικείμενο που δώσαμε με βάση τις πλευρές(vertices) της παραμέτρου.

Προσδιορισμός

```
FORCEINLINE static void DrawIndexed(uint32 Count, uint32 FirstIndex,  
                                     uint32 BaseVertex)
```

Παράμετροι

uint32 Count : Αριθμός πλευρών.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

uint32 FirstIndex : Διεύθυνση πλευρών.

uint32 BaseVertex : Σταθερά.

Header file

OpenGL.h

Περιγραφή

DrawCall που εμφανίζει στην οθόνη το αντικείμενο που δώσαμε με βάση τις πλευρές(indices).

Προσδιορισμός

`FVertexBuffer(int64 InSize, const void *InitialData)`

Παράμετροι

int64 InSize : Το μέγεθος του buffer.

*const void *InitialData* : Τα δεδομένα που περιέχει ο buffer.

Header file

VertexBuffer.h

Περιγραφή

Ο constructor δημιουργεί ένα αντικείμενο τύπου vertex buffer που περιέχει δεδομένα όπως συντεταγμένες πλευρών,εικόνας,καθέτων ενός αντικειμένου τις οποίες διαχειρίζεται η κάρτα γραφικών.

Προσδιορισμός

`~FVertexBuffer()`

Παράμετροι

Δεν έχει.

Header file

VertexBuffer.h

Περιγραφή

Ο destructor αποδεσμεύει την μνήμη μου χρησιμοποιεί το αντικείμενο του vertex buffer(vertex buffer object-vbo).

Προσδιορισμός

```
FIndexBuffer(int64 InSize, const void *InitialData)
```

Παράμετροι

int64 InSize : Το μέγεθος του buffer.

*const void *InitialData* : Τα δεδομένα που περιέχει ο buffer.

Header file

VertexBuffer.h

Περιγραφή

Ο constructor δημιουργεί ένα αντικείμενο τύπου Index buffer που είναι ίδιος με τον Vertex Buffer με την διαφορά ότι είναι πιο κατανοητός με λιγότερο overhead(λιγότερες πλευρές π.χ).

Προσδιορισμός

```
~FIndexBuffer()
```

Παράμετροι

Δεν έχει.

Header file

VertexBuffer.h

Περιγραφή

Ο destructor αποδεσμεύει την μνήμη μου χρησιμοποιεί το αντικείμενο του Index buffer(Index buffer object-ibo).

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

FORCEINLINE uint32 GetResourceId() **const**

Παράμετροι

Δεν έχει.

Header file

VertexBuffer.h

IndexBuffer.h

Περιγραφή

Η συνάρτηση επιστρέφει το αντικείμενο του buffer.

Προσδιορισμός

void Bind()

Παράμετροι

Δεν έχει.

Header file

VertexBuffer.h

IndexBuffer.h

Περιγραφή

Η συνάρτηση στην περίπτωση μας λέει στην OpenGL ότι το αντικείμενο που δημιουργήσουμε θα χρησιμοποιηθεί ως GL_ARRAY_BUFFER ή GL_ELEMENT_ARRAY_BUFFER δηλαδή ως αποθήκευση δεδομένων τύπου Vertex ή Index αντίστοιχα.

Προσδιορισμός

FORCEINLINE FVertexAttribute()

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δεν έχει.

Header file

VertexDeclaration.h

Περιγραφή

Default constructor.

Προσδιορισμός

```
FORCEINLINE FVertexAttribute(int32 AttributeIndex,  
                              int32 AttributeCount, int32 Offset,  
                              int32 Stride, int32 bIsFloat,  
                              int32 bIsNormalized)
```

Παράμετροι

int32 AttributeIndex : Δείκτης που δείχνει το χαρακτηριστικό που θέλουμε να επεξεργαστούμε(0 για πλευρά,1 για εικόνα 2 για χρώμα).

int32 AttributeCount : Το μέγεθος του χαρακτηριστικού.

int32 Offset : Η διεύθυνση αρχής των χαρακτηριστικών στον buffer.

int32 Stride : Η απόσταση που έχει το κάθε χαρακτηριστικό στον buffer.

int32 bIsFloat : Δείχνει αν τα δεδομένα είναι τύπου GL_FLOAT ή
GL_UNSIGNED_BYTE.

int32 bIsNormalized : Δείχνει αν τα δεδομένα που περιέχει ο buffer βρίσκονται στο
εύρος τιμών -1.0 μέχρι 1.0.

Header file

VertexDeclaration.h

Περιγραφή

Ο constructor αρχικοποιεί τα δεδομένα.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

```
void AddAttribute(const FVertexAttribute &Attribute);
```

Παράμετροι

const FVertexAttribute &Attribute : Δεδομένα ενός χαρακτηριστικού.

Header file

VertexDeclaration.h

Περιγραφή

Αποθηκεύει τα δεδομένα ενός χαρακτηριστικού σε έναν πίνακα.

Προσδιορισμός

```
void AddAttributes(const TArray<FVertexAttribute> &InAttributes)
```

Παράμετροι

const FVertexAttribute &Attribute : Δεδομένα των χαρακτηριστικών.

Header file

VertexDeclaration.h

Περιγραφή

Αποθηκεύει τα δεδομένα όλων των χαρακτηριστικών σε έναν πίνακα.

Προσδιορισμός

```
void BindAttributes() const
```

Παράμετροι

Δεν έχει.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Header file

VertexDeclaration.h

Περιγραφή

Μεταφέρει τα δεδομένα στην κάρτα γραφικών.

Προσδιορισμός

FVertexShader(**const** FString &InSourceCode)

Παράμετροι

const FString &InSourceCode : Ο πηγαίος κώδικας του Vertex Shader.

Header file

VertexShader.h

Περιγραφή

Η συνάρτηση δημιουργεί τον Vertex Shader. Ελέγχει και εμφανίζει σχετικό μήνυμα λάθους αν δεν τα καταφέρει.

Οι shaders είναι μικροπρογράμματα που αναπτύσσουμε και μένουν στην κάρτα γραφικών. Ο vertex shader μπορεί να χειριστεί ιδιότητες όπως συντεταγμένες θέσης, χρώματος και υφής.

Προσδιορισμός

~FVertexShader()

Παράμετροι

Δεν έχει.

Header file

VertexShader.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Αποδεσμέυει τον shader από την μνήμη.

Προσδιορισμός

`FPixelShader(const FString &InSourceCode)`

Παράμετροι

`const FString &InSourceCode` : Ο πηγαίος κώδικας του Pixel Shader.

Header file

PixelShader.h

Περιγραφή

Η συνάρτηση δημιουργεί τον pixel shader. Ελέγχει και εμφανίζει σχετικό μήνυμα λάθους αν δεν τα καταφέρει.

Ο pixel shader είναι υπεύθυνος για τον χρωματισμό του κάθε pixel.

Προσδιορισμός

`~FPixelShader()`

Παράμετροι

Δεν έχει.

Header file

PixelShader.h

Περιγραφή

Αποδεσμέυει τον shader από την μνήμη.

Προσδιορισμός

`FPipelineState(const FPipelineStateInitializer &Initializer)`

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

`const FPipelineStateInitializer &Initializer` : Το αντικείμενο στα
οποιά είναι αποθηκευμένα ο vertex και ο pixel shader.

Header file

PipelineState.h

Περιγραφή

Η συνάρτηση δημιουργεί το τελικό πρόγραμμα ενώνοντας τους δυο shaders.

Προσδιορισμός

`~FPixelShader()`

Παράμετροι

Δεν έχει.

Header file

PipelineState.h

Περιγραφή

Αποδεσμεύει το πρόγραμμα από την μνήμη.

Προσδιορισμός

`void Bind()`

Παράμετροι

Δεν έχει.

Header file

PipelineState.h

Περιγραφή

Εγκαθιστά ένα αντικείμενο προγράμματος ως μέρος της τρέχουσας κατάστασης
απόδοσης(Rendering state)

Προσδιορισμός

```
void SetUniformMatrix(const TCHAR *UniformName, FMatrix &Matrix)
```

Παράμετροι

*const TCHAR *UniformName* : Το όνομα της uniform μεταβλητής στον shader.

FMatrix &Matrix : Ο πίνακας δεδομένων.

Header file

PipelineState.h

Περιγραφή

Μεταφέρει τα δεδομένα μέσω πίνακα από τον επεξεργαστή στην gpu.

Προσδιορισμός

```
void SetUniformMatrix(const TCHAR *UniformName, FMatrix &Matrix)
```

Παράμετροι

*const TCHAR *UniformName* : Το όνομα της uniform μεταβλητής στον shader.

FMatrix &Matrix : Ο πίνακας δεδομένων.

Header file

PipelineState.h

Περιγραφή

Μεταφέρει τα δεδομένα μέσω πίνακα από τον επεξεργαστή στην gpu.

Προσδιορισμός

```
static int32 GetTextureFormat(EImageFormat Format, int32 *OutInternalFormat)
```

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

EImageFormat Format : Η μορφή της εικόνας (πχ RGB).

*int32 *OutInternalFormat* : Αποθηκεύει την μορφή.

Header file

Texture2D.h

Περιγραφή

Επιστρέφει και αποθηκεύει την μορφή της εικόνας .

Προσδιορισμός

`FTexture2D(int32 InSizeX, int32 InSizeY, EImageFormat InFormat)`

Παράμετροι

int32 InSizeX : Μήκος της εικόνας.

int32 InSizeY : Πλάτος της εικόνας.

EImageFormat InFormat : Μορφή της εικόνας.

Header file

Texture2D.h

Περιγραφή

Δημιουργεί ένα αντικείμενο texture χωρίς να του παρέχουμε δεδομένα πέρα από τα χαρακτηριστικά που θα έχει.

Προσδιορισμός

`~FTexture2D()`

Παράμετροι

Δεν έχει.

Header file

Texture2D.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Περιγραφή

Διαγράφει το αντικείμενο texture και τη σχετική μνήμη που δεσμεύτηκε.

Προσδιορισμός

```
void UpdateTexture(const void *InData, int32 InOffsetX,  
                  int32 InOffsetY, int32 InSizeX,  
                  int32 InSizeY)
```

Παράμετροι

<i>const void *InData</i>	: Δεδομένα της εικόνας.
<i>int32 InOffsetX</i>	: Η θέση στον οριζόντιο άξονα που ξεκινάει η εικόνα.
<i>int32 InOffsetY</i>	: Η θέση στον κατακόρυφο άξονα που ξεκινάει η εικόνα.
<i>int32 InSizeX</i>	: Μήκος της εικόνας.
<i>int32 InSizeY</i>	: Πλάτος της εικόνας.

Header file

Texture2D.h

Περιγραφή

Ενημερώνει το αντικείμενο texture που δημιουργήσαμε με τις παραμέτρους.

Προσδιορισμός

```
void BindTexture() const
```

Παράμετροι

Δεν έχει.

Header file

Texture2D.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Περιγραφή

Διαγράφει το αντικείμενο texture και τη σχετική μνήμη που δεσμεύτηκε.

5.3.4 Δομή Memory

AllocatorInterface

Προσδιορισμός

```
FORCEINLINE FAllocatorInterface(void *Memory, uint64 MemorySize)
```

Παράμετροι

*void *Memory* : Διεύθυνση μνήμης.

uint64 MemorySize : Ποσότητα μνήμης.

Header file

AllocatorInterface.h

Περιγραφή

Αρχικοποιεί τα Memory Headers για όλα τα allocation buckets.

Προσδιορισμός

```
FORCEINLINE ~FAllocatorInterface()
```

Παράμετροι

Δεν έχει.

Header file

AllocatorInterface.h

Περιγραφή

Διαγράφει το αντικείμενο και τη σχετική μνήμη που δεσμεύει.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

```
virtual void *Allocate(uint64 Size, uint8 Alignment)
```

Παράμετροι

uint64 Size : Μέγεθος μνήμης.

uint8 Alignment : Ευθυγραμμία μνήμης.

Header file

AllocatorInterface.h

Περιγραφή

Επιστρέφει μνήμη με βάση την παράμετρο εάν υπάρχει διαθέσιμη στα buckets. Η δεύτερη παράμετρος δείχνει πόσο απέχουν τα buckets μεταξύ τους.

Προσδιορισμός

```
virtual void DeAllocate(void *Address)
```

Παράμετροι

*void *Address* : Διεύθυνση μνήμης.

Header file

AllocatorInterface.h

Περιγραφή

Αποδεσμεύει από την μνήμη το αντικείμενο του Allocator.

HeapAllocator

Προσδιορισμός

```
FORCEINLINE FHeapAllocator(void *Memory, uint64 MemorySize)
```

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

*void *Memory* : Διεύθυνση μνήμης.

uint64 MemorySize : Ποσότητα μνήμης.

Header file

HeapAllocator.h

Περιγραφή

Αρχικοποιεί τον allocator με βάση τις παραμέτρους.

Προσδιορισμός

FORCEINLINE ~FHeapAllocator()

Παράμετροι

Δεν έχει.

Header file

HeapAllocator.h

Περιγραφή

Διαγράφει την σχετική μνήμη.

Προσδιορισμός

FORCEINLINE void *Allocate(uint64 Size, uint8 Alignment)

Παράμετροι

uint64 Size : Μέγεθος μνήμης.

uint8 Alignment : Ευθυγραμμία μνήμης.

Header file

HeapAllocator.h

Περιγραφή

Κάνει allocate ακριβώς με τον ίδιο τρόπο που δουλεύει η malloc με την First Fit μεθολογία.

Προσδιορισμός

FORCEINLINE void DeAllocate(void *Address)

Παράμετροι

*void *Address* : Διεύθυνση μνήμης.

Header file

HeapAllocator.h

Περιγραφή

Αποδεσμεύει από την μνήμη το αντικείμενο του Allocator.

StackAllocator

Προσδιορισμός

FORCEINLINE FStackAllocator(void *Memory, uint64 MemorySize)

Παράμετροι

*void *Memory* : Διεύθυνση μνήμης.

uint64 MemorySize : Ποσότητα μνήμης.

Header file

StackAllocator.h

Περιγραφή

Αρχικοποιεί τον allocator με βάση τις παραμέτρους.

Προσδιορισμός

FORCEINLINE ~FStackAllocator()

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δεν έχει.

Header file

StackAllocator.h

Περιγραφή

Διαγράφει την σχετική μνήμη.

Προσδιορισμός

```
FORCEINLINE void *Allocate(uint64 Size, uint8 Alignment)
```

Παράμετροι

uint64 Size : Μέγεθος μνήμης.

uint8 Alignment : Ευθυγραμμία μνήμης.

Header file

StackAllocator.h

Περιγραφή

Κάνει allocate ακριβώς με τον ίδιο τρόπο που δουλεύει το stack.

Προσδιορισμός

```
FORCEINLINE void DeAllocate(void *Address)
```

Παράμετροι

*void *Address* : Διεύθυνση μνήμης.

Header file

StackAllocator.h

Περιγραφή

Αποδεσμεύει από την μνήμη το αντικείμενο του Allocator.

5.3.5 Δομή GameFramework

GameCharCodeMap

Προσδιορισμός

```
void SetGameLanguage(EGameLanguage InGameLanguage)
```

Παράμετροι

EGameLanguage InGameLanguage : Γλώσσα που θα χρησιμοποιηθεί.

Header file

GameCharCodeMap.h

Περιγραφή

Συνάρτηση που θέτει την γλώσσα που θα φορτωθεί και θα χρησιμοποιηθεί. Ο πίνακας αυτός περιέχει το αλφάβητο της γλώσσας καθώς και αριθμούς.

Προσδιορισμός

```
FGameCharCodeMap *Get()
```

Παράμετροι

Δεν έχει.

Header file

GameCharCodeMap.h

Περιγραφή

Επιστρέφει το αντικείμενο του πίνακα χαρακτήρων της γλώσσας που έχει φορτωθεί.

Προσδιορισμός

```
bool IsCharCodePresent(const FCharCodeMap &CharCodeMap)
```

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Παράμετροι

const FCharCodeMap &CharCodeMap : Πίνακας χαρακτήρων.

Header file

GameCharCodeMap.h

Περιγραφή

Ελέγχει αν ο χαρακτήρας υπάρχει στον πίνακα και τον επιστρέφει. Στην αντίθετη περίπτωση επιστρέφει 0.

Προσδιορισμός

TCHAR GetMappedCharCode(**const** FCharCodeMap &CharCodeMap)

Παράμετροι

const FCharCodeMap &CharCodeMap : Πίνακας χαρακτήρων.

Header file

GameCharCodeMap.h

Περιγραφή

Επιστρέφει έναν χαρακτήρα σε μορφή "greeklish".

GameMainMenu

Προσδιορισμός

void OnRelease()

Παράμετροι

Δεν έχει.

Header file

GameMainMenu.h

Περιγραφή

Σχετίζεται με το πάτημα των παρακάτω κουμπιών.

- **Κουμπί FPlayButton** : Είναι το κουμπί Έναρξη το οποίο μεταβιβάζει το χρήστη στο μενού Play.
- **Κουμπί FRecordsButton** : Είναι το κουμπί Ρεκόρ το οποίο μεταβιβάζει το χρήστη στο μενού Records.
- **Κουμπί FQuitButton** : Είναι το κουμπί Έξοδος το οποίο κλείνει την εφαρμογή.

Προσδιορισμός

FMainMenu()

Παράμετροι

Δεν έχει.

Header file

GameMainMenu.h

Περιγραφή

Δημιουργεί το αρχικό μενού. Αποτελείται από τρία κουμπιά Έναρξη, Ρεκόρ, Έξοδος.

Με το πάτημα του κουμπιού Έναρξη, ο χρήστης μεταβαίνει στο μενού Play.

Με το πάτημα του κουμπιού Ρεκόρ, ο χρήστης μεταβαίνει στο μενού Records.

Με το πάτημα του κουμπιού Έξοδος, ο χρήστης κλείνει την εφαρμογή.

Προσδιορισμός

Render(*FCanvasRenderer* *InCanvasRenderer)

Παράμετροι

FCanvasRenderer *InCanvasRenderer : Αντικείμενο απεικόνισης κανβάς.

Header file

GameMainMenu.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Απεικονίζει το αρχικό μενού με γραφικά.

GamePlayMenu

Προσδιορισμός

FORCEINLINE FCategoryGameWord()

FORCEINLINE FCategoryGameWord(EPlayCategory Category,
const FString &Hint,
const FString &GameWord)

Παράμετροι

EPlayCategory Category : Κατηγορία παιχνιδιού.

const FString &Hint : Βοήθεια παιχνιδιού.

const FString &GameWord : Η λέξη του παιχνιδιού.

Header file

GamePlayMenu.h

Περιγραφή

Στην πρώτη περίπτωση δημιουργεί ένα μηδενικό αντικείμενο τύπου *FcategoryGameWord*, ενώ στην δεύτερη επιπλέον εκχωρεί την κατηγορία την βοήθεια και την λέξη του παιχνιδιού.

Προσδιορισμός

static void GetGameWordArrayByCategory(TArray<FCategoryGameWord>&OutArray,
int32 Category)

Παράμετροι

TArray<FCategoryGameWord> &OutArray : Πίνακας επιθυμητής κατηγορίας
παιχνιδιού .

int32 Category : Κατηγορία παιχνιδιού.

Header file

GameMainMenu.h

Περιγραφή

Στατική συνάρτηση που εκχωρεί σε έναν πίνακα(πρώτη παράμετρος *OutArray*) τις λέξεις και τις αντίστοιχες βοήθειες του παιχνιδιού μιας ορισμένης κατηγορίας(παράμετρος *Category*). Η κατηγορία προκύπτει από ένα *enum EplayCategory*.

Προσδιορισμός

FEnumSelectionPreviousButton(*int32* *InValue, *int32* InMin, *int32* InMax)

Παράμετροι

int32 *InValue : Μεταβλητή προς επεξεργασία. Στην προκειμένη περίπτωση μπορεί να είναι φύλο, κατηγορία παιχνιδιού ή δυσκολία παιχνιδιού.

int32 InMin : Ελάχιστη τιμή.

int32 InMax : Μέγιστη τιμή.

Header file

GamePlayMenu.h

Περιγραφή

Δημιουργεί ένα κουμπί στο οποίο εκχωρείται η ελάχιστη και η μέγιστη τιμή μιας μεταβλητής με τέτοιο τρόπο ώστε όταν πατιέται, να ελέγχεται η τωρινή τιμή με την ελάχιστη. Σε περίπτωση που είναι μικρότερη επιστρέφει την μέγιστη τιμή αλλιώς αφαιρείται κατά 1.

Προσδιορισμός

FEnumSelectionNextButton(*int32* *InValue, *int32* InMin, *int32* InMax)

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

`int32 *InValue` : Μεταβλητή προς επεξεργασία. Στην προκειμένη περίπτωση μπορεί να είναι φύλο, κατηγορία παιχνιδιού ή δυσκολία παιχνιδιού.

`int32 InMin` : Ελάχιστη τιμή.

`int32 InMax` : Μέγιστη τιμή.

Header file

GamePlayMenu.h

Περιγραφή

Δημιουργεί ένα κουμπί στο οποίο εκγχωρείται η ελάχιστη και η μέγιστη τιμή μιας μεταβλητής με τέτοιο τρόπο ώστε όταν πατιέται, να ελέγχεται η τωρινή τιμή με την μέγιστη. Σε περίπτωση που είναι μεγαλύτερη επιστρέφει την ελάχιστη τιμή αλλιώς αυξάνεται κατά 1.

Προσδιορισμός

```
bool IsValid() const
```

Παράμετροι

Δεν έχει.

Header file

GamePlayMenu.h

Περιγραφή

Ελέγχει αν το όνομα που πληκτρολογήσε ο χρήστης είναι μεγαλύτερο των τριων χαρακτήρων και ο πρώτος χαρακτήρας δεν είναι νούμερο. Σε αυτή τη περίπτωση επιστρέφει 1, αλλιώς 0.

Προσδιορισμός

```
FString &GetPlayerName() const
```

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Παράμετροι

Δεν έχει.

Header file

GamePlayMenu.h

Περιγραφή

Επιστρέφει το όνομα που πληκτρολόγησε ο χρήστης.

Προσδιορισμός

```
void ReInitialize()
```

Παράμετροι

Δεν έχει.

Header file

GamePlayMenu.h

Περιγραφή

Αρχικοποιεί το φύλο, την κατηγορία και την δυσκολία του παιχνιδιού στις προκαθορισμένες τιμές και το αν βρίσκεται ο χρήστης στο μενού Play.

Προσδιορισμός

```
void OnRelease()
```

Παράμετροι

Δεν έχει.

Header file

GamePlayMenu.h

Περιγραφή

Σχετίζεται με το πάτημα των παρακάτω κουμπιών.

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

- **Κουμπί Home** : Επιστρέφει τον χρήστη στο αρχικό μενού.
- **Κουμπί FEnumSelectionNextButton**: Πρόκειται για το κουμπί που εμφανίζεται ως το σύμβολο «μεγαλύτερο» και λειτουργεί με βάση τον constructor που ορίστηκε. Στην προκειμένη περίπτωση εμφανίζεται στο φύλο, την κατηγορία και την δυσκολία του παιχνιδιού.
- **Κουμπί FenumSelectionPreviousButton** : Πρόκειται για το κουμπί που εμφανίζεται ως το σύμβολο «μικρότερο» και λειτουργεί με βάση τον constructor που ορίστηκε. Στην προκειμένη περίπτωση εμφανίζεται στο φύλο, την κατηγορία και την δυσκολία του παιχνιδιού.
- **FplayMenuTickButton** : Είναι το κουμπί «tick» το οποίο ελέγχει αν το όνομα είναι ορθό και με βάση τα δεδομένα που δίνει ο χρήστης (φύλο, κατηγορία, δυσκολία) μεταβιβάζει τον χρήστη στο μενού του παιχνιδιού.

Προσδιορισμός

`FplayMenu()`

Παράμετροι

Δεν έχει.

Header file

GamePlayMenu.h

Περιγραφή

Δημιουργεί το μενού Έναρξη. Περιέχει όλα τα κουμπιά και την λογική πίσω από αυτά, καθώς και τα γραφικά και τη μορφή κειμένου που περιέχουν. (Προσοχή! Η απεικόνιση γίνεται μέσω της εντολής *Render* που ορίζεται παρακάτω).

Προσδιορισμός

`void Render()`

Παράμετροι

Δεν έχει.

Header file

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

GamePlayMenu.h

Περιγραφή

Διακρίνεται σε

- Render που ανήκει στην κλάση **FnameButton** όπου απεικονίζει το όνομα και την πληκτρολόγηση του με γραφικά.
- Render που ανήκει στην κλάση **FplayMenu** όπου απεικονίζει με γραφικά το μενού Έναρξη του παιχνιδιού.

GameMenu

Προσδιορισμός

```
void InitializeState()
```

Παράμετροι

Δεν έχει.

Header file

GameMenu.h

Περιγραφή

Αρχικοποιεί τον γύρο, τα γράμματα που πατήθηκαν, και τα γράμματα που περιέχει η λέξη στις προκαθορισμένες τιμές.(ένα,μηδέν και μηδέν αντίστοιχα).

Προσδιορισμός

```
void UpdateState(const TCHAR &InLetter)
```

Παράμετροι

Δεν έχει.

Header file

GameMenu.h

Περιγραφή

Σχετίζεται με την λογική του παιχνιδιού. Αρχικά ελέγχει αν το γράμμα που πατήθηκε υπάρχει στην λέξη. Εάν υπάρχει τότε εμφανίζεται στην οθόνη και απενεργοποιείται ώστε να μη μπορεί να ξαναπατηθεί. Επιπλέον ο χρήστης κερδίζει κάποιους πόντους για κάθε σωστή απάντηση οι οποίοι αυξάνονται μέχρι να κάνει λάθος. Σε περίπτωση λάθους αφαιρούνται πόντοι. Τέλος μετά την εύρεση της λέξης ή αν ο χρήστης ξεμείνει από καρδίες καταγράφεται το ρεκόρ που έκανε σε ένα αρχείο ini.

Προσδιορισμός

`FGameMenu()`

Παράμετροι

Δεν έχει.

Header file

GameMenu.h

Περιγραφή

Δημιουργεί το μενού του παιχνιδιού. Περιέχει όλα τα κουμπιά και την λογική πίσω από αυτά, καθώς και τα γραφικά και τη μορφή κειμένου που περιέχουν. (Προσοχή! Η απεικόνιση γίνεται μέσω της εντολής *Render* που ορίζεται παρακάτω).

Προσδιορισμός

`void OnRelease()`

Παράμετροι

Δεν έχει.

Header file

GameMenu.h

Περιγραφή

Σχετίζεται με το πάτημα των παρακάτω κουμπιών.

Πανεπιστήμιο Δυτικής Αττικής

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

- **Κουμπί Home** : Επιστρέφει τον χρήστη στο αρχικό μενού.
- **Κουμπί FGameMenuLetterButton** : Πρόκειται για τα κουμπιά της αλφαβήτου. Με το πάτημα τους ανάλογα με το αν υπάρχει ή όχι εμφανίζεται το σχετικό χρώμα (κόκκινο για λάθος, πράσινο για σωστό) και απενεργοποιείται με τέτοιο τρόπο ώστε ο χρήστης να μη μπορεί να το ξαναπατήσει.
- **Κουμπί FGameMenuHintButton** : Πρόκειται για το κουμπί που εμφανίζεται ως μια λάμπα το οποίο δίνει μια βοήθεια στον χρήστη εμφανίζοντας ένα σχετικό κείμενο που αφορά την εύρεση της λέξης.

Προσδιορισμός

`void Render()`

Παράμετροι

Δεν έχει.

Header file

GameMenu.h

Περιγραφή

Απεικονίζει με γραφικά το μενού του παιχνιδιού.

GameState

Προσδιορισμός

`FGameState()`

Παράμετροι

Δεν έχει.

Header file

GameState.h

Περιγραφή

Αρχικοποιεί τους πόντους και το σερί από σωστές απαντήσεις σε 0.

Προσδιορισμός

`void ReInitialize()`

Παράμετροι

Δεν έχει.

Header file

GameState.h

Περιγραφή

Αρχικοποιεί το όνομα, τους πόντους και το σερί από σωστές απαντήσεις του παίχτη σε 0.

GameWinMenu

Προσδιορισμός

`FWinMenu()`

Παράμετροι

Δεν έχει.

Header file

GameWinMenu.h

Περιγραφή

Δημιουργεί το μενού νίκης του παιχνιδιού. Περιέχει όλα τα κουμπιά και την λογική πίσω από αυτά, καθώς και τα γραφικά και τη μορφή κειμένου που περιέχουν. (Προσοχή! Η απεικόνιση γίνεται μέσω της εντολής *Render* που ορίζεται παρακάτω).

Προσδιορισμός

`void OnRelease()`

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Παράμετροι

Δεν έχει.

Header file

GameWinMenu.h

Περιγραφή

Σχετίζεται με το πάτημα των παρακάτω κουμπιών.

- **Κουμπί Home** : Επιστρέφει τον χρήστη στο αρχικό μενού.
- **Κουμπί FContinueButton** : Πρόκειται για το κουμπί συνέχεια το οποίο όταν πατηθεί ο χρήστης επιστρέφει στο μενού της έναρξης του παιχνιδιού, όπου μπορεί να αλλάξει την δυσκολία του παιχνιδιού αλλά όχι την κατηγορία.

Προσδιορισμός

`void` Render()

Παράμετροι

Δεν έχει.

Header file

GameWinMenu.h

Περιγραφή

Απεικονίζει με γραφικά το μενού της νίκης του παιχνιδιού.

GameLoseMenu

Προσδιορισμός

`FLoseMenu`()

Παράμετροι

Δεν έχει.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Header file

GameLoseMenu.h

Περιγραφή

Δημιουργεί το μενού ήττας του παιχνιδιού. Περιέχει όλα τα κουμπιά και την λογική πίσω από αυτά, καθώς και τα γραφικά και τη μορφή κειμένου που περιέχουν. (Προσοχή! Η απεικόνιση γίνεται μέσω της εντολής *Render* που ορίζεται παρακάτω).

Προσδιορισμός

`void OnRelease()`

Παράμετροι

Δεν έχει.

Header file

GameLoseMenu.h

Περιγραφή

Σχετίζεται με το πάτημα των παρακάτω κουμπιών.

- **Κουμπί FNewGameButton** : Πρόκειται για το κουμπί νέο παιχνίδι το οποίο όταν πατηθεί ο χρήστης επιστρέφει στο μενού της έναρξης του παιχνιδιού, όπου μπορεί να αλλάξει την δυσκολία και την κατηγορία του παιχνιδιού.

Προσδιορισμός

`void Render()`

Παράμετροι

Δεν έχει.

Header file

GameLoseMenu.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Περιγραφή

Απεικονίζει με γραφικά το μενού της ήττας του παιχνιδιού.

GameRecordsMenu

Προσδιορισμός

`bool IsValid() const`

Παράμετροι

Δεν έχει.

Header file

GameRecordsMenu.h

Περιγραφή

Ελέγχει αν το όνομα που πληκτρολογήσε ο χρήστης είναι μεγαλύτερο των τριων χαρακτήρων και ο πρώτος χαρακτήρας δεν είναι νούμερο. Σε αυτή τη περίπτωση επιστρέφει 1, αλλιώς 0.

Προσδιορισμός

`FString &GetPlayerName() const`

Παράμετροι

Δεν έχει.

Header file

GameRecordsMenu.h

Περιγραφή

Επιστρέφει το όνομα που πληκτρολόγησε ο χρήστης.

Προσδιορισμός

`FRecordsMenu()`

Παράμετροι

Δεν έχει.

Header file

GameRecordsMenu.h

Περιγραφή

Δημιουργεί το μενού ρεκόρ του παιχνιδιού. Περιέχει όλα τα κουμπιά και την λογική πίσω από αυτά, καθώς και τα γραφικά και τη μορφή κειμένου που περιέχουν. (Προσοχή! Η απεικόνιση γίνεται μέσω της εντολής Render που ορίζεται παρακάτω).

Προσδιορισμός

```
void OnRelease()
```

Παράμετροι

Δεν έχει.

Header file

GameLoseMenu.h

Περιγραφή

Σχετίζεται με το πάτημα των παρακάτω κουμπιών.

- **Κουμπί Home** : Επιστρέφει τον χρήστη στο αρχικό μενού.
- **Κουμπί FTickButton** : Πρόκειται για το κουμπί που εμφανίζεται ως το σύμβολο «tick» το οποίο ελέγχει αν το όνομα που πληκτρολογήσαμε υπάρχει. Σε περίπτωση που υπάρχει εμφανίζεται ο παίχτης με το αντίστοιχο ρεκόρ του, αλλιώς εμφανίζει μήνυμα λάθους.

Προσδιορισμός

```
void Render()
```

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δεν έχει.

Header file

GamePlayMenu.h

Περιγραφή

Διακρίνεται σε

- Render που ανήκει στην κλάση **FRecordsNameButton** όπου απεικονίζει το «placeholder», το όνομα, και την πληκτρολόγηση του με γραφικά.
- Render που ανήκει στην κλάση **FRecordsMenu** όπου απεικονίζει με γραφικά το μενού ρεκόρ του παιχνιδιού.

5.3.6 Δομή Renderer

CanvasButton

Προσδιορισμός

FContentBrush()

Παράμετροι

Δεν έχει.

Header file

CanvasButton.h

Περιγραφή

Δημιουργεί ένα αντικείμενο χωρίς εικόνα και χρώμα.

Προσδιορισμός

FContentBrush(const FColor32 &InColorAndOpacity)

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

const FColor32 &InColorAndOpacity : Χρώμα προς εμφάνιση.

Header file

CanvasButton.h

Περιγραφή

Δημιουργεί ένα χρώμα με το οποίο θα εμφανιστεί κουμπί.

Προσδιορισμός

FContentBrush(*const* FTexture2D *InSprite)

Παράμετροι

*const FTexture2D *InSprite* : Εικόνα προς εμφάνιση.

Header file

CanvasButton.h

Περιγραφή

Δημιουργεί την εικόνα με την οποία θα εμφανιστεί το κουμπί.

Προσδιορισμός

FContentBrush(*const* FTexture2D *InSprite, *const* FColor32 &InColorAndOpacity)

Παράμετροι

*const FTexture2D *InSprite* : Εικόνα προς εμφάνιση.

const FColor32 &InColorAndOpacity : Χρώμα προς εμφάνιση.

Header file

CanvasButton.h

Περιγραφή

Δημιουργεί τον συνδυασμό εικόνας,χρώματος με τα οποία θα εμφανιστεί το κουμπί.

Προσδιορισμός

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

`FCanvasButton()`

Παράμετροι

Δεν έχει.

Header file

CanvasButton.h

Περιγραφή

Δημιουργείται αντικείμενο *FCanvasButton*(Κουμπί)

Προσδιορισμός

`virtual ~FCanvasButton()`

Παράμετροι

Δεν έχει.

Header file

CanvasButton.h

Περιγραφή

Διαγράφεται από την μνήμη το κουμπί που δημιουργείθηκε.

Προσδιορισμός

`virtual void OnHover()`

Παράμετροι

Δεν έχει.

Header file

CanvasButton.h

Περιγραφή

Ελέγχει αν ο δείκτης ποντικιού βρίσκεται πάνω στο κουμπί.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

```
virtual void OnClick()
```

Παράμετροι

Δεν έχει.

Header file

CanvasButton.h

Περιγραφή

Ελέγχει αν το κουμπί πατήθηκε.

Προσδιορισμός

```
virtual void OnRelease()
```

Παράμετροι

Δεν έχει.

Header file

CanvasButton.h

Περιγραφή

Ελέγχει αν το κουμπί σταμάτησε να πατιέται.

Προσδιορισμός

```
virtual void Render(FCanvasRenderer *InCanvasRenderer)
```

Παράμετροι

*FCanvasRenderer *InCanvasRenderer* : Ο τύπος *Renderer* που θα χρησιμοποιηθεί.

Header file

CanvasButton.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Εμφανίζει στην οθόνη το κουμπί με γραφικά.

Προσδιορισμός

`FCanvasMenu *GetMenu()`

Παράμετροι

Δεν έχει.

Header file

CanvasButton.h

Περιγραφή

Επιστρέφει το μενού στο οποίο ανήκει.

Προσδιορισμός

`FCanvasMenuState *GetMenuState()`

Παράμετροι

Δεν έχει.

Header file

CanvasButton.h

Περιγραφή

Επιστρέφει τη κατηγορία μενού στο οποίο ανήκει.

CanvasMenu

Προσδιορισμός

`FCanvasMenu()`

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δεν έχει.

Header file

CanvasMenu.h

Περιγραφή

Δημιουργεί ένα μενού που αποτελείται από κουμπία.

Προσδιορισμός

```
virtual ~FCanvasMenu()
```

Παράμετροι

Δεν έχει.

Header file

CanvasMenu.h

Περιγραφή

Διαγράφει το μενού και τη σχετική μνήμη που δεσμεύει.

Προσδιορισμός

```
void AddButton(FCanvasButton *InButton)
```

Παράμετροι

*FCanvasButton *InButton* : Αντικείμενο κουμπιού.

Header file

CanvasMenu.h

Περιγραφή

Προσθέτει ένα κουμπί στην στίβα.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

```
void RemoveButton(FCanvasButton *InButton)
```

Παράμετροι

*FCanvasButton *InButton* : Αντικείμενο κουμπιού.

Header file

CanvasMenu.h

Περιγραφή

Αφαιρεί από την στίβα ένα κουμπί.

Προσδιορισμός

```
virtual void ReInitialize()
```

Παράμετροι

Δεν έχει.

Header file

CanvasMenu.h

Περιγραφή

Αρχικοποιεί το σκορ σε 0 και διαγράφει το όνομα του παίχτη.

Προσδιορισμός

```
virtual void Update(FInputState *InputState)
```

Παράμετροι

*FInputState *InputState* : Κατάσταση του κουμπιού.

Header file

CanvasMenu.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Ενημερώνει την κατάσταση στην οποία βρίσκεται το κουμπί.

Προσδιορισμός

`FCanvasMenuState *GetMenuState()`

Παράμετροι

Δεν έχει.

Header file

CanvasMenu.h

Περιγραφή

Επιστρέφει το μενού.

Προσδιορισμός

`void Render(FCanvasRenderer *InCanvasRenderer)`

Παράμετροι

*FCanvasRenderer *InCanvasRenderer* : Ο τύπος *Renderer* που θα χρησιμοποιηθεί.

Header file

CanvasMenu.h

Περιγραφή

Εμφανίζει στην οθόνη το μενού με γραφικά.

CanvasMenuState

Προσδιορισμός

`virtual ~FCanvasMenuState()`

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δεν έχει.

Header file

CanvasMenuState.h

Περιγραφή

Διαγράφει όλα τα μενού και τη σχετική μνήμη τους.

Προσδιορισμός

```
void AddMenu(FCanvasMenu *InMenu)
```

Παράμετροι

*FCanvasMenu *InMenu* : Προσθήκη αντικειμένου μενού.

Header file

CanvasMenuState.h

Περιγραφή

Προσθέτει ένα μενού στην στίβα.

Προσδιορισμός

```
void RemoveMenu(FCanvasMenu *InMenu)
```

Παράμετροι

*FCanvasMenu *InMenu* : Αντικείμενο μενού προς αφαίρεση.

Header file

CanvasMenuState.h

Περιγραφή

Αφαιρεί ένα μενού από την στίβα.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

```
virtual void UpdateAndRender(FCanvasRenderer *InCanvasRenderer, FInputState  
*InputState)
```

Παράμετροι

*FCanvasRenderer *InCanvasRenderer* : Ο τύπος *Renderer* που θα χρησιμοποιηθεί.

*FInputState *InputState* : Κατάσταση του κουμπιού.

Header file

CanvasMenuState.h

Περιγραφή

Ενημερώνει και εμφανίζει στην οθόνη το μενού που έχουμε επιλέξει εκείνη την στιγμή.

Προσδιορισμός

```
FORCEINLINE FCanvasMenu *GetMenuById(int32 Id)
```

Παράμετροι

int32 Id : Κλειδί του μενού.

Header file

CanvasMenuState.h

Περιγραφή

Επιστρέφει το μενού με βάση το κλειδί.

Προσδιορισμός

```
FORCEINLINE FCanvasMenu *GetMenu()
```

Παράμετροι

Δεν έχει.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Header file

CanvasMenuState.h

Περιγραφή

Επιστρέφει το μενού.

Προσδιορισμός

FORCEINLINE FCanvasMenu *GetMenuState()

Παράμετροι

Δεν έχει.

Header file

CanvasMenuState.h

Περιγραφή

Επιστρέφει το μενού.

FontCache

Προσδιορισμός

FFontCache()

Παράμετροι

Δεν έχει.

Header file

FontCache.h

Περιγραφή

Δημιουργεί και επιστρέφει ένα αντικείμενο στο οποίο θα φορτωθούν χαρακτήρες από ένα bitmap.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

`~FFontCache()`

Παράμετροι

Δεν έχει.

Header file

FontCache.h

Περιγραφή

Διαγράφει το αντικείμενο και την σχετική μνήμη που δεσμεύει.

Προσδιορισμός

`void LoadFont(const TCHAR *Filename)`

Παράμετροι

*const TCHAR *Filename* : Path στο οποίο βρίσκεται το αρχείο.

Header file

FontCache.h

Περιγραφή

Φορτώνει χαρακτήρες από ένα αρχείο ttf το οποίο βρίσκεται από την παράμετρο. Επιπλέον ορίζει το μέγεθος των χαρακτήρων.

Προσδιορισμός

`void GetFontBoundRect(const FString &InText, float InTextSize,
uint32 *OutBoundSizeX, uint32 *OutBoundSizeY) const`

Παράμετροι

const FString &InText : Αλφαριθμητικό.

float InTextSize : Μέγεθος αλφαριθμητικού σε bytes.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

*uint32 *OutBoundSizeX* : Μέγεθος αλφαριθμητικού ως προς τον άξονα x.

*uint32 *OutBoundSizeY* : Μέγεθος αλφαριθμητικού ως προς τον άξονα y.

Header file

FontCache.h

Περιγραφή

Υπολογίζει το μέγεθος ενός αλφαριθμητικού σε pixels που εμφανίζεται στην οθόνη ως προς τον άξονα x και y.

Προσδιορισμός

```
const FTexture2D *GetFontTexture() const
```

Παράμετροι

Δεν έχει.

Header file

FontCache.h

Περιγραφή

Επιστρέφει ένα αντικείμενο FTexture2D (Μια εικόνα δηλαδή).

Προσδιορισμός

```
const FTextCharacter &GetCharacter(uint8 Index) const
```

Παράμετροι

uint8 Index : Θέση χαρακτήρα προς εύρεση.

Header file

FontCache.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Επιστρέφει τον χαρακτήρα που βρίσκεται στην παράμετρο *Index*.

CanvasRenderer

Προσδιορισμός

```
virtual ~FRendererItem()
```

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Διαγράφει όλα τα instances του renderer και τη σχετική μνήμη που δεσμεύουν.

Προσδιορισμός

```
virtual void Render()
```

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Πρόκειται για την συνάρτηση που εμφανίζει στην οθόνη με γραφικά τις εντολές που του δίνουμε. Διακρίνεται σε TileRenderer(δηλαδή σε αντικείμενα που έχουν μόνο χρώμα) , σε SpriteRenderer(δηλαδή σε αντικείμενα που έχουν εικόνα ή και χρώμα) και σε TextRenderer(δηλαδή σε αντικείμενα που έχουν κείμενο).

Προσδιορισμός

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

```
virtual FTileRenderer *GetTileRenderer()
```

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Επιστρέφει το αντικείμενο Renderer που αφορά εμφάνιση μόνο με χρώμα.

Προσδιορισμός

```
virtual FSpriteRenderer *GetSpriteRenderer()
```

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Επιστρέφει το αντικείμενο Renderer που αφορά εμφάνιση εικόνα και χρώμα.

Προσδιορισμός

```
virtual FTextRenderer *GetTextRenderer()
```

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Επιστρέφει το αντικείμενο Renderer που αφορά εμφάνιση με μορφή κειμένου στην οθόνη.

Προσδιορισμός

`FCanvasRenderer();`

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Δημιουργεί τον γενικό Renderer κάνβας.

Προσδιορισμός

`~FCanvasRenderer();`

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Διαγράφει το αντικείμενο και την σχετική μνήμη που δεσμεύει.

Προσδιορισμός

`void FlushToViewport();`

Παράμετροι

Δεν έχει.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Header file

CanvasRenderer.h

Περιγραφή

Προετοιμάζει προς εμφάνιση στην οθόνη τον Renderer και τους ταξινομεί ανάλογα το DepthSortKey.

Προσδιορισμός

```
void SortElements();
```

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Ταξινομεί τα RenderBatches ανάλογα το κλειδί τους.

Προσδιορισμός

```
void DrawSortedElements();
```

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Εμφανίζει στην οθόνη ότι εργασία δόθηκε στον κάνβας Renderer ανάλογα το κλειδί DepthsortKey.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

`void PostDrawElements()`

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Διαγράφει όλη την σχετική μνήμη των RenderBatches, τα αδειάζει και αρχικοποιεί το κλειδί σε 0.

Προσδιορισμός

`void PushDepthSortKey(int32 SortKey)`

Παράμετροι

`int32 SortKey` : Κλειδί προς ταξινόμηση.

Header file

CanvasRenderer.h

Περιγραφή

Προσθέτει ένα κλειδί προς ταξινόμηση στην στίβα.

Προσδιορισμός

`void PopDepthSortKey()`

Παράμετροι

Δεν έχει.

Header file

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

CanvasRenderer.h

Περιγραφή

Επιστρέφει το πάνω πάνω κλειδί προς ταξινόμηση από την στίβα και την αφαιρεί.

Προσδιορισμός

```
int32 GetDepthSortKey()
```

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Επιστρέφει το κλειδί προς ταξινόμηση DepthSortKey.

Προσδιορισμός

```
void DrawTile(int32 X, int32 Y, int32 SizeX, int32 SizeY,  
              const FColor32 &Color, EContentAlignX AlignX,  
              EContentAlignY AlignY)
```

Παράμετροι

<i>int32 X</i>	: Σημείο X.
<i>int32 Y</i>	: Σημείο Y.
<i>int32 SizeX</i>	: Μήκος αντικειμένου.
<i>int32 SizeY</i>	: Πλάτος αντικειμένου.
<i>const FColor32 &Color</i>	: Χρώμα αντικειμένου.
<i>EContentAlignX AlignX</i>	: Ευθυγραμμία ως προς x.
<i>EContentAlignY AlignY</i>	: Ευθυγραμμία ως προς y.

Header file

CanvasRenderer.h

Περιγραφή

Εμφανίζει στην οθόνη ένα ορθογώνιο στα σημεία x,y με μήκος, πλάτος και χρώμα που ορίζεται από τις παραμέτρους με ευθυγράμμιση.

Προσδιορισμός

```
void DrawSprite(const FTexture2D *Sprite, int32 X, int32 Y,  
               int32 SizeX, int32 SizeY, const FColor32 &Color,  
               EContentAlignX AlignX, EContentAlignY AlignY)
```

Παράμετροι

<i>const FTexture2D *Sprite</i>	: Εικόνα.
<i>int32 X</i>	: Σημείο X.
<i>int32 Y</i>	: Σημείο Y.
<i>int32 SizeX</i>	: Μήκος αντικειμένου.
<i>int32 SizeY</i>	: Πλάτος αντικειμένου.
<i>const FColor32 &Color</i>	: Χρώμα αντικειμένου.
<i>EContentAlignX AlignX</i>	: Ευθυγράμμιση ως προς x.
<i>EContentAlignY AlignY</i>	: Ευθυγράμμιση ως προς y.

Header file

CanvasRenderer.h

Περιγραφή

Εμφανίζει στην οθόνη ένα ορθογώνιο στα σημεία x,y με μήκος, πλάτος,εικόνα και χρώμα που ορίζεται από τις παραμέτρους με ευθυγράμμιση.

Προσδιορισμός

```
void DrawText(const FFontCache *FontCache, float FontScale,
```

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

```
const FString &Text, int32 X, int32 Y,  
const FColor32 &Color, EContentAlignX AlignX,  
EContentAlignY AlignY)
```

Παράμετροι

<i>const FFontCache *FontCache</i>	: Γραμματοσειρά.
<i>float FontScale</i>	: Μέγεθος γραμματοσειράς.
<i>const FString &Text</i>	: Αλφαριθμητικό λέξης.
<i>int32 X</i>	: Σημείο X.
<i>int32 Y</i>	: Σημείο Y.
<i>const FColor32 &Color</i>	: Χρώμα γραμματοσειράς.
<i>EContentAlignX AlignX</i>	: Ευθυγραμμία ως προς x.
<i>EContentAlignY AlignY</i>	: Ευθυγραμμία ως προς y.

Header file

CanvasRenderer.h

Περιγραφή

Εμφανίζει στην οθόνη μια λέξη στα σημεία x,y με μέγεθος και χρώμα γραμματοσειράς που ορίζεται απο τις παραμέτρους με ευθυγράμμιση.

Προσδιορισμός

```
FTileRenderer *FindOrCreateTileRenderer()
```

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Ελέγχει αν υπάρχει το αντικείμενο του *TileRenderer* με το ορισμένο *DepthSortKey*. Σε επιτυχία το επιστρέφει αλλιώς το δημιουργεί.

Προσδιορισμός

`FSpriteRenderer *FindOrCreateSpriteRenderer()`

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Ελέγχει αν υπάρχει το αντικείμενο του *SpriteRenderer* με το ορισμένο *DepthSortKey*. Σε επιτυχία το επιστρέφει αλλιώς το δημιουργεί.

Προσδιορισμός

`FTextRenderer *FindOrCreateTextRenderer()`

Παράμετροι

Δεν έχει.

Header file

CanvasRenderer.h

Περιγραφή

Ελέγχει αν υπάρχει το αντικείμενο του *TextRenderer* με το ορισμένο *DepthSortKey*. Σε επιτυχία το επιστρέφει αλλιώς το δημιουργεί.

[SpriteRenderer](#)

Προσδιορισμός

`FSpriteRenderer()`

Παράμετροι

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Δεν έχει.

Header file

SpriteRenderer.h

Περιγραφή

Δημιουργεί και αρχικοποιεί ένα αντικείμενο τύπου *SpriteRenderer*.

Προσδιορισμός

~FSpriteRenderer()

Παράμετροι

Δεν έχει.

Header file

SpriteRenderer.h

Περιγραφή

Διαγράφει το αντικείμενο και την σχετική μνήμη που δεσμεύει.

Προσδιορισμός

virtual FSpriteRenderer *GetSpriteRenderer()

Παράμετροι

Δεν έχει.

Header file

SpriteRenderer.h

Περιγραφή

Επιστρέφει το αντικείμενο τύπου *SpriteRenderer*.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

```
void AddItem(const FCanvasSpriteItem &Item)
```

Παράμετροι

const FCanvasSpriteItem &Item : Αντικείμενο *FcanvasSpriteItem* προς αφαίρεση.

Header file

SpriteRenderer.h

Περιγραφή

Προσθέτει ένα αντικείμενο *FcanvasSpriteItem* στην στίβα η οποία θα εμφανιστεί στην οθόνη.

Προσδιορισμός

```
void RemoveItem(const FCanvasSpriteItem &Item)
```

Παράμετροι

const FCanvasSpriteItem &Item : Αντικείμενο *FcanvasSpriteItem* προς εμφάνιση.

Header file

SpriteRenderer.h

Περιγραφή

Αφαιρεί ένα αντικείμενο *FcanvasSpriteItem* από την στίβα.

Προσδιορισμός

```
FORCEINLINE const FMatrix  
GetOrthographic(float X, float Y, float Width, float Height)
```

Παράμετροι

float X : Σημείο X.

float Y : Σημείο Y.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

float width : Μήκος.

float height : Ύψος.

Header file

SpriteRenderer.h

Περιγραφή

Εμφανίζει το αντικείμενο σε Orthographic Projection(δηλαδή σε 2D) στα σημεία, το μήκος και ύψος που ορίζεται από τις παραμέτρους.

TileRenderer

Προσδιορισμός

FTileRenderer()

Παράμετροι

Δεν έχει.

Header file

TileRenderer.h

Περιγραφή

Δημιουργεί και αρχικοποιεί ένα αντικείμενο τύπου *TileRenderer*.

Προσδιορισμός

~FTileRenderer()

Παράμετροι

Δεν έχει.

Header file

TileRenderer.h

Περιγραφή

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Διαγράφει το αντικείμενο και την σχετική μνήμη που δεσμεύει.

Προσδιορισμός

```
virtual FTileRenderer *GetTileRenderer()
```

Παράμετροι

Δεν έχει.

Header file

TileRenderer.h

Περιγραφή

Επιστρέφει το αντικείμενο τύπου *TileRenderer*.

Προσδιορισμός

```
void AddItem(const FCanvasTileItem &Item)
```

Παράμετροι

const FCanvasTileItem &Item : Αντικείμενο *FcanvasTileItem* προς εμφάνιση.

Header file

TileRenderer.h

Περιγραφή

Προσθέτει ένα αντικείμενο *FcanvasTileItem* στην στίβα η οποία θα εμφανιστεί στην οθόνη.

Προσδιορισμός

```
void RemoveItem(const FCanvasTileItem &Item)
```

Παράμετροι

const FCanvasTileItem &Item : Αντικείμενο *FcanvasTileItem* προς διαγραφή.

Header file

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

TileRenderer.h

Περιγραφή

Αφαιρεί ένα αντικείμενο *FcanvasTileItem* από την στίβα.

Προσδιορισμός

```
FORCEINLINE const FMatrix  
GetOrthographic(float X, float Y, float Width, float Height)
```

Παράμετροι

float X : Σημείο X.

float Y : Σημείο Y.

float Width : Μήκος.

float Height : Ύψος.

Header file

TileRenderer.h

Περιγραφή

Εμφανίζει το αντικείμενο σε Orthographic Projection(δηλαδή σε 2D) στα σημεία, το μήκος και ύψος που ορίζεται απο τις παραμέτρους.

TextRenderer

Προσδιορισμός

```
FTextRenderer()
```

Παράμετροι

Δεν έχει.

Header file

TextRenderer.h

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Περιγραφή

Δημιουργεί και αρχικοποιεί ένα αντικείμενο τύπου *TextRenderer*.

Προσδιορισμός

`~FTextRenderer()`

Παράμετροι

Δεν έχει.

Header file

TextRenderer.h

Περιγραφή

Επιστρέφει το αντικείμενο και τη σχετική μνήμη που δεσμεύει.

Προσδιορισμός

`virtual FTextRenderer *GetTextRenderer()`

Παράμετροι

Δεν έχει.

Header file

TextRenderer.h

Περιγραφή

Επιστρέφει το αντικείμενο τύπου *TextRenderer*.

Προσδιορισμός

`void AddItem(const FCanvasTextItem &Item)`

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Παράμετροι

const FCanvasTextItem &Item : Αντικείμενο *FCanvasTextItem* προς εμφάνιση.

Header file

TextRenderer.h

Περιγραφή

Προσθέτει ένα αντικείμενο *FCanvasTextItem* στην στίβα η οποία θα εμφανιστεί στην οθόνη.

Προσδιορισμός

```
void RemoveItem(const FCanvasTextItem &Item)
```

Παράμετροι

const FCanvasTextItem &Item : Αντικείμενο *FCanvasTextItem* προς διαγραφή.

Header file

TextRenderer.h

Περιγραφή

Αφαιρεί ένα αντικείμενο *FCanvasTextItem* από την στίβα.

Προσδιορισμός

```
FORCEINLINE const FMatrix  
GetOrthographic(float X, float Y, float Width, float Height)
```

Παράμετροι

float X : Σημείο X.

float Y : Σημείο Y.

float Width : Μήκος.

float Height : Ύψος.

Header file

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

TextRenderer.h

Περιγραφή

Εμφανίζει το αντικείμενο σε Orthographic Projection(δηλαδή σε 2D) στα σημεία, το μήκος και ύψος που ορίζεται απο τις παραμέτρους.

RenderResourceAllocator

Προσδιορισμός

```
static FRenderResourceAllocator *Get()
```

Παράμετροι

Δεν έχει.

Header file

RenderResourceAllocator.h

Περιγραφή

Επιστρέφει τον global allocator.

Προσδιορισμός

```
bool CreateVertexBuffer(FVertexBufferAllocation **OutAllocation,uint64  
Capacity)
```

Παράμετροι

*FVertexBufferAllocation **OutAllocation* : Δεσμευμένη ποσότητα μνήμης.

uint64 Capacity : Ποσότητα Μνήμης.

Header file

RenderResourceAllocator.h

Περιγραφή

Δημιουργεί έναν ενιάιο VertexBuffer με προκαθορισμένη μνήμη.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Προσδιορισμός

```
bool FreeVertexBuffer(FVertexBufferAllocation *InAllocation)
```

Παράμετροι

*FVertexBufferAllocation *InAllocation* : Δεσμευμένη ποσότητα μνήμης.

Header file

RenderResourceAllocator.h

Περιγραφή

Αποδεσμεύει τον *VertexBuffer* από την μνήμη.

Προσδιορισμός

```
bool CreateIndexBuffer(FIndexBufferAllocation **OutAllocation,  
                      uint64 Capacity)
```

Παράμετροι

*FIndexBufferAllocation **OutAllocation* : Δεσμευμένη ποσότητα μνήμης.

uint64 Capacity : Ποσότητα Μνήμης.

Header file

RenderResourceAllocator.h

Περιγραφή

Δημιουργεί έναν ενιάιο *IndexBuffer* με προκαθορισμένη μνήμη.

Προσδιορισμός

```
bool FreeIndexBuffer(FIndexBufferAllocation *InAllocation)
```

Παράμετροι

*FIndexBufferAllocation *InAllocation* : Δεσμευμένη ποσότητα μνήμης.

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Header file

RenderResourceAllocator.h

Περιγραφή

Αποδεσμεύει τον *IndexBuffer* από την μνήμη.

Προσδιορισμός

```
void TickPendingResources()
```

Παράμετροι

Δεν έχει.

Header file

RenderResourceAllocator.h

Περιγραφή

Ελέγχει κάθε 3 frames αν δεν έχει χρησιμοποιηθεί κάποιο allocator resource.

Προσδιορισμός

```
void FreePendingResources()
```

Παράμετροι

Δεν έχει.

Header file


RenderResourceAllocator.h

Περιγραφή

Διαγράφει το allocator resource που δεν έχει χρησιμοποιηθεί για 3 frames.

Κεφάλαιο 6°

6.1 Βιβλιογραφία-Πηγές-Αναφορές

1. Κρεμάλα (παιχνίδι), Βικιπεδία, [https://el.wikipedia.org/wiki/Κρεμάλα_\(παιχνίδι\)](https://el.wikipedia.org/wiki/Κρεμάλα_(παιχνίδι))
2. [Μάθηση με Ψηφιακά Παίγνια στο Σημασιολογικό Ιστό: διεθνοποιημένη μέθοδος μέσω της Ελληνικής DBpedia Learning with Digital Games in Semantic Web: Internationalized Method via the Greek DBpedia](#), Δέσποινα-Ελεάνα Ιωάννη Χρυσού, 2012, [http://invenio.lib.auth.gr/record/131020/files/GRI-2013-9852.pdf], Μεταπτυχιακή διπλωματική εργασία, [ΑΠΘ](#)
3. [↑estcomp.ro, κρεμάλα](#)
4. Εφαρμογή υπολογιστή, Βικιπεδία, https://el.wikipedia.org/wiki/Εφαρμογή_υπολογιστή
5. «Η Προέλευση της C++», Ζαφείρης Φιλίτσας. Ανακτήθηκε στις 2006-08-25.
6. [The invention of C++](#) — Συνέντευξη του Μπιάρνε Στρούστρουπ για τη δημιουργία της γλώσσας. (Αγγλικά)
7. [C++ στο DMOZ](#)
8. Διεπαφή προγραμματισμού εφαρμογών, Βικιπεδία, https://el.wikipedia.org/wiki/Διεπαφή_προγραμματισμού_εφαρμογών
9. OpenGL, Βικιπεδία, <https://el.wikipedia.org/wiki/OpenGL>
10. [«Products: Software: OpenGL: Licensing and Logos»](#). SGI. Αρχειοθετήθηκε [από το πρωτότυπο](#) στις 1 Νοεμβρίου 2012. Ανακτήθηκε στις 7 Νοεμβρίου 2012.
11. [«Πρότυπο OpenGL 4.0»](#) (PDF).
12. [«Khronos Membership»](#). *The Khronos Group*. Ανακτήθηκε στις 15 Νοεμβρίου 2016.
13. [«WebGL - OpenGL ES 2.0 for the Web»](#). *The Khronos Group*. Ανακτήθηκε στις 15 Νοεμβρίου 2016.
14. [«WGL \(API\)»](#) (στα αγγλικά). *Wikipedia*. 2016-07-08.
15. [«Core OpenGL»](#) (στα αγγλικά). *Wikipedia*. 2016-04-03.
16. [«GLX»](#) (στα αγγλικά). *Wikipedia*. 2016-09-01.
17. Inc, The Khronos Group. [«OpenGL® Registry»](#). www.opengl.org. Ανακτήθηκε στις 15 Νοεμβρίου 2016.
18. http://www.opengl.org/registry/specs/NV/half_float.txt
19. [«Immediate mode \(computer graphics\)»](#) (στα αγγλικά). *Wikipedia*. 2016-06-07.
20. [«OpenGL ARB to Pass Control of OpenGL Specification to Khronos Group - Khronos Group Press Release»](#). *The Khronos Group*. Ανακτήθηκε στις 15 Νοεμβρίου 2016.
21. Digital Pedagogy - A Guide for Librarians, Faculty, and Students , <https://guides.library.utoronto.ca/c.php?g=448614&p=3505475>
22. Games and learning, Βικιπεδία, https://el.wikipedia.org/wiki/Games_and_learning
23. Shaffer, D., Squire, K., Halverson, R., & Gee, J. P. (2005)
24. [Jump up to: ↑](#) Gee, James Paul (2010)
25. National Research Council, 2005
26. Salen, K., & Zimmerman, E. 2004
27. Clark, Nelson, Sengupta, & D'Angelo, 2009
28. Klopfer, Osterweil, & Salen, 2009
29. Koster, 2004
30. McGonigal, 2011
31. Annetta, Minogue, Holmes, & Cheng, 2009
32. Clark et al., 2011
33. Hickey, Ingram-Goble, & Jameson, 2009
34. Ketelhut, Dede, Clarke, & Nelson, 2006
35. Klopfer, Scheintaub, Huang, Wendal, & Roque, 2009
36. Moreno & Mayer, 2000, 2004
37. Barr, Matthew (2017). ["Video games can develop graduate skills in higher education students"](#). *Computers & Education*. **113**: 86–97. doi:10.1016/j.compedu.2017.05.016. 

Πανεπιστήμιο Δυτικής Αττικής
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

38. Barab et al., 2007
39. Neulight, Kafai, Kao, Foley, & Galas, 2007
40. Barab et al., 2009
41. Galas, 2006
42. McQuiggan, Rowe, & Lester, 2008
43. Clark, D, Tanner-Smith, E., Killingsworth, S., & Bellamy, S. (2013)
44. [Jump up to: ^a ^b](#) Gee, J. P. (2003). *What Video Games Have to Teach Us About Learning and Literacy*. New York: Palgrave Macmillan. [ISBN 978-1-4039-6538-7](#)
45. [Jump up to: ^a ^b ^c ^d ^e](#) Malykhina, E. (2014). ["Fact or Fiction?: Video Games Are the Future of Education"](#). *Scientific American*.
46. Chiappe, Dan, Mark Conger, Janet Liao, J. Lynn Caldwell, and Kim-Phoung L. Vu
47. Sitzmann, Traci (Summer 2011). "A meta-analytic examination of the instructional effectiveness of computer-based simulation games". *Personnel Psychology*. 64. Issue 2 (2): 489–528. [doi:10.1111/j.1744-6570.2011.01190.x](#).
48. Wouters, van Nimwegen, van Oostendorp, van der Spek, Pieter, Christof, Herre, Erik D. (February 4, 2013). ["A Meta-Analysis of the Cognitive and Motivational Effects of Serious Games"](#) (PDF). *Journal of Educational Psychology*. 105 (2): 249–265. [doi:10.1037/a0031311](#).
49. Iowa State University (March 24, 2014). ["Life lessons: Children learn aggressive ways of thinking and behaving from violent video games, study finds"](#). *ScienceDaily*. Retrieved August 20, 2016.
50. Iowa State University Study, 2014
51. Bergland, C. 2013
52. Κρεμάλα (παιχνίδι) , Βικιπεδία, [https://el.wikipedia.org/wiki/Κρεμάλα_\(παιχνίδι\)](https://el.wikipedia.org/wiki/Κρεμάλα_(παιχνίδι))
53. Welcome to OpenGL, <https://learnopengl.com/>
54. Get Started with Win32 and C++, <https://docs.microsoft.com/en-us/windows/win32/learnwin32/learn-to-program-for-windows>