



Project αντικειμενοστραφούς προγραμματισμού 3ου εξαμήνου υλοποιημένο από τους :

<b>Αλέξανδρος Συμπέθερος</b>	<b>1115 2009 00261</b>
<b>Ιωάννης Χρόνης</b>	<b>1115 2009 00125</b>

Αρχεία που περιέχει το project.zip :

- block.cpp
- functions.cpp
- grid.cpp
- hero.cpp
- item.cpp
- living.cpp
- main.cpp
- monster.cpp
- Party.cpp
- position.cpp
- spell.cpp
- block.h
- grid.h
- header.h
- hero.h
- item.h
- living.h
- monster.h
- Party.h
- position.h
- spell.h
- map01.txt
- armor\_list.txt
- firespell\_list.txt
- icespell\_list.txt
- lightning\_list.txt
- monster\_list.txt
- potion\_list.txt
- weapon\_list.txt
- README.pdf
- README.txt

Εντολή για τη δημιουργία του εκτελεσίμου:

**g++ -o ChronicSeg.exe block.cpp functions.cpp grid.cpp hero.cpp item.cpp living.cpp main.cpp monster.cpp Party.cpp position.cpp spell.cpp**

Εντολή εκτέλεσης προγράμματος :

**ChronicSeg.exe**

(Τα αρχεία \*.txt πρέπει να βρίσκονται στον ίδιο κατάλογο με το εκτελέσιμο.)

## Τρόπος Λειτουργίας

Το πρόγραμμα έχει μοντελοποιηθεί και τρέχει με το εξής τρόπο :

Στη **main** δημιουργούνται στιγμιότυπα των κλάσεων grid, party και καλείται η συνάρτηση operate της κλάσης party, αυτή ουσιαστικά λειτουργεί το παιχνίδι.

**Grid** : είναι μία κλάση που περιέχει ένα δισδιάστατο πίνακα με δείκτες προς blocks (Nonaccessible, Market, Common). Τα blocks δημιουργούνται μέσα κατά την δημιουργία του grid. Το μέγεθος του grid καθώς και η διάταξη των διαφόρων ειδών block στο grid λαμβάνεται ως δεδομένο από ένα αρχείο (εδώ: map01.txt). Στο Grid υπάρχουν Common, Nonaccessible και ένα Market block. Υπάρχουν 3 χαρτες καλό θα ήταν να επιλεγεται τον πρώτο καθώς είναι πιο μικρός και το market είναι κοντά στο σημείο εκκίνησης.

**Market** : περιέχει λίστες με weapons, armors, potion, spells διαθέσιμα προς αγορά. Οι λίστες αυτές δημιουργούνται μαζί με το market και τα αντικείμενα προς πώληση διαβάζονται από αρχεία που περιλαμβάνουν τα ονόματα και τα χαρακτηριστικά τους, ώστε κάθε φορά να είναι διαθέσιμα τα ίδια αντικείμενα.

**Party** είναι μια κλάση που δημιουργεί και διατηρεί μια λίστα με τους ήρωες και τη θέση τους στο grid. Σκοπός της party είναι να πακετάρει τους ήρωες και τις λειτουργίες τους.

**Operate** : συνάρτηση της party, αναλαμβάνει να επικοινωνεί με το grid, να μετακινήσει τους παίκτες και να καλεί την συνάρτηση μέλος κάθε block engage, μέσα στην οποία γίνονται ο χρήστης αλληλεπιδρά με το συγκεκριμένο block.

**Engage** : συνάρτηση μέλος κάθε block. Αναλαμβάνει να δείξει στο χρήστη τις επιλογές που είναι διαθέσιμες για κάθε περίπτωση block και να τις εκτελέσει. Σε κάθε επίσκεψη του party σε Common τετράγωνο κατά την είσοδο στην engage αποφασίζεται βάση μιας πιθανότητας αν τεράτα θα επιτεθούν στο party. Τα τεράτα δημιουργούνται στην engage και τα χαρακτηριστικά τους υπολογίζονται βάση των αντίστοιχων χαρακτηριστικών των ηρώων του party, ο αριθμός των τεράτων υπολογίζεται τυχαία εντός ορίων. Η μάχη ξεκινάει αμέσως.

**Battle** : συνάρτηση μέλος του block Common. Έχει το δικό της μενού, διεξάγεται σε γύρους. Κάθε γύρος περιλαμβάνει μία ενέργεια κάθε ήρωα και επίθεση κάθε τεράτος προς κάποιο ήρωα. Με τη σειρά ο χρήστης επιλέγει την ενέργεια κάθε ήρωα. Τα τεράτα επιτίθενται ένα-ένα σε κάποιον ήρωα, επιλεγμένο τυχαία. Η μάχη συνεχίζεται ώσπου όλα τα τεράτα ή όλοι οι ήρωες να χάσουν όλο το health power τους. Αν οι ήρωες νικήσουν τότε εάν έχει πεθάνει κάποιος από αυτούς αναγεννιέται και όλοι παίρνουν experience και χρήματα ανάλογα με το μέσο

όρο των level των τεράτων και αν συγκετρώνουν το απαραίτητο ποσο ανεβαίνουν level. Αν νικήσουν τα τέρατα οι ήρωες χάνουν τα μισά λεφτά τους και αναγεννιούνται με μειωμένα χαρακτηριστικά. Με την επιλογή flee from battle οι ήρωες χάνουν το 30% των χρημάτων τους και ξεφεύγουν από τα τέρατα. Αν κάποιος ήρωας σκοτώσει ένα τέρας τότε παίρνει ένα ποσο experience ανάλογα με το level του τέρατος που σκότωσε. Η επίθεση ενός ηρωα μπορεί να είναι είτε με το wearon του είτε με spell, μόνο ένα spell κάθε τύπου μπορεί να γίνει στο ίδιο τέρας όσο διαρκει η επίδραση του προηγούμενου spell του ίδιου τύπου.

Η κάθε επίθεση γίνεται σε δύο στάδια πρώτα μια συνάρτηση του επιτιθέμενου υπολογίζεται το ποσό του damage που ο ήρωας ή το τέρας προκαλει με τα χαρακτηριστικά του και το όπλο του και ότι άλλη ενίσχυση του damage έχει. Μετά μεσα σε μία συνάρτηση του αμυνόμενου ελέγχεται αν η επιθεση αποφεύγεται πόση τελικά ζημιιά θα προκαλέσει ανάλογα με το defense και τέλος ελέγχεται αν ο αμυνόμενος πέθανε ώστε να γίνουν οι κατάλληλες ενέργειες.

## Παραδοχές

- ℵ Το health power μπορεί να πάρει τιμή μεγαλύτερη του max health power
- ℵ Εάν το party νικήσει μια μάχη τότε ακόμα και οι ήρωες που πέθαναν στη μάχη παίρνουν experience
- ℵ Ένας ήρωας μπορεί να έχει ως εξοπλισμό το πολύ δύο αντικείμενα, ένα όπλο που απαιτεί δυο χέρια, δυο όπλα, δυο armor, ένα όπλο και ένα armor
- ℵ Το grid είναι τετράγωνο

## Σημειώσεις

Για την μετακίνηση χρησιμοποιούνται λέξεις και όχι βελάκια για να τρέχει και σε windows, όπως επίσης και για τον ίδιο λόγο δεν χρησιμοποιήσαμε το clear για τον καθαρισμό του terminal.

the end