

## Υλοποίηση του διαγωνισμού Micromouse στον προσομοιωτή Webots

---

### Το πρόβλημα/θέμα

Το Micromouse είναι ένας διαγωνισμός όπου ένας μικρός ρομποτικός πράκτορας προσπαθεί να φτάσει στο κέντρο ενός λαβυρίνθου μεγέθους 16x16 κελιών. Ο διαγωνισμός είναι αρκετά δημοφιλής στον χώρο της «χομπιίστικης» ρομποτικής και γίνεται από τα τέλη της δεκαετίας του 1970 σε διάφορα πανεπιστήμια ή συλλόγους σχετικούς με το αντικείμενο της ρομποτικής. Η εργασία αυτή είναι μια προσπάθεια δημιουργίας εικονικής αναπαράστασης του διαγωνισμού στο περιβάλλον του Webots, που είναι λογισμικό προσομοίωσης ρομποτικών πρακτόρων.

Η υλοποίηση αυτή περιορίζεται στην δημιουργία ενός αυτόνομου πράκτορα που προσπαθεί να φτάσει στο κέντρο του λαβυρίνθου, κυρίως λόγω των τεχνικών δυσκολιών που αντιμετωπίστηκαν και του περιορισμένου χρόνου ενασχόλησης με το πρόβλημα. Μετά από χρόνια εξέλιξης του παιχνιδιού και των παικτών που συμμετέχουν τα αποτελέσματα είναι εκθαμβωτικά με τα ποντίκια που συμμετέχουν να αναπτύσσουν αρκετά μεγάλες ταχύτητες για το μέγεθός τους, να χρησιμοποιούν πολύ έξυπνες στρατηγικές και να κάνουν εξαιρετικά μικρούς χρόνους και οι νικητές να κρίνονται σε κλάσματα του δευτερολέπτου. Εδώ και αρκετά χρόνια τώρα υπάρχει κορεσμός στους χρόνους τερματισμού και γι' αυτό τα τελευταία χρόνια δημιουργήθηκε μια παραλλαγή του διαγωνισμού όπου το μέγεθος του λαβυρίνθου είναι 32x32 κελιά τα οποία με την σειρά τους έχουν το μισό μέγεθος από αυτά του αρχικού διαγωνισμού. Αυτή η υλοποίηση είναι μια μικρή προσπάθεια και σε καμία περίπτωση δεν συγκρίνεται με τα «ποντίκια» που λαμβάνουν μέρος σε διαγωνισμούς.

### ΚΑΝΟΝΙΣΜΟΙ ΔΙΑΓΩΝΙΣΜΟΥ

Μερικοί διαγωνισμοί διαφέρουν μεταξύ τους στην διαδικασία της βαθμολόγησης αλλά κατά βάση όλοι έχουν ίδιους κανόνες για τα ποντίκια και το λαβύρινθο (Παράδειγμα: Κανονισμοί IEEE 2018 [λήψη](#))

Αυτή η υλοποίηση εφόσον δεν ασχολείται με το κομμάτι της απόδοσης, δηλαδή την εκτέλεση προσπαθειών για τον μικρότερο δυνατό χρόνο, δεν περιέχει κάποια διαδικασία βαθμολόγησης ούτε χρονομέτρησης.

# Προσέγγιση/λύση

Η υλοποίηση αυτή διακρίνεται σε δυο μέρη: την δημιουργία του περιβάλλοντος και του πράκτορα και τον προγραμματισμό του ρομποτικού πράκτορα.

## ΠΕΡΙΒΑΛΛΟΝ ΚΑΙ ΡΟΜΠΟΤ

Κατά το πρώτο στάδιο της υλοποίησης πίστευα ότι θα ήταν λογικό να δημιουργήσω το περιβάλλον-λαβύρινθο και το ρομπότ-ποντίκι από την αρχή. Γρήγορα κατάλαβα ότι αυτό θα απαιτούσε πολύ περισσότερο χρόνο και εξοικείωση με το Webots. Με την βοήθεια του Δημήτρη Χατζηπαράσχη, του υπεύθυνου του εργαστηρίου, τον οποίο θα ήθελα να ευχαριστήσω για τις πολύτιμες συμβουλές του κατά την διάρκεια ενασχόλησής μου με την εργασία, βάσιμα την υλοποίησή μου στο ήδη υπάρχον demo που υπάρχει στο Webots για τον διαγωνισμό Rats Life.

Οπότε, το πρώτο μέρος της υλοποίησης αφορά στην τροποποίηση του περιβάλλοντος του Rats life που ουσιαστικά αποτελείται από έναν λαβύρινθο 10x10 και 2 ρομπότ e-ruck. Αρχικά, από το γραφικό περιβάλλον του Webots έπρεπε να προστεθούν αρκετοί τοίχοι αφού ο λαβύρινθος έπρεπε να είναι 16x16 και όχι 10x10 και στην συνέχεια να γίνουν παραμετροποιήσεις στα αρχεία που αφορούν την δημιουργία του περιβάλλοντος.

Τα βασικά βήματα που ακολουθήθηκαν είναι: η αφαίρεση των feeders και ενός από τα δυο e-ruck, η προσθήκη της δυνατότητας δημιουργίας λαβυρίνθων που έχουν χρησιμοποιηθεί σε παλαιότερους διαγωνισμούς και η αρχικοποίηση του e-ruck σε γωνία με την αριστερή του πλευρά προς άκρη του λαβυρίνθου (σύμφωνα με τους κανονισμούς). Για την δημιουργία των λαβυρίνθων δημιουργήθηκε η δυνατότητα χρήσης αρχείων από ένα Github [repository](#) αφιερωμένο στην συλλογή λαβυρίνθων από τους παλαιότερους διαγωνισμούς αλλά και μερικούς μικρούς λαβυρίνθους για δοκιμές, με πάνω από 400 αρχεία.

Ο κώδικας της εργασίας, λοιπόν, χωρίζεται σε δυο κομμάτια: τον controller για το περιβάλλον και για το ρομπότ. Η δομή του έχει ως εξής:

- contest\_manager (για τον λαβύρινθο, γραμμένο σε γλώσσα C)
  - contest\_manager.c (βασική ρουτίνα του προγράμματος)
  - maze\_generator.c (δημιουργία λαβυρίνθου)
  - maze\_builder.c (τοποθέτηση στο 3D περιβάλλον)
- Rat0.java (για το e-ruck, γραμμένο σε γλώσσα Java)

Παραπάνω περιλαμβάνονται μόνο τα βασικά αρχεία που περιέχουν την υλοποίηση, ενώ υπάρχουν – κυρίως για το πρώτο σκέλος – και άλλα αρχεία που αφορούν τα textures του κόσμου, υλοποιημένες βοηθητικές συναρτήσεις αλλά και αρχεία του Rats life που δεν αφαιρέθηκαν για ευκολία.

## Δημιουργία λαβυρίνθου

Στον *maze\_generator.c* υλοποιήθηκαν δυο βασικές συναρτήσεις:

- `void genetate_saved_maze(Maze *maze)`

Περιέχει έναν πίνακα με προκαθορισμένο λαβύρινθο (από το repository που αναφέρθηκε παραπάνω) και εκχωρεί στην βασική δομή που περιέχει την πληροφορία για τον λαβύρινθο τις θέσεις που πρέπει να τοποθετηθούν τοίχοι αλλά και την αρχική θέση και προσανατολισμό του ρομπότ.

- `void print_maze(Maze *maze)`

Βοηθητική συνάρτηση για debugging. Τυπώνει στο αρχείο `debugPrintFile.txt` το αποτέλεσμα της παραπάνω συνάρτησης. (Χρειάζεται αλλαγή του `path` για να δουλέψει)

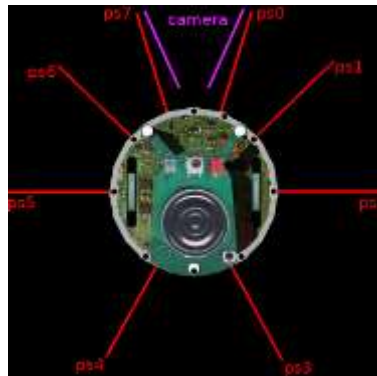
Οι παραμετροποιήσεις που έγιναν στο περιβάλλον ήταν πολύ χρονοβόρες λόγω του ότι δεν είχα εξοικειωθεί με το χειρισμό του προγράμματος Webots αλλά και του δυσανάγνωστου εγχειριδίου χρήσης του και της έλλειψης ενεργής διαδικτυακής κοινότητας όσον αφορά της δυνατότητες και την λειτουργία του.

## Προγραμματισμός του ρομπότ

Στο `Rat0.java` σε κάθε καινούριο χρονικό βήμα εκτελούνται μια σειρά από διαδικασίες που αφορούν τα εξής δομικά στοιχεία της «συμπεριφοράς» του ρομπότ:

- Οδομετρία:

Χρήση των δεδομένων από τους αισθητήρες απόστασης (8 υπέρυθρους και 2 περιστροφικούς κωδικοποιητές στα βηματικά μοτέρ) για την αναγνώριση της κίνησής του e-ruck στο χώρο. Σε αυτή την υλοποίηση, λόγω απλότητας, η ανίχνευση των κελιών γίνεται με βάση την απόσταση που διανύει το ρομπότ αλλά και τους τοίχους που το περιβάλλουν κάθε χρονική στιγμή.



*1 e-puck IR sensors*

- Ανίχνευση τοίχων:

Γίνεται με τον συνδυασμό των δεδομένων από τους υπέρυθρους αισθητήρες που βρίσκονται μπροστά και στα πλάγια του ρομπότ.

- Χαρτογράφηση και Εντοπισμός:

Εφόσον, η αρχική και η τελική θέση του ρομπότ είναι γνωστή και ο λαβύρινθος έχει συγκεκριμένες διαστάσεις και όρια, μπορούμε να εφαρμόσουμε ένα μοντέλο τοπικής ρομποτικής πλοήγησης. Η χαρτογράφηση του λαβυρίνθου και ο εντοπισμός του ρομπότ είναι αναγκαία για την επίλυση του προβλήματος.

Για την χαρτογράφηση έχουμε ως δεδομένο την αρχική θέση του ρομπότ, τις διαστάσεις των κελιών και ολόκληρου λαβυρίνθου και έτσι μπορούμε να κατασκευάσουμε μια δομή που να καταχωρείται σε κάθε κελί που πηγαίνει το ρομπότ η πληροφορία για την δομή του κελιού, δηλαδή τους τοίχους που το περιβάλλουν.

Ο εντοπισμός της θέσης είναι εφικτός αφού το ρομπότ έχει συγκεκριμένη κατεύθυνση στην αρχική του θέση και μέσω της καταχώρησης των βημάτων αλλά και των στροφών που έχει κάνει μπορεί σε κάθε χρονική στιγμή να «γνωρίζει» την θέση και την κατεύθυνση του στο λαβύρινθο.

- Αλγόριθμος Flood Fill:

Ένας αρκετά διαδεδομένος τρόπος -στην κοινότητα του Micromouse- για την εύρεση του κέντρου είναι η χρήση μιας παραλλαγής του αλγορίθμου Bellman-Ford, που ονομάζεται Flood Fill. Κάθε κελί έχει βάρος όσο και η απόσταση του από το κέντρο (όχι διαγώνια). Κάθε φορά που το ποντίκι βρίσκεται σε ένα νέο κελί και εντοπίζει τους τοίχους που το περιβάλλουν ανανεώνει την δομή που περιέχει τα βάρη των κελιών. Αν υπάρχει τοίχος ανάμεσα σε δυο κελιά το ρομπότ πρέπει να επιλέξει την διαδρομή που τον περικυκλώνει. Το ίδιο κάνει και αλγόριθμος Flood Fill. Περιοχές που δεν ενώνονται με κάποιο τρόπο με το κέντρο παίρνουν την τιμή -1.

Παράδειγμα:

```

0---0---0---0---0---0---0---0---0---0---0---0---0---0---0---0
|014 013 012 011 010 009 008 007 007 008 009 010 011 012 013 014|
|013 012 011 010 009 008 007 006 006 007 008 009 010 011 012 013|
|012 011 010 009 008 007 006 005 005 006 007 008 009 010 011 012|
|011 010 009 008 007 006 005 004 004 005 006 007 008 009 010 011|
|010 009 008 007 006 005 004 003 003 004 005 006 007 008 009 010|
|009 008 007 006 005 004 003 002 002 003 004 005 006 007 008 009|
|008 007 006 005 004 003 002 001 001 002 003 004 005 006 007 008|
|007 006 005 004 003 002 001 000 000 001 002 003 004 005 006 007|
|007 006 005 004 003 002 001 000 000 001 002 003 004 005 006 007|
|008 007 006 005 004 003 002 001 001 002 003 004 005 006 007 008|
|009 008 007 006 005 004 003 002 002 003 004 005 006 007 008 009|
| 01 -01 -01 -01 -01 -01 -01 -01 -01 -01|005 006 007 008 009 010|
| -01 -01 -01 -01 -01 -01 -01 -01 -01 -01|006 007 008 009 010 011|
| 01 -01 -01 -01 -01 -01 -01 -01 -01 -01|007 008 009 010 011 012|
| -01 -01 -01 -01 -01 -01 -01 -01 -01 -01|008 009 010 011 012 013|
| 01 -01 -01 -01 -01 -01 -01 -01 -01 -01|009 010 011 012 013 014|
0---0---0---0---0---0---0---0---0---0---0---0---0---0---0---0

```

1

```

0---0---0---0---0---0---0---0---0---0---0---0---0---0---0---0
|014 013 012 011 010 009 008 007 007 008 009 010 011 012 013 014|
|013 012 011 010 009 008 007 006 006 007 008 009 010 011 012 013|
|012 011 010 009 008 007 006 005 005 006 007 008 009 010 011 012|
|011 010 009 008 007 006 005 004 004 005 006 007 008 009 010 011|
|010 009 008 007 006 005 004 003 003 004 005 006 007 008 009 010|
|009 008 007 006 005 004 003 002 002 003 004 005 006 007 008 009|
|008 007 006 005 004 003 002 001 001 002 003 004 005 006 007 008|
|007 006 005 004 003 002 001 000 000 001 002 003 004 005 006 007|
|007 006 005 004 003 002 001 000 000 001 002 003 004 005 006 007|
|008 007 006 005 004 003 002 001 001 002 003 004 005 006 007 008|
|009 008 007 006 005 004 003 002 002 003 004 005 006 007 008 009|
|025 024|021 020 021 022 021 020 021 020|005 006 007 008 009 010|
|026|023 022|019 018 017 018 019 020 019|006 007 008 009 010 011|
|027|020 019 018 017 016|015 016 017 018|007 008 009 010 011 012|
|028|019 018 017 016 015|014 013 012 011|008 009 010 011 012 013|
|029|018 017 016 015 014 013 012 011 010 009 010 011 012 013 014|
0---0---0---0---0---0---0---0---0---0---0---0---0---0---0---0

```

2

1: Αποκλεισμένο από το κέντρο κομμάτι με βάρη -1 για τα κελιά που δεν συνδέονται με κάποιον τρόπο στο κέντρο

2: Ένα βήμα πριν αποκλειστεί το κάτω αριστερό κομμάτι από το κέντρο

Σε κάθε διαγωνισμό υπάρχουν κανονισμοί σχετικά με πόσες φορές ή/και πόσο χρόνο μπορεί να αφιερώσει κάθε διαγωνιζόμενος στην χαρτογράφηση του λαβυρίνθου. Μια προσπάθειας εύρεσης μονοπατιού προς το κέντρο μπορεί να χαρτογραφήσει ένα μικρό ή μεγάλο κομμάτι του λαβυρίνθου. Ένα παράδειγμα μιας προσπάθειας ακολουθεί παρακάτω:

```

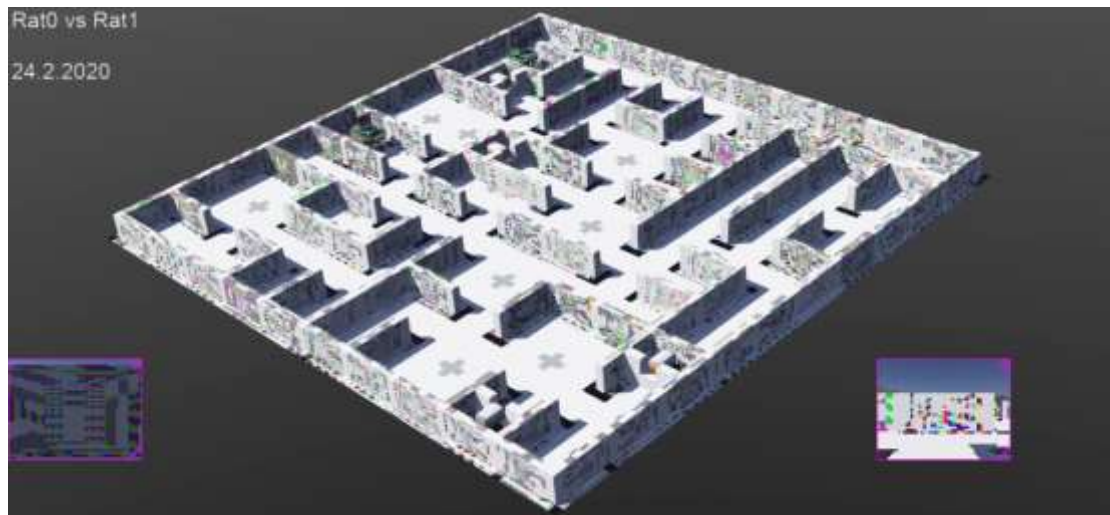
0---0---0---0---0---0---0---0---0---0---0---0---0---0---0---0
|031 030 029 028 029 030 031 030 029 028 027 026 025 024 023 024|
|030 029 028 027 028 029 030 031 030|029 030 031 032 033|022 023|
|029 028 027 026 027 028 029 030 031|028|035 034 033 032|021 022|
|030|029 028|025 024 023 024 025 026 027 028 029 030 031|020 021|
|029|030|027 026|029|022 021 020 019 018 017 016 017 018 019 020|
|028|029 028|027 028|021 020 019 018 017 016 015 016 017 018 019|
|027 026 027|028|029|022|019 020|001 002|015 014 015 016 017 018|
|028|025 024|029|030|021|018|000 000|003 004|013 014 015 016 017|
|029|024 023|030|031|020|017|000 000|006 005|012 013 014 015 016|
|028|025|022|031 032|019|016 015 014|007 008|011 012 013 014 015|
|027|026|021 020 019 018 017 016|013 012|009 010 011 012 013 014|
|026|025|022 021 020 019 018|015 014|011 010|011 012 013 014 015|
|025 024|021 020 019 018 017 016|013 012 011 012 013 014 015 016|
|024 023|020 019 018 017 016 015 014 013 014 013 014 015 016 017|
|025|022 021 020 019 018 017 016 015 014 015 014 015 016 017 018|
|026|023 022 021 020 019 018 017 016 015 016 015 016 017 018 019|
0---0---0---0---0---0---0---0---0---0---0---0---0---0---0---0

```

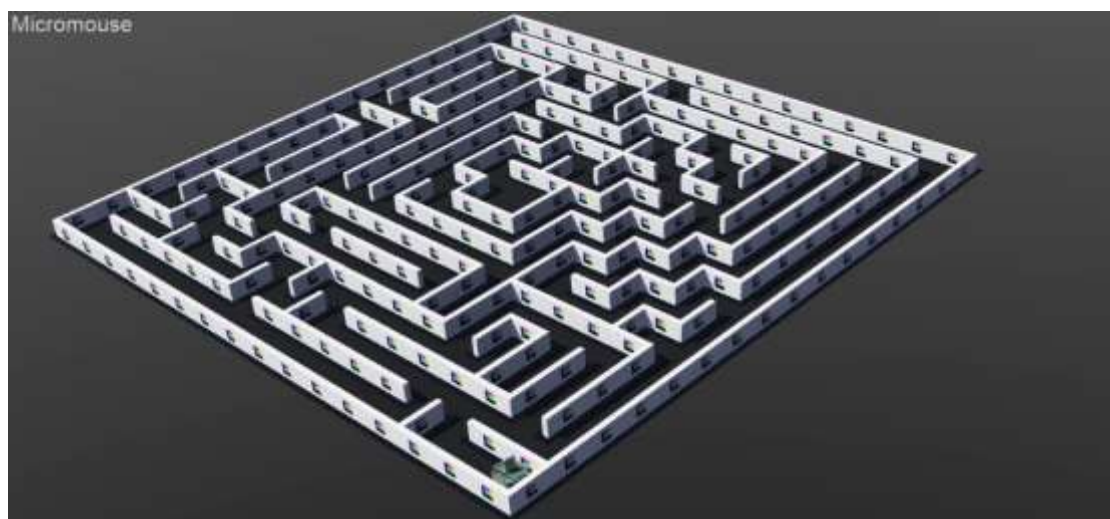


## Αποτελέσματα/επίδειξη

Το περιβάλλον του Rats Life που -όπως αναφέρθηκε- χρησιμοποιήθηκε ως βάση για την υλοποίηση της εργασίας ήταν το παρακάτω:



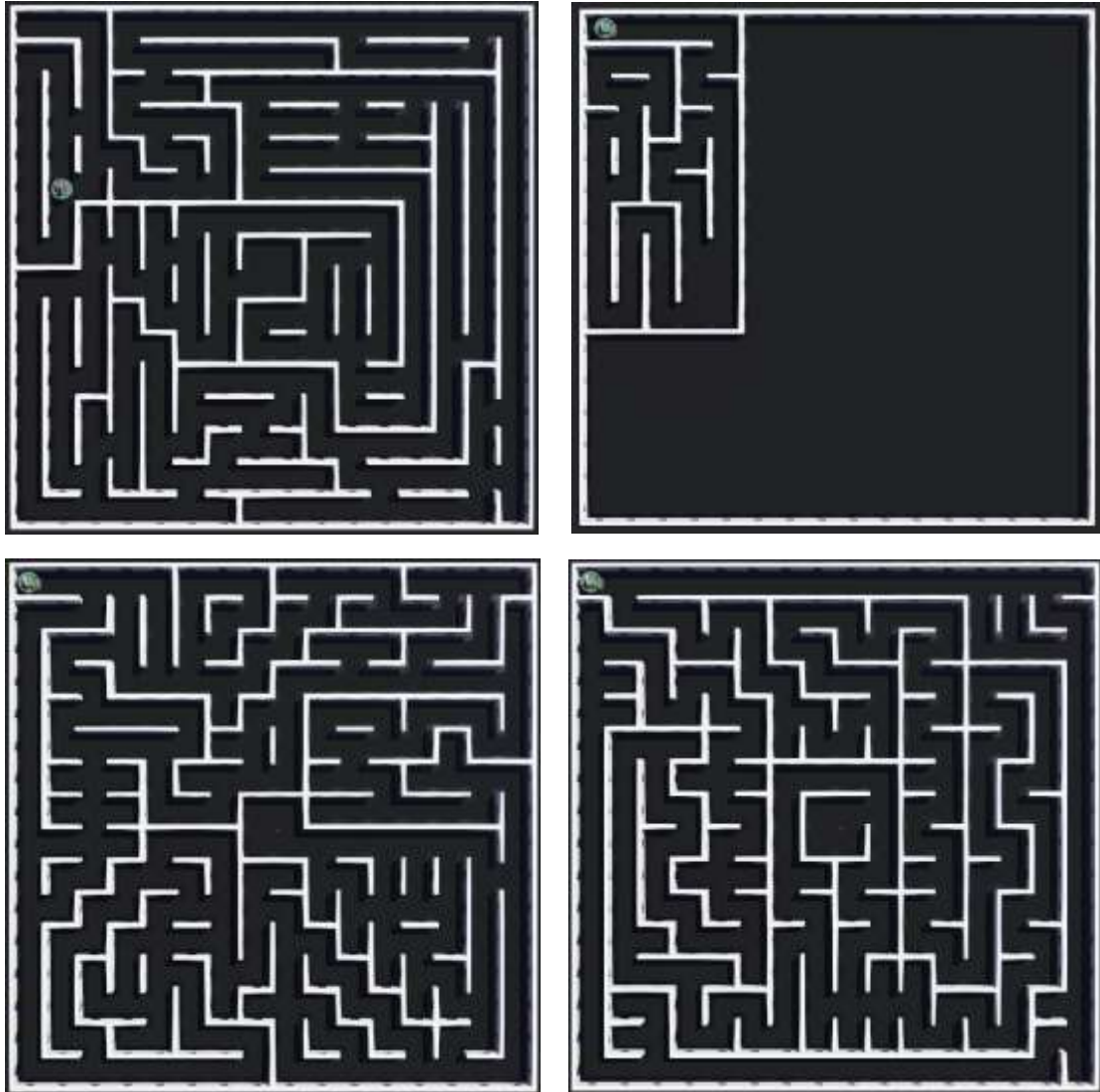
Το περιβάλλον του Micromouse τελικά:



Το ρομπότ e-ruck που χρησιμοποιήθηκε ως Micromouse:



Μερικά παραδείγματα από τους χάρτες που δημιουργήθηκαν με βάση το [repository](#) που περιέχει χάρτες από παλαιότερους διαγωνισμούς αλλά και χάρτες για δοκιμές:



Τα αποτελέσματα της συνάρτησης *print\_maze* που τυπώνει την δομή που περιέχει την πληροφορία για τον λαβύρινθο και τα βάρη των κελιών όπως ορίζονται στην αρχή και στο τέλος μιας εκτέλεσης του αλγορίθμου Flood Fill:

```

Console
[Rat0] Counter: 1
[Rat0] WALL: RIGHT
[Rat0] WALL: LEFT
[Rat0] going straight
[Rat0] o---o---o---o---o---o---o---o---o---o---o---o---o---o---o
[Rat0] |014 013 012 011 010 009 008 007 007 008 009 010 011 012 013 014|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |013 012 011 010 009 008 007 006 006 007 008 009 010 011 012 013|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |012 011 010 009 008 007 006 005 005 006 007 008 009 010 011 012|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |011 010 009 008 007 006 005 004 004 005 006 007 008 009 010 011|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |010 009 008 007 006 005 004 003 003 004 005 006 007 008 009 010|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |009 008 007 006 005 004 003 002 002 003 004 005 006 007 008 009|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |008 007 006 005 004 003 002 001 001 002 003 004 005 006 007 008|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |007 006 005 004 003 002 001 000 000 001 002 003 004 005 006 007|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |007 006 005 004 003 002 001 000 000 001 002 003 004 005 006 007|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |008 007 006 005 004 003 002 001 001 002 003 004 005 006 007 008|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |009 008 007 006 005 004 003 002 002 003 004 005 006 007 008 009|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |010 009 008 007 006 005 004 003 003 004 005 006 007 008 009 010|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |011 010 009 008 007 006 005 004 004 005 006 007 008 009 010 011|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |012 011 010 009 008 007 006 005 005 006 007 008 009 010 011 012|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |013|012 011 010 009 008 007 006 006 007 008 009 010 011 012 013|
[Rat0] o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o  o
[Rat0] |014|013 012 011 010 009 008 007 007 008 009 010 011 012 013 014|
[Rat0] o---o---o---o---o---o---o---o---o---o---o---o---o---o---o
[Rat0] POSITION: [0,1] Orientation: 0

```

*Το Micromouse στο κελί [0,1] κατά το πρώτο βήμα*



