

# Robot Labyrinth

## Τεκμηρίωση συναρτήσεων

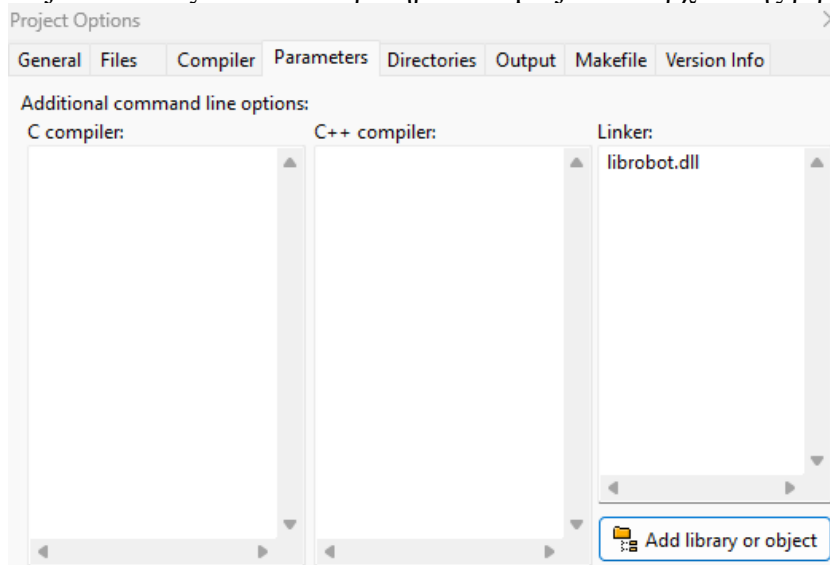
### Συναρτήσεις για επικοινωνία με τον server RobotLabyrinth

Για την αξιοποίηση των συναρτήσεων της κατηγορίας αυτής απαιτείται:

- η συμπερίληψη του αρχείου επικεφαλίδας (header file) [robot.h](#) στον πηγαίο κώδικα του προγράμματος. Η συμπερίληψη γίνεται με την εντολή `#include` του προ-επεξεργαστή (pre-processor). Το αρχείο επικεφαλίδας περιέχει τα πρωτότυπα των συναρτήσεων (function prototypes) που απαιτούνται από τον μεταγλωττιστή για να κάνει τον έλεγχο τύπων (type checking) στις προτάσεις κλήσης των αντίστοιχων συναρτήσεων.
- η αξιοποίηση, κατά τον χρόνο διασύνδεσης, της βιβλιοθήκης Robot που εμφανίζεται ως αρχείο [libRobot.dll](#).

### Αξιοποίηση της βιβλιοθήκης libRobot.dll

Τοποθετήστε την βιβλιοθήκη στο ευρετήριο που έχετε το project σας. Στο DevC++ επιλέξτε Project->Projects Options>Parameters και στην συνέχεια Add library or Object. Επιλέξτε από το ευρετήριο του project το αρχείο της βιβλιοθήκης.



Οι συναρτήσεις επικοινωνίας με τον server RobotLabyrinth διακρίνονται σε δύο επιμέρους κατηγορίες:

#### 1. Συναρτήσεις γενικού σκοπού

Οι συναρτήσεις αυτές αποστέλλουν αιτήματα γενικού σκοπού στον server RobotLabyrinth [connect2Labyrinth](#), [loadLabyrinth](#), [insertRobot](#), [destroyRobot](#), [disconnect](#)

#### 2. Συναρτήσεις που αφορούν το ρομπότ

Οι συναρτήσεις αυτές αποστέλλουν στον server RobotLabyrinth αιτήματα που αφορούν το Robot.

[robotGoAhead](#), [robotStep](#), [robotTurnLeft](#),  
[robotTurnRight](#), [robotStop](#), [robotGoHome](#),  
[robotGetState](#), [robotExitFound](#)

## 1) Γενικού σκοπού συναρτήσεις

**SOCKET connect2Labyrinth(const char \*ip, int port);**

Αποκαθιστά σύνδεση με το πρόγραμμα RobotLabyrinth το οποίο θα πρέπει να έχει ενεργοποιηθεί (run) και να έχει οριστεί να λειτουργεί σε Remote mode. Πατώντας το πλήκτρο START το πρόγραμμα εμφανίζει την IP διεύθυνση του υπολογιστή στον οποίο τρέχει.

Παράμετροι:

*server\_addr* : Η IP διεύθυνση (π.χ. "150.140.184.167") του υπολογιστή στον οποίο τρέχει το πρόγραμμα RobotLabyrinth. Την διεύθυνση αυτή σας την δίνει το πρόγραμμα RobotLabyrinth όταν το βάλετε σε mode remote και πατήσετε START.

*server\_port* : Ο αριθμός της πόρτας στην οποία «ακούει» ο server (εξ' υποθέσεως τιμή 12345).

Επιστρεφόμενη τιμή:

Ένα socket file descriptor : αν η διαδικασία της σύνδεσης ολοκληρωθεί επιτυχώς. Αποδώστε την επιστρεφόμενη τιμή σε μεταβλητή τύπου SOCKET.  
INVALID\_SOCKET: αν παρουσιασθεί κάποιο σφάλμα.

### Παράδειγμα

Για πραγματοποίηση σύνδεσης με το πρόγραμμα RobotLabyrinth που έχει ενεργοποιηθεί σε υπολογιστή με IP 150.140.184.167 και θύρα (port) 12345

```
SOCKET sockfd = connect2Labyrinth("150.140.184.167",  
12345);
```

---

**int loadLabyrinth(SOCKET sock, char \*fname);**

Αποστέλλει στον server αίτημα (request) να φορτώσει τον λαβύρινθο που περιγράφεται στο αρχείο με όνομα fname. Για την ορθή εκτέλεση της πρέπει να έχει αποκατασταθεί σύνδεση με το πρόγραμμα RobotLabyrinth και να μην υπάρχει ρομπότ μέσα στον λαβύρινθο.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()  
*fname* : Το όνομα του αρχείου στο οποίο είναι αποθηκευμένος ο λαβύρινθος.

Επιστρεφόμενη τιμή:

0: αν η διαδικασία ολοκληρωθεί επιτυχώς  
-1: αν παρουσιασθεί κάποιο σφάλμα.

### Παράδειγμα

Για να φορτωθεί ο λαβύρινθος που περιγράφεται στο αρχείο labyrinth6.lab  
`labLoadLabyrinth(socketfd, "labyrinth6.lab");`

---

**int insertRobot(SOCKET sock, char \*name);**

Αποστέλλει στον server αίτημα (request) να εισάγει ένα νέο ρομπότ στο λαβύρινθο. Απαραίτητη προϋπόθεση να έχει αποκατασταθεί η σύνδεση με το RobotLabyrinth. Μπορείτε να εισάγεται μόνο ένα ρομπότ.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()  
*name* : Το όνομα του ρομπότ.

Επιστρεφόμενη τιμή:

0: αν η διαδικασία ολοκληρωθεί επιτυχώς  
-1: αν παρουσιασθεί κάποιο σφάλμα.

### Παράδειγμα

`labInsertRobot(socketfd, "cleo");`

---

**int destroyRobot(SOCKET sock);**

Αποστέλλει στον server αίτημα (request) να καταστρέψει το ρομπότ που υπάρχει στον λαβύρινθο του server που αντιστοιχεί στο **sock**.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

Επιστρεφόμενη τιμή:

0: αν η διαδικασία ολοκληρωθεί επιτυχώς  
-1: αν παρουσιασθεί κάποιο σφάλμα.

---

**int disconnect(SOCKET sock) ;**

Διακόπτει την σύνδεση με τον server RobotLabyrinth που αντιστοιχεί στο **sock**..

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

Επιστρεφόμενη τιμή:

0: αν η διαδικασία ολοκληρωθεί επιτυχώς  
-1: αν παρουσιασθεί κάποιο σφάλμα.

---

## 2) Συναρτήσεις για επικοινωνία με το Robot

Για την χρησιμοποίηση των συναρτήσεων αυτής της κατηγορίας πρέπει προηγουμένως να έχει αποκατασταθεί η σύνδεση με το πρόγραμμα RobotLabyrinth και να έχει εισαχθεί επιτυχώς ένα ρομπότ στον λαβύρινθο.

### **int robotSetMode(SOCKET sock, int mode);**

Αποστέλλει στον server αίτημα (request) να θέσει την κατάσταση (mode) του ρομπότ είτε σε AUTO (1) είτε σε MANUAL (0). Πρέπει να έχει αποκατασταθεί σύνδεση και να έχει εισαχθεί επιτυχώς ένα ρομπότ.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

*mode* : Το mode λειτουργίας του ρομπότ (τιμή 0 ή 1)

Επιστρεφόμενη τιμή:

0: αν η διαδικασία ολοκληρωθεί επιτυχώς

-1: αν παρουσιασθεί κάποιο σφάλμα.

---

### **int robotGoAhead(SOCKET sock);**

Αποστέλλει στον server αίτημα (request) να δώσει εντολή στο ρομπότ να αρχίσει να κινείται. Το ρομπότ κινείται μόνο προς τα εμπρός εκτός αν πάρει εντολή να στρίψει. Η κίνηση του σταματά μόνο όταν πάρει εντολή stop.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

Επιστρεφόμενη τιμή:

0: αν η διαδικασία ολοκληρωθεί επιτυχώς

-1: αν παρουσιασθεί κάποιο σφάλμα.

---

### **int robotStep(SOCKET sock, int numberOfSteps);**

Αποστέλλει στον server αίτημα (request) να δώσει εντολή στο ρομπότ να κάνει συγκεκριμένο αριθμό βημάτων και να σταματήσει. Κάθε τετράγωνο του λαβυρίνθου αντιστοιχεί σε 8 βήματα. Η συνάρτηση επιστέφει μόνο αφού το ρομπότ ολοκληρώσει όλα τα βήματα.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

*numberOfSteps* : Ο αριθμός των βημάτων που το ρομπότ καλείται να εκτελέσει. Η τιμή πρέπει να είναι θετική και μικρότερη ή ίση του 8.

Επιστρεφόμενη τιμή:

0: αν η διαδικασία ολοκληρωθεί επιτυχώς

-1: αν παρουσιασθεί κάποιο σφάλμα.

---

**int robotTurnLeft(SOCKET sock) ;**

Αποστέλλει στον server αίτημα (request) να δώσει εντολή στο ρομπότ να στρίψει αριστερά.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

Επιστρεφόμενη τιμή:

0: αν η διαδικασία ολοκληρωθεί επιτυχώς  
-1: αν παρουσιασθεί κάποιο σφάλμα.

---

**int robotTurnRight(SOCKET sock) ;**

Αποστέλλει στον server αίτημα (request) να δώσει εντολή στο ρομπότ να στρίψει δεξιά.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

Επιστρεφόμενη τιμή:

0: αν η διαδικασία ολοκληρωθεί επιτυχώς  
-1: αν παρουσιασθεί κάποιο σφάλμα.

---

**int robotStop(SOCKET sock) ;**

Αποστέλλει στον server αίτημα (request) να δώσει εντολή στο ρομπότ να σταματήσει.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

Επιστρεφόμενη τιμή:

0 αν η διαδικασία ολοκληρωθεί επιτυχώς  
-1 αν παρουσιασθεί κάποιο σφάλμα.

---

**int robotGoHome(SOCKET sock) ;**

Αποστέλλει στον server αίτημα (request) να δώσει εντολή στο ρομπότ να πάει στην θέση εκκίνησης. Το ρομπότ παραμένει ακίνητο στη θέση αυτή.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

Επιστρεφόμενη τιμή:

0 αν η διαδικασία ολοκληρωθεί επιτυχώς  
-1 αν παρουσιασθεί κάποιο σφάλμα.

---

**int robotGetState(SOCKET sock) ;**

Αποστέλλει στον server αίτημα (request) να του αποστείλει την κατάσταση του ρομπότ.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

Επιστρεφόμενη τιμή:

STATE\_STOPED = 1 αν το ρομπότ είναι σταματημένο.

STATE\_MOVING = 2 αν το ρομπότ κινείται

-1 αν παρουσιασθεί κάποιο σφάλμα.

---

**int robotExitFound(SOCKET sock) ;**

Αποστέλλει στον server αίτημα (request) να ελέγξει αν το ρομπότ έχει βρει την έξοδο.

Παράμετροι:

*sock*: socket file descriptor που έχει αποκτηθεί με κλήση της συνάρτησης connect2Labyrinth()

Επιστρεφόμενη τιμή:

0 η έξοδος δεν έχει ακόμα βρεθεί

1 το ρομπότ έχει βρει την έξοδο

-1 αν παρουσιασθεί κάποιο σφάλμα.

---