



ΕΠΛ033 – Εισαγωγή στον Προγραμματισμό για Μηχανικούς: Εργαστηριακή Άσκηση 2

Στόχος

Η εξοικείωση των φοιτητών με τη γλώσσα προγραμματισμού C, τις συνθήκες ελέγχου, τη χρήση βρόχων, και τη χρήση συναρτήσεων.

Εκφώνηση

Η εργαστηριακή άσκηση 2 περιλαμβάνει τη δημιουργία δύο προγραμμάτων (ασκήσεων) στη γλώσσα προγραμματισμού C.

1. Το πρώτο πρόγραμμα ονομάζεται «Μάντεψε τον αριθμό» και είναι ένα παιχνίδι για δύο παίκτες.
2. Το δεύτερο πρόγραμμα ονομάζεται «Ο κοντινότερος αριθμός κερδίζει» και είναι ένα παιχνίδι για περισσότερους από δύο παίκτες.

1η Άσκηση: Μάντεψε τον Αριθμό

(45 Μονάδες)

Ζητούμενα:

1. Στόχος του προγράμματος αυτού είναι η δημιουργία ενός παιχνιδιού για δύο παίκτες. Ο 1ος παίκτης καλείται να δώσει ένα αριθμό τον οποίο θα πρέπει να μαντέψει ο δεύτερος παίκτης στη συνέχεια.
2. Πιο συγκεκριμένα το πρόγραμμα θα ξεκινάει με ένα μενού δύο επιλογών. Η πρώτη επιλογή θα είναι η Έναρξη παιχνιδιού (Play Game) και η δεύτερη ο τερματισμός του (Exit Game).
3. Όταν ο χρήστης επιλέξει την Έναρξη του παιχνιδιού τότε ο πρώτος παίκτης καλείται να βάλει έναν ακέραιο αριθμό μεταξύ 0-100. Σε περίπτωση που ο αριθμός δεν είναι μέσα στα επιτρεπτά όρια το πρόγραμμα θα πρέπει να εμφανίζει σχετικό μήνυμα λάθους και να ξαναζητά αριθμό από τον πρώτο παίκτη μέχρι να δώσει αριθμό εντός ορίων.
4. Αν ο αριθμός είναι μέσα στα επιτρεπτά όρια τότε το πρόγραμμα θα ζητάει από τον δεύτερο παίκτη να δώσει τον πρώτο του αριθμό προσπαθώντας να μαντέψει τον αριθμό που εισήγαγε ο πρώτος παίκτης. Ο δεύτερος παίκτης έχει απεριόριστο αριθμό προσπαθειών. Σε περίπτωση όμως που θελήσει να εγκαταλείψει (δηλώσει ήττα) θα μπορεί ανά πάσα στιγμή να το κάνει εισάγοντας οποιονδήποτε αρνητικό αριθμό.
5. Όταν ο δεύτερος παίκτης βρει τον αριθμό ή εγκαταλείψει την προσπάθεια του, τότε το πρόγραμμα θα ανακοινώνει τον νικητή (1ος παίκτης ή 2ος παίκτης) και θα επανέρχεται στο αρχικό μενού.



Για τη δημιουργία του παιχνιδιού πρέπει να υλοποιήσετε τις εξής συναρτήσεις:

- void **main**(void);
- void **print_menu**();
- int **get_valid_int_input**(int orio1, int orio2);
- void **play_game**();
- int **player1_get_number**();
- int **player2_guess**(int secret);

Συνάρτηση **void main**():

Η συνάρτηση **main**() είναι συγκεκριμένη και πρέπει να χρησιμοποιήσετε αυτήν που σας δίνεται με όνομα **game.c**. Εσείς απλά θα υλοποιήσετε τις επιμέρους συναρτήσεις μετά την **main**() που χρησιμοποιεί η **main**() για να επιτύχετε το σκοπό της άσκησης.

Συνάρτηση **void print_menu**():

Η συνάρτηση αυτή απλά θα εμφανίζει τις επιλογές του κεντρικού μενού αριθμημένες.

Συνάρτηση **int get_valid_int_input**(int orio1, int orio2):

Στη συνάρτηση αυτή θα πρέπει να ζητάτε από το χρήστη να σας δώσει έναν ακέραιο μεταξύ του **orio1** και **orio2**. Σε περίπτωση επιλογής εκτός των ορίων, θα εμφανίζει σχετικό μήνυμα λάθους και θα ζητά ξανά την είσοδο του χρήστη. Όταν ο χρήστης δώσει έγκυρη είσοδο τότε η συνάρτηση θα επιστρέφει τον αριθμό που έδωσε ο χρήστης.

Συνάρτηση **void play_game**():

Η συνάρτηση αυτή υλοποιεί το παιχνίδι. Ουσιαστικά θα εμφανίζει μήνυμα χαιρετισμού για το παιχνίδι και θα καλεί τη συνάρτηση **player1_get_number**() η οποία επιστρέφει τον μυστικό αριθμό. Έπειτα η **play_game**() θα δίνει αυτόν τον αριθμό στη συνάρτηση **player2_guess**(), η οποία θα αναλαμβάνει τη σειρά του 2^{ου} παίκτη και θα τυπώνει τον νικητή.

Συνάρτηση **int player1_get_number**():

Η συνάρτηση αυτή θα εμφανίζει σχετικό μήνυμα και θα ζητά από τον 1^ο παίκτη ένα αριθμό μεταξύ 0-100. Θα χρησιμοποιεί τη συνάρτηση **get_valid_int_input**() με τα κατάλληλα όρια για αυτόν τον σκοπό. Τέλος, τον αριθμό που θα επιστρέψει η **get_valid_int_input**() θα τον επιστρέφει και η **player1_get_number**().

Συνάρτηση **void player2_guess**(int secret):

Η συνάρτηση αυτή θα εμφανίζει σχετικό μήνυμα στον 2^ο παίκτη και θα ζητά έναν αριθμό. Η συνάρτηση θα πληροφορεί τον 2^ο παίκτη εάν ο αριθμός που έδωσε είναι μικρότερος ή μεγαλύτερος από τον **secret** και θα ξαναζητάει νέο αριθμό από τον 2^ο παίκτη. Σε περίπτωση που αριθμός εισόδου είναι ο ίδιος με τον αριθμό **secret** τότε η συνάρτηση θα τυπώνει το μήνυμα νίκης για τον 2^ο παίκτη. Αν ο χρήστης δώσει αρνητικό αριθμό τότε η συνάρτηση θα τυπώνει μήνυμα ήττας/παράδοσης του 2^{ου} παίκτη.



Παράδειγμα εκτέλεσης:

(με κόκκινο η είσοδος του χρήστη):

Game: Guess the Number

=====

Main Menu:

1. Play Game
2. Exit Game

Choose a number between 1 and 2: **0**

Wrong Choice!

Choose a number between 1 and 2: **1**

Game Starts!

=====

Player 1 Give a secret number.

Give a number between 0 and 100: **8**

Player 2 Turn:

Hint: The number is between 0 and 100

Make a guess: **90**

The secret number is smaller! Try again.

Make a guess: **20**

The secret number is smaller! Try again.

Make a guess: **2**

The secret number is bigger! Try again.

Make a guess: **-1**

Player 1 Wins! (Player 2 gave up...)

Game: Guess the Number

=====

Main Menu:

1. Play Game
2. Exit Game

Choose a number between 1 and 2: **1**

Game Starts!

=====

Player 1 Give a secret number.

Choose a number between 0 and 100: **800**

Wrong Choice!

Choose a number between 0 and 100: **50**



Player 2 Turn:

Hint: The number is between 0 and 100

Make a guess: **5**

The secret number is bigger! Try again.

Make a guess: **50**

Player 2 Wins! He/She found the secret number!

Main Menu:

1. Play Game

2. Exit Game

Choose a number between 1 and 2: **2**

Thank you for playing! Bye!



2^η Άσκηση: Ο κοντινότερος αριθμός κερδίζει

(50 Μονάδες)

Ζητούμενα:

1. Στόχος του προγράμματος αυτού είναι η δημιουργία ενός παιχνιδιού για πολλαπλούς παίκτες όπου ο παίκτης «μάννα» καλείτε να δώσει έναν αριθμό τον οποίο θα πρέπει να μαντέψουν οι επόμενοι παίκτες στη συνέχεια.
2. Κάθε παίκτης έχει μία ευκαιρία και ο νικητής είναι αυτός που μάντεψε τον μυστικό αριθμό ή αυτός ο οποίος έχει δώσει τον πιο κοντινό αριθμό στον μυστικό αριθμό. Με τον τερματισμό του κάθε γύρου του παιχνιδιού θα εμφανίζεται ο νικητής στην οθόνη. Για τον τερματισμό ενός γύρου θα πρέπει ο παίκτης ο οποίος έχει σειρά να εισάξει τον αριθμο -1.
3. Πιο συγκεκριμένα το πρόγραμμα θα ξεκινάει με ένα μενού δύο επιλογών. Η πρώτη επιλογή θα είναι η Έναρξη παιχνιδιού (Play Game) και η δεύτερη ο τερματισμός του (Exit Game).
4. Όταν ο χρήστης επιλέξει την Έναρξη του παιχνιδιού τότε ο παίκτης «μάννα» καλείτε να βάλει έναν ακέραιο αριθμό μεταξύ 0-100. Σε περίπτωση που ο αριθμός δεν είναι μέσα στα επιτρεπτά όρια το πρόγραμμα θα πρέπει να εμφανίζει σχετικό μήνυμα λάθους και να συνεχίζει να ζητά αριθμό από τον παίκτη «μάννα» μέχρι να δώσει ένα αριθμό ο οποίος είναι εντός ορίων. Εφ' όσον το κάνει, σειρά παίρνουν οι επόμενοι παίκτες.
5. Οι παίκτες που ακολουθούν θα δίνουν αριθμούς ένας-ένας με τη σειρά, προσπαθώντας να μαντέψουν το μυστικό αριθμό. Σε περίπτωση που ο αριθμός δεν είναι μέσα στα επιτρεπτά όρια το πρόγραμμα θα πρέπει να εμφανίζει σχετικό μήνυμα λάθους και να συνεχίζει να ζητά αριθμό από τον παίκτη μέχρι να δοθεί ένας αριθμός ο οποίος είναι εντός ορίων. Για να τερματίσει το πρόγραμμα ένας παίκτης θα πρέπει να εισάξει τον αριθμό -1.
6. Εάν ένας παίκτης μαντέψει το μυστικό αριθμό τότε ανακηρύσσεται αυτόματα ο νικητής του παιχνιδιού και το πρόγραμμα επανέρχεται στο αρχικό μενού. Σε αντίθετη περίπτωση, ως νικητής ορίζεται ο παίκτης που έχει δώσει τον αριθμό που είναι πιο κοντά στο μυστικό αριθμό. Σε περίπτωση **ισοπαλίας** θα εμφανίζεται ως νικητής μόνο ο τελευταίος κοντινότερος παίκτης.

Για τη δημιουργία του παιχνιδιού πρέπει να υλοποιήσετε τις εξής συναρτήσεις:

- void **main**(void);
- void **print_menu**();
- int **get_valid_int_input**(int orio1, int orio2);
- void **play_game**();
- int **player_mana_get_number**();
- void **players_guess**(int secret);
- int **who_is_closer**(int current_closest, int new_number, int secret);



Συνάρτηση **main()**:

Ο σκελετός της συνάρτησης **main()** σας δίνεται έτοιμος με όνομα **game.c** (είναι ίδια με την προηγούμενη άσκηση). Δεν μπορείτε να κάνετε αλλαγές στην **main()**.

Συνάρτηση **void print_menu()**:

Η συνάρτηση αυτή απλά θα εμφανίζει τις επιλογές του κεντρικού μενού αριθμημένες.

Συνάρτηση **int get_valid_int_input(int orio1, int orio2)**:

Στη συνάρτηση αυτή θα πρέπει να ζητάτε από το χρήστη να σας δώσει έναν ακέραιο μεταξύ του **orio1** και **orio2**. Σε περίπτωση επιλογής εκτός των ορίων, θα εμφανίζει σχετικό μήνυμα λάθους και θα ζητά ξανά την είσοδο του χρήστη. Όταν ο χρήστης δώσει έγκυρη είσοδο τότε η συνάρτηση θα επιστρέφει τον αριθμό που έδωσε ο χρήστης.

Συνάρτηση **void play_game()**:

Η συνάρτηση αυτή υλοποιεί το παιχνίδι. Ουσιαστικά θα εμφανίζει μήνυμα χαιρετισμού για το παιχνίδι και θα καλεί τη συνάρτηση **player_mana_get_number()**, η οποία επιστρέφει τον μυστικό αριθμό. Έπειτα η **play_game()** θα δίνει αυτόν τον αριθμό στη συνάρτηση **players_guess()**, η οποία θα αναλαμβάνει τη σειρά των υπόλοιπων παικτών και αναλαμβάνει η ίδια την εκτύπωση του νικητή. Η συνάρτηση αυτή δεν επιστρέφει τίποτε.

Συνάρτηση **int player_mana_get_number()**:

Η συνάρτηση αυτή θα εμφανίζει σχετικό μήνυμα και θα ζητά από τον 1^ο παίκτη ένα αριθμό μεταξύ 0-100. Θα χρησιμοποιεί τη συνάρτηση **get_valid_int_input()** με τα κατάλληλα όρια για αυτόν τον σκοπό. Τέλος τον αριθμό που θα επιστρέψει η **get_valid_int_input()** θα τον επιστρέφει και η **player1_get_number()**.

Συνάρτηση **void players_guess(int secret)**:

Η συνάρτηση αυτή θα εμφανίζει σχετικό μήνυμα στον επόμενο παίκτη και θα του ζητά να εισάξει έναν αριθμό από το 0 μέχρι το 100. Αυτό θα επαναλαμβάνεται μέχρι να βρεθεί ο μυστικός αριθμός ή ένας παίκτης εισάξει τον αριθμό εξόδου (-1). Θα πρέπει να πραγματοποιείται ο αντίστοιχος έλεγχος ότι ο αριθμός που έχει δοθεί είναι μέσα στα επιτρεπτά όρια. Κάθε φορά που κάποιος παίκτης δίνει έναν αριθμό η συνάρτηση θα χρησιμοποιεί την συνάρτηση **who_is_closer()** για να κρατάει τον τρέχον καλύτερο αριθμό (δηλ. τον κοντινότερο στο μυστικό αριθμό) που δόθηκε. Σε περίπτωση που κάποιος παίκτης βρει το μυστικό αριθμό τότε ο γύρος ολοκληρώνεται και η συνάρτηση εκτυπώνει το ανάλογο μήνυμα, μαζί με το μυστικό αριθμό και τον παίκτη ο οποίος έχει κερδίσει. Σε περίπτωση που δεν βρεθεί ο μυστικός αριθμός, ο γύρος ολοκληρώνεται μόλις ένας παίκτης δώσει τον αριθμό εξόδου -1, και η συνάρτηση τότε τυπώνει τον αριθμό που ήταν πιο κοντά στο μυστικό αριθμό καθώς και ποιος παίκτης τον έδωσε (πχ., ο 5^{ος} παίκτης).

Συνάρτηση **int who_is_closer(int current_closest, int new_number, int secret)**:

Η συνάρτηση αυτή παίρνει σαν παραμέτρους εισόδου 3 ακέραιους αριθμούς, τον τρέχον πλησιέστερο αριθμό (που είναι γνωστός μέχρι τώρα), τον αριθμό που μόλις έδωσε ο παίκτης που είχε σειρά, και τον μυστικό αριθμό. Η συνάρτηση ελέγχει αν ο **new_number** είναι πιο κοντά στον **secret** από τον **current_closest**. Αν ναι, τότε επιστρέφει τον **new_number**, αλλιώς επιστρέφει τον **current_closest**.



Παράδειγμα εκτέλεσης:

```
=====
Main Menu:
1.Play Game
2.Exit Game

Choose a number between 1 and 2: 2

Thank you for playing! Bye!
=====

=====
Main Menu:
1.Play Game
2.Exit Game

Choose a number between 1 and 2: 1

Game Starts!
=====

Player 'Mana' Turn:
Choose a number between 0 and 100: -1

Wrong Choice!
Choose a number between 0 and 100: 111

Wrong Choice!
Choose a number between 0 and 100: 0

Player 2 Turn:
Hint: The number is between 0 and 100:
Make a guess (-1 to exit): -100

Wrong Choice!!
Player 2 Turn:
Hint: The number is between 0 and 100:
Make a guess (-1 to exit): 111

Wrong Choice!
Player 2 Turn:
Hint: The number is between 0 and 100:
Make a guess (-1 to exit): 55

Player 3 Turn:
Hint: The number is between 0 and 100
Make a guess (-1 to exit): 66

Player 4 Turn:
```



Hint: The number is between 0 and 100
Make a guess (-1 to exit): **77**

Player 5 Turn:
Hint: The number is between 0 and 100
Make a guess (-1 to exit): **0**

Player 5 Wins! He/She found the secret number: 0!!

=====

Main Menu:
1.Play Game
2.Exit Game

Choose a number between 1 and 2: **1**

Game Starts!
=====

Player 'Mana' Turn:

Choose a number between 0 and 100: **1**

Player 2 Turn:
Hint: The number is between 0 and 100:
Make a guess (-1 to exit): **1**

Player 2 Wins! He/She found the secret number: 1!!

Game: Guess the Number
=====

Main Menu:
1.Play Game
2.Exit Game

Choose a number between 1 and 2: **1**



Game Starts!

=====

Player 'Mana' Turn:

Choose a number between 0 and 100: **33**

Player 2 Turn:

Hint: The number is between 0 and 100:

Make a guess (-1 to exit): **66**

Player 3 Turn:

Hint: The number is between 0 and 100

Make a guess (-1 to exit): **55**

Player 4 Turn:

Hint: The number is between 0 and 100

Make a guess (-1 to exit): **44**

Player 5 Turn:

Hint: The number is between 0 and 100

Make a guess (-1 to exit): **22**

Player 6 Turn:

Hint: The number is between 0 and 100

Make a guess (-1 to exit): **11**

Player 7 Turn:

Hint: The number is between 0 and 100

Make a guess (-1 to exit): **99**

Player 8 Turn:

Hint: The number is between 0 and 100

Make a guess (-1 to exit): **88**

Player 9 Turn:

Hint: The number is between 0 and 100

Make a guess (-1 to exit): **0**

Player 10 Turn:

Hint: The number is between 0 and 100

Make a guess (-1 to exit): **-1**

Player 5 Wins! Closest Number was 22. Secret number was 33.

Game: Guess the Number

=====

Main Menu:

1.Play Game

2.Exit Game

Choose a number between 1 and 2: **2**

Thank you for playing! Bye!

=====



3η Άσκηση

(5 Μονάδες)

Ζητούμενα:

Να εκτελέσετε τα παραδείγματα των εκφωνήσεων και να τα αποθηκεύσετε σε ένα typescript αρχείο με την χρήση της εντολής script.

Για τη δημιουργία του typescript θα πρέπει να χρησιμοποιηθεί η εντολή script και όχι η εντολή printscreen της οθόνης.

Το typescript αρχείο σας δεν πρέπει να είναι επεξεργασμένο.

Παραδοτέα:

Τα αρχεία του πηγαίου κώδικά σας με την πιο κάτω ονομασία:

1η Άσκηση: *epl033_assignment02_01_XXXXXX.c*

2η Άσκηση: *epl033_assignment02_02_XXXXXX.c*

3η Άσκηση: *epl033_assignment02_03_XXXXXX_typescript.txt*

όπου XXXXXX θα είναι ο αριθμός της ταυτότητάς σας.

Τα αρχεία θα πρέπει να περιέχουν υπό μορφή σχολίων το ονοματεπώνυμο και τον αριθμό ταυτότητάς σας (εκτός από το αρχείο της 3ης άσκησης). Η παράδοση του πηγαίου κώδικα θα γίνει μέσω της πλατφόρμας Blackboard (<https://blackboard.ucy.ac.cy>). Σε περίπτωση που αντιμετωπίσετε οποιοδήποτε πρόβλημα με την παράδοση στο Blackboard, παρακαλώ αποστείλετε τον κώδικα σας στο ηλεκτρονικό ταχυδρομείο ntemen01@cs.ucy.ac.cy και CC το ηλεκτρονικό ταχυδρομείο σας.

Βαθμολογία:

Για να πάρει η εργασία σας όλες τις μονάδες είναι απαραίτητο ο κώδικάς σας να:

1. περιλαμβάνει στην αρχή σε σχόλια τα στοιχεία σας και τα στοιχεία του προγράμματός σας (πχ. όνομα, επίθετο, τίτλος προγράμματος, ημερομηνία, στόχος προγράμματος κτλ.).
2. περιλαμβάνει σε σχόλια τα βήματα της λύσης ένθετα με τις εντολές.
3. είναι ευθυγραμμισμένος και στοιχισμένος.
4. χρησιμοποιεί αυτό-επεξηγηματικά ονόματα για τις μεταβλητές.
5. μεταγλωττίζεται χωρίς λάθη.
6. παράγει σωστά αποτελέσματα.
7. έχει κατανοητή «επικοινωνία» με τον χρήστη.

→Ο κώδικας σας θα πρέπει να **μεταγλωττίζει και να εκτελείτε στις μηχανές του εργαστηρίου B103**. Σε αντίθετη περίπτωση η αντίστοιχη άσκηση **θα μηδενίζεται**.

→Ο κώδικας θα ελέγχεται από αυτόματο σύστημα ανίχνευσης αντιγραφών. **Περιπτώσεις αντιγραφής θα μηδενίζονται**.

→**Καθυστερημένες υποβολές θα μηδενίζονται**.

Τελική Ημερομηνία παράδοσης: Παρασκευή 01/11/2019, 23:59