



ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ ΚΑΙ ΤΗ ΡΟΜΠΟΤΙΚΗ

Μάθημα 8ο

Στο σημερινό μάθημα :

- Θα μάθουμε τι είναι μεταβλητή και ποια είναι τα είδη των μεταβλητών.
- Θα μάθουμε τι είναι η δομή επιλογής.
- Θα μάθουμε τον τρόπο χρήσης των μπλοκ που δημιουργούνται κατά τη δημιουργία μίας μεταβλητής.
- Θα κατανοήσουμε πλήρως τον προγραμματισμό μιας αριθμομηχανής.
- Θα μπορούμε να προγραμματίσουμε σωστά τις προτεινόμενες δραστηριότητες.
- Θα μπορούμε να απαντήσουμε σωστά τις ερωτήσεις του φύλλου αξιολόγησης στο τέλος του κεφαλαίου.

ι *

-

Το συγκεκριμένο project δεν απαιτεί να κατασκευασθεί κάποιο ρομπότ. Για τις ανάγκες του μαθήματος, είναι απαραίτητο μόνο το Hub του Spike Prime και οι αισθητήρες που θα χρειαστούν για τα λυμένα παραδείγματα και τις προτεινόμενες δραστηριότητες.

Μεταβλητές

- Με τον όρο μεταβλητή, εννοούμε μια θέση στη μνήμη, στην οποία έχουμε δώσει ένα όνομα. Η μεταβλητή μπορεί να αποθηκεύσει μια τιμή ή να διαβάσει την αποθηκευμένη τιμή.
- Η τιμή που αποθηκεύεται μέσα στη μεταβλητή μας παραμένει έως ότου δοθεί μια νέα τιμή, με την παλιά τιμή να χάνεται.

Τύποι δεδομένων

Οι τύποι δεδομένων που μπορούν να αποθηκευτούν μέσα σε μια μεταβλητή είναι:

- Αριθμός
- Κείμενο
- Λογική τιμή

Δομή Επιλογής

Καθημερινά, καλούμαστε να πάρουμε κάποιες αποφάσεις ανάλογα με το αν ισχύουν ή όχι κάποιες προϋποθέσεις. Έτσι και στον προγραμματισμό, για να μπορέσει το πρόγραμμα να καταλάβει ποιες εντολές θα εκτελέσει, χρησιμοποιούμε τη δομή επιλογής.

Η δομή επιλογής έχει 4 μορφές:

- Απλή Επιλογή
- Σύνηθετη Επιλογή
- Πολλαπλή Επιλογή
- Εμφωλευμένη Επιλογή

Το Spike δεν μας δίνει την επιλογή να χρησιμοποιήσουμε απευθείας την πολλαπλή επιλογή.

Η χρησιμοποίησή της γίνεται με χρήση της εμφωλευμένης επιλογής.

Απλή Επιλογή

```
If <Συνθήκη = Αληθής> then
```

```
<Εντολές>
```

Σύνηθετη Επιλογή

```
If <Συνθήκη = Αληθής> then
```

```
<Εντολές-1 >
```

```
Else
```

```
<Εντολές-2>
```

Πολλαπλή Επιλογή (Δεν γίνεται στο Spike)

If <Συνθήκη-1 = Αληθής>then

< Εντολές-1>

Else if <Συνθήκη-2 = Αληθής> then

<Εντολές-2>

...

...

Else

<Εντολές-η>

Εμφωλευμένη Επιλογή

If <Συνθήκη-1 = Αληθής> then

< Εντολές-1 >

Else

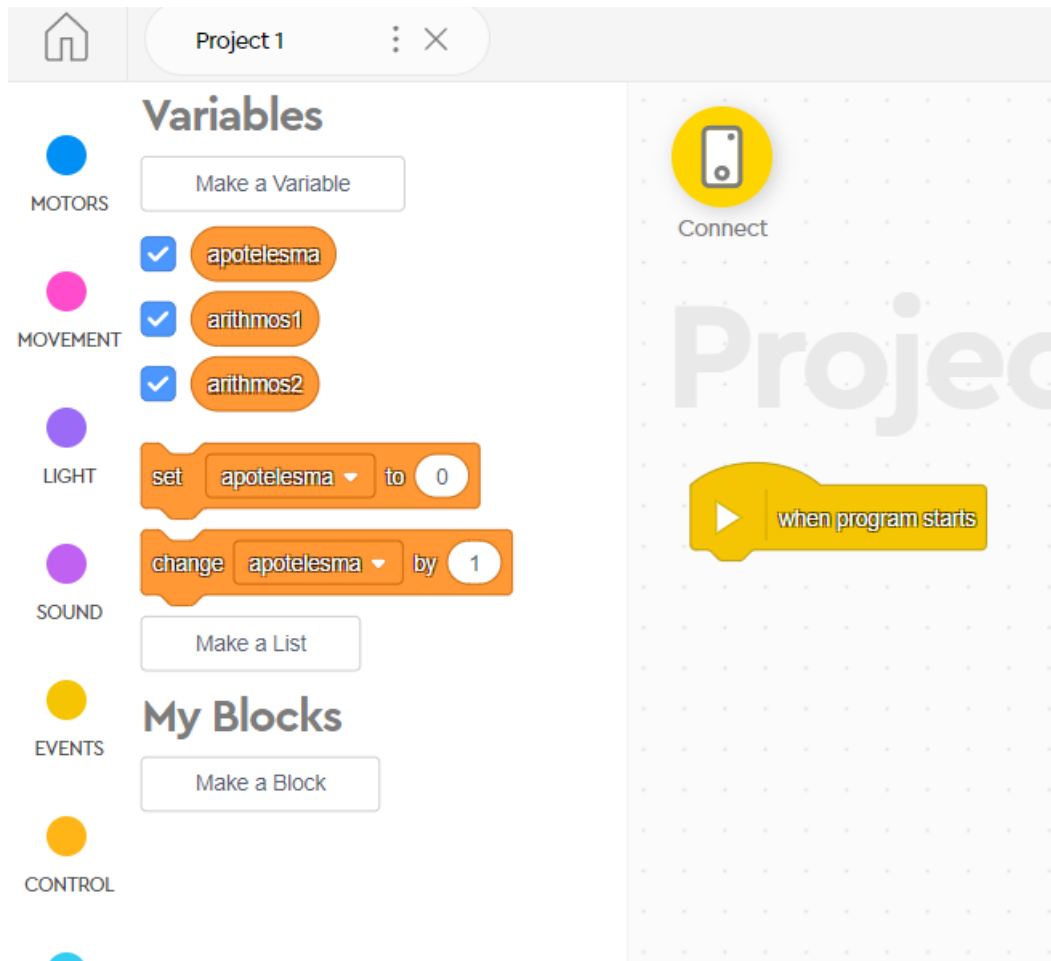
if <Συνθήκη-2 = Αληθής> then

<Εντολές-2>

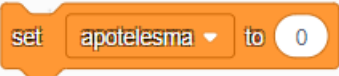
Δημιουργία αριθμομηχανής

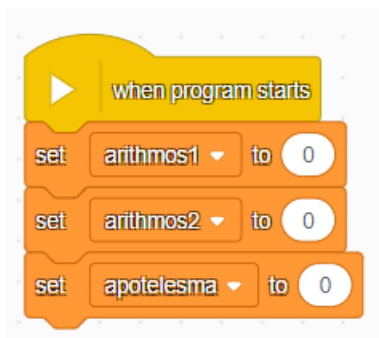
Εκφώνηση: Να δημιουργηθεί πρόγραμμα που θα δέχεται 2 αριθμούς που θα αποθηκεύονται σε 2 διαφορετικές μεταβλητές. Με τη χρήση ενός Color Sensor, θα εκτελείται μία μαθηματική πράξη. Κάθε χρώμα θα είναι υπεύθυνο για διαφορετική πράξη.

Βήμα 1: Από την κατηγορία **VARIABLES** επέλεξε το **Make a variable** για να δηλώσεις το όνομα της 1^{ης} μεταβλητής (1^{ος} αριθμός). Επανάλαβε την ίδια διαδικασία 2 φορές για τον 2^ο αριθμό και για το αποτέλεσμα της μαθηματικής πράξης.

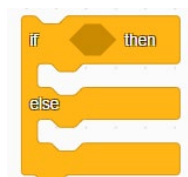


Βήμα 2: Αρχικοποιούμε τις μεταβλητές (=0) κάτω από το when program starts επιλέγοντας το

μπλοκ  και αλλάζοντας το 0 στις μεταβλητές arithmos1 και arithmos2 με τους αριθμούς με τους οποίους θέλουμε να κάνουμε τη μαθηματική πράξη.



Βήμα 3: Από την κατηγορία **CONTROL** επέλεξε το διπλανό μπλοκ για να ελέγξεις όλες τις μαθηματικές πράξεις, ορίζοντας ως συνθήκη ελέγχου



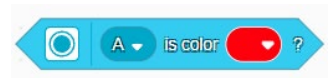
διαφορετικό χρώμα κάθε φορά.



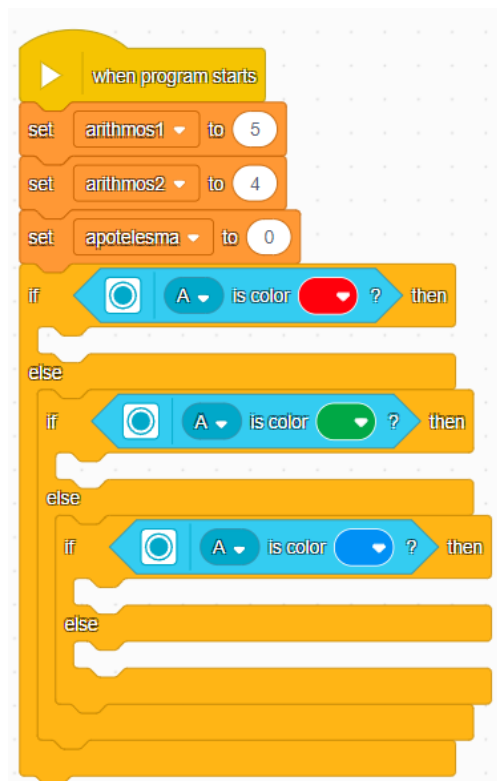
Ο λόγος που επιλέξαμε το συγκεκριμένο μπλοκ είναι γιατί υπάρχουν παραπάνω από 2 περιπτώσεις πράξεων. Αν επιλέξεις να κάνεις μόνο μία πράξη μπορείς να χρησιμοποιήσεις το διπλανό μπλοκ «if.. then».



Βήμα 4: Από την κατηγορία **SENSORS** επέλεξε το τοποθετώντας το μπλοκ στον ρόμβο του if.



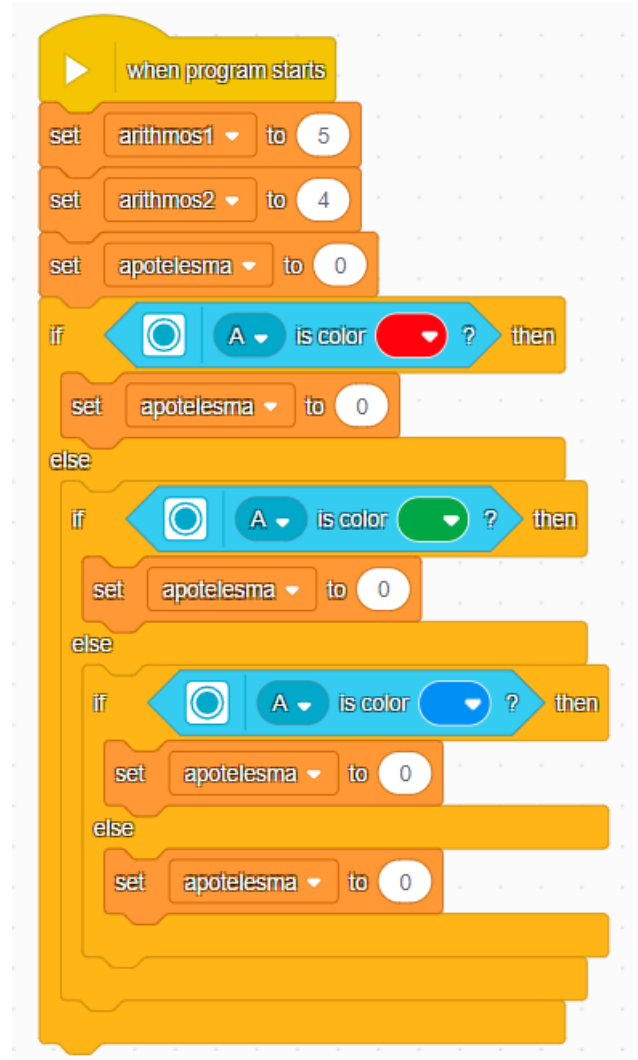
Επανάλαβε τα βήματα (3-4) άλλες 2 φορές (4 μαθηματικές πράξεις {+,-,*,/}) αλλάζοντας το χρώμα του αισθητήρα όπως φαίνεται δίπλα.



Σημείωση: Η συνθήκη του *if* γίνεται Αληθής και εκτελούνται οι εντολές που βρίσκονται μέσα του όταν ο αισθητήρας διαβάσει το σωστό χρώμα. Σε διαφορετική περίπτωση, ελέγχει ξανά το επόμενο *if*. Τέλος, αν δεν εντοπίσει κανένα σωστό χρώμα εκτελεί την τελευταία μαθηματική πράξη που θα οριστεί μέσα στο τελευταίο *else*.

Βήμα 5: Με το διάβασμα του σωστού χρώματος από τον αισθητήρα, θα πρέπει να υπολογίσεις το αποτέλεσμα της κάθε μαθηματικής πράξης ξεχωριστά και το αποτέλεσμα να το αποθηκεύσεις μέσα στην 3^η μεταβλητή που είχες δημιουργήσει προηγουμένως (result).

Από την κατηγορία **VARIABLES** επέλεξε το **set apotelesma to 0**. Το ίδιο μπλοκ θα πρέπει να μπει 4 φορές (4 μαθηματικές πράξεις {+, -, *, /}).



Βήμα 6: Από την κατηγορία OPERATORS επέλεξε τα μπλοκ και τοποθέτησε τα στη θέση του 0.



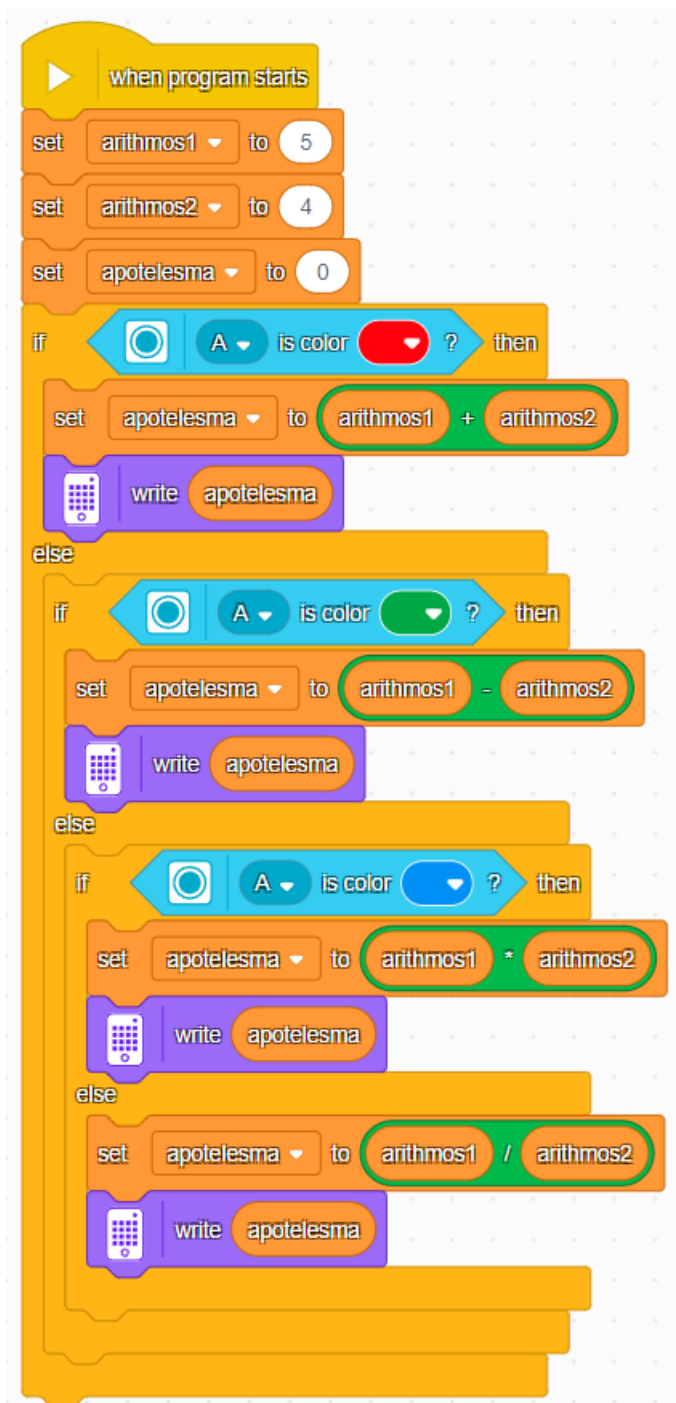
Σε κάθε κυκλάκι θα πρέπει να μπου οι μεταβλητές arithmos1 και arithmos2. Από την κατηγορία **VARIABLES**.

```
when program starts
  set arithmos1 to 5
  set arithmos2 to 4
  set apotelesma to 0
  if A is color red ? then
    set apotelesma to arithmos1 + arithmos2
  else
    if A is color green ? then
      set apotelesma to arithmos1 - arithmos2
    else
      if A is color blue ? then
        set apotelesma to arithmos1 * arithmos2
      else
        set apotelesma to arithmos1 / arithmos2
```

Βήμα 7: Τέλος για να μπορέσεις να δεις το αποτέλεσμα πάνω στο Hub, θα πρέπει να



εισάγεις από την κατηγορία LIGHT το μπλοκ και στη θέση του «Hello» να μπει η μεταβλητή «apotelesma» από την κατηγορία **VARIABLES**. Η διαδικασία πρέπει να γίνει 4 φορές.

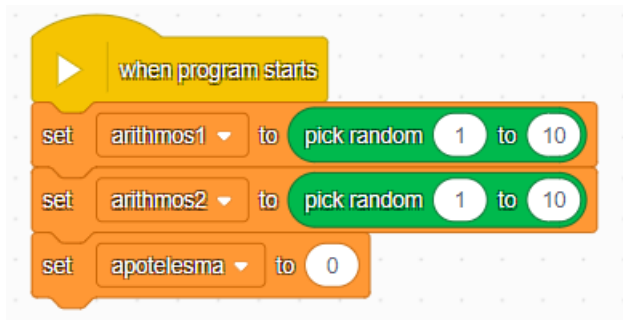


Βήμα 8 (προαιρετικό) : Αν δεν θες να συμπληρώνεις έπειτα από κάθε εκτέλεση νέους

αριθμούς, μπορείς χρησιμοποιώντας το μπλοκ “pick random”

A Scratch 'pick random' block with the range '1 to 10'.

να ορίσεις ένα εύρος τιμών, ώστε κατά την εκτέλεση του κώδικα να λαμβάνονται τυχαίοι αριθμοί. Το μπλοκ θα μπει μέσα στην αρχικοποίηση των αριθμών.



Παρατήρηση: Τι παρατηρείς κατά την εκτέλεση του κώδικά σου;

ΑΠΑΝΤΗΣΗ: Κάθε φορά που εκτελείς το πρόγραμμα, εμφανίζεται στην οθόνη του Hub το αποτέλεσμα της τελευταίας μαθηματικής πράξης. Για να μπορέσεις να διορθώσεις αυτό το σφάλμα, θα πρέπει να προσθέσεις το wait από την κατηγορία Control πριν το 1o if, ώστε να έχεις χρόνο να δείξεις στον αισθητήρα το χρώμα της επιλογής σου.

```
when program starts
  set arithmos1 to pick random 1 to 10
  set arithmos2 to pick random 1 to 10
  set apotelesma to 0
  wait 1 seconds
  if A is color red ? then
    set apotelesma to arithmos1 + arithmos2
    write apotelesma
  else
    if A is color green ? then
      set apotelesma to arithmos1 - arithmos2
      write apotelesma
    else
      if A is color blue ? then
        set apotelesma to arithmos1 * arithmos2
        write apotelesma
      else
        set apotelesma to arithmos1 / arithmos2
        write apotelesma
```

Προτεινόμενες δραστηριότητες

- Επεκτείνετε τον προηγούμενο κώδικα, ώστε να εκτελεί τις μαθηματικές πράξεις ανάμεσα σε 3 ή περισσότερους αριθμούς. Σημείωση: Δεν υπάρχει operator μπλοκ με απευθείας εκτέλεση 3 ή παραπάνω αριθμών.
- Επεκτείνετε το παραπάνω κώδικα, ώστε το πρόγραμμα να εκτελείται για πάντα. Έπειτα από κάθε πράξη θα πρέπει οι αριθμοί να αρχικοποιούνται (-0).
- Διορθώστε τον αρχικό κώδικα του παραδείγματος, ώστε σε περίπτωση που δοθεί σαν $\text{num2} = 0$ να μην μπορεί να γίνει η πράξη της διαίρεσης.