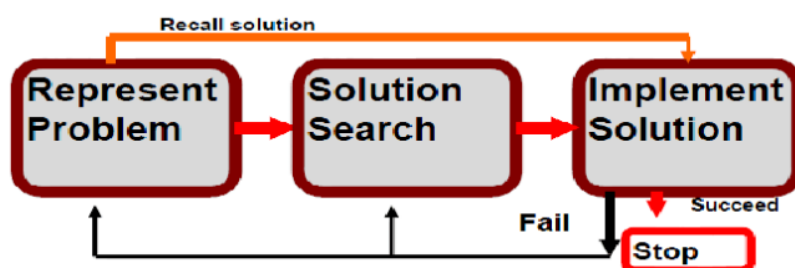


Μοντέλα επίλυσης προβλημάτων

Για γενικά προβλήματα



A Model of the Problem Solving Process (Gick, 1986)

1. Αναπαράσταση του προβλήματος
2. Αναζήτηση για λύση-ορισμός στόχου και επινόηση σχεδίου δράσης
3. Υλοποίηση της λύσης – εκτέλεση του σχεδίου και αξιολόγηση των αποτελεσμάτων

Για μαθηματικά ή μαθητικοποιημένα προβλήματα, μοντέλο “How to solve it” του Polya

1. Κατανόηση του προβλήματος
2. Δημιουργία σχεδίου
3. Εκτέλεση σχεδίου
4. Έλεγχος μαζί με τις επιμέρους αρχές, στρατηγικές και ευρετικές τεχνικές κάθε φάσης που το συνοδεύουν, σε κάθε ευκαιρία.

Αλγοριθμική

Δομή Ακολουθίας

(ψευδοκώδικας)

Αλγόριθμος: Σχεδίαση_τετραγώνου

Δεδομένα: 10

Προχώρα 10 βήματα μπροστά

Στρίψε 90°

Προχώρα 10 βήματα μπροστά

Στρίψε 90°

Προχώρα 10 βήματα μπροστά

Στρίψε 90°

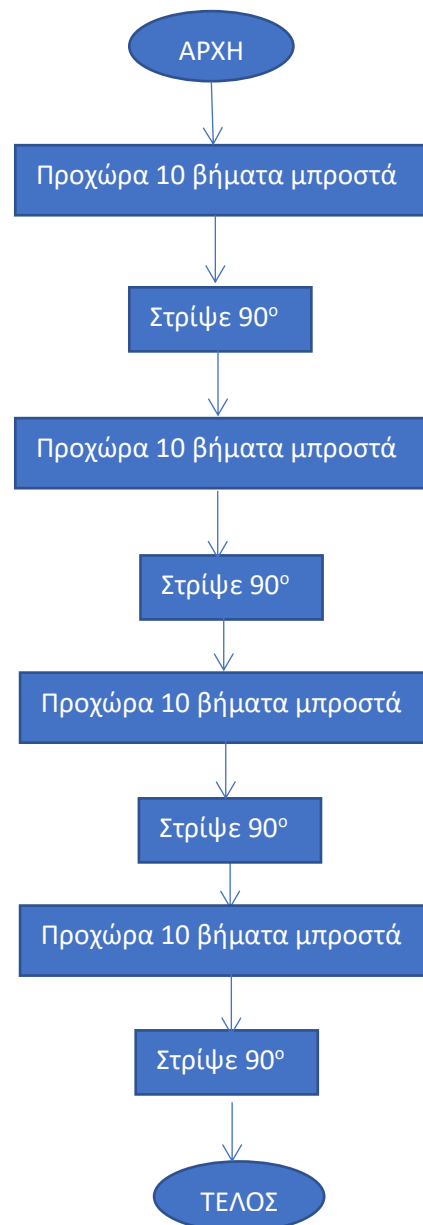
Προχώρα 10 βήματα μπροστά

Στρίψε 90°

Αλγόριθμος: Σχεδίαση_τετραγώνου

T

(διάγραμμα ροής)



*Σημείωση: Το πρόγραμμα που θα προκύψει από το συγκεκριμένο αλγόριθμο θα σχεδιάζει **πάντα** τετράγωνα με μήκος πλευράς 10. Αν θέλουμε να σχεδιάζουμε τετράγωνα με διαφορετικό μήκος πλευράς κάθε φορά χρησιμοποιώντας το ίδιο πρόγραμμα θα πρέπει να αντικαταστήσουμε το **10** με μια μεταβλητή π.χ. **μήκος-πλ***

Παράδειγμα -2

(ψευδοκώδικας)

Αλγόριθμος: Πρόσθεση αριθμών

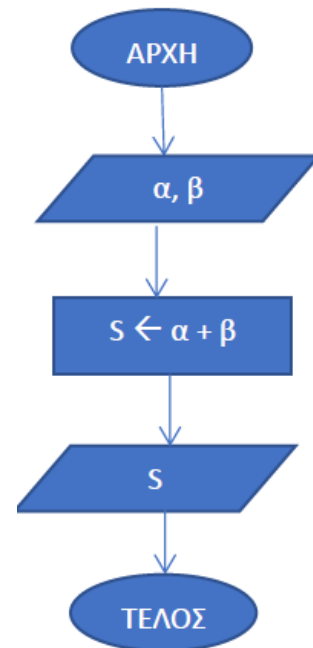
Δεδομένα: α, β

$S \leftarrow \alpha + \beta$

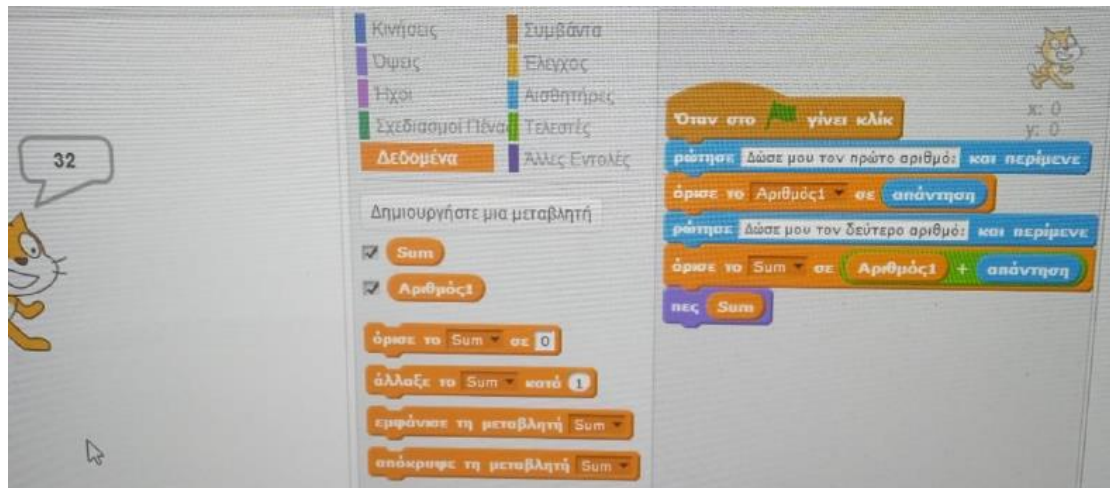
Εμφάνισε: S

Αλγόριθμος: Πρόσθεση αριθμών

(διάγραμμα ροής)



Πως υλοποιήσαμε αυτόν τον αλγόριθμο στο scratch



Δομή Επιλογής

(ψευδοκώδικας)

Αλγόριθμος: Θετικός ή Αρνητικός

Δεδομένα: Αριθμός

Αν **Αριθμός > 0** τότε

Τύπωση: «θετικός»

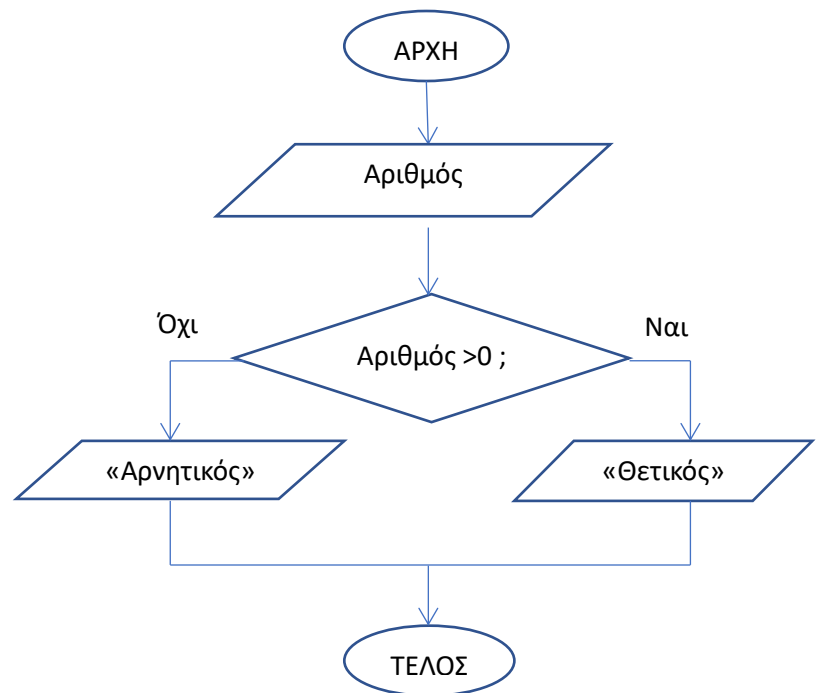
αλλιώς

Τύπωση: «αρνητικός»

Τέλος αν

Αλγόριθμος: Θετικός ή Αρνητικός

(διάγραμμα ροής)



Η σχέση **Αριθμός > 0** λέγεται **Λογική Συνθήκη**. Είτε ισχύει είτε δεν ισχύει. Δηλαδή η απάντηση στην ερώτηση αν ένας αριθμός είναι θετικός θα είναι ή Ναι ή Όχι. **Αν** η απάντηση είναι **Ναι** εκτελείτε η εντολή Τύπωση: «θετικός» ενώ **αν** η απάντηση είναι **Όχι** εκτελείτε η εντολή Τύπωση: «αρνητικός» που βρίσκεται κάτω από το αλλιώς

Δομή Επανάληψης

Παράδειγμα -1

(ψευδοκώδικας)

Αλγόριθμος: Σχεδίαση Τετραγώνου

Δεδομένα: ΜΠ

$i = 0$

Από $i = 1$ μέχρι 4 επανάλαβε:

Προχώρα ΜΠ μπροστά

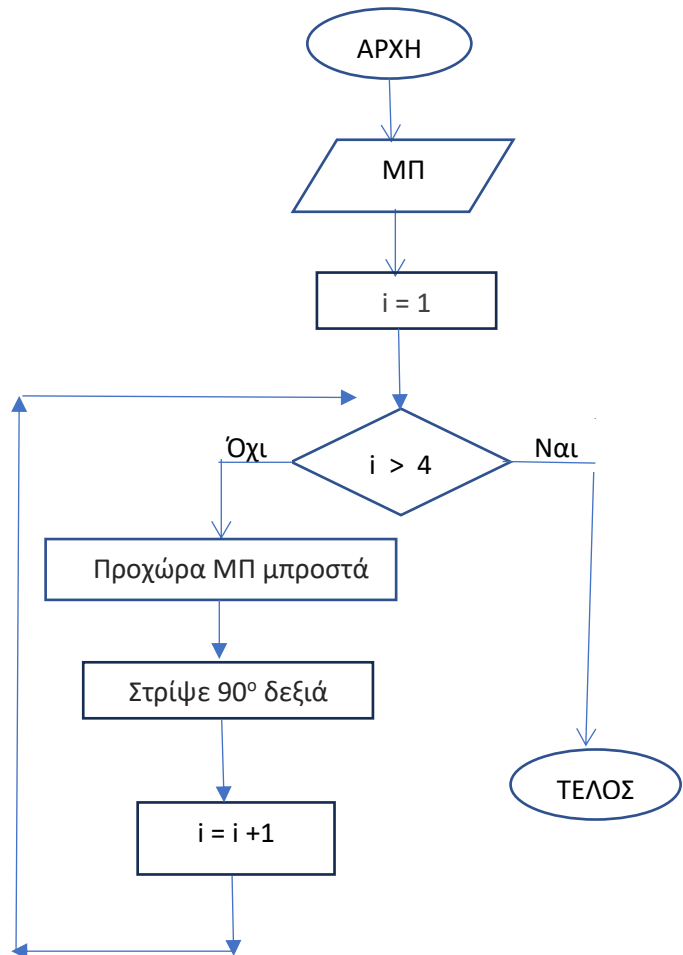
Στρίψε 90° δεξιά

$i = i + 1$

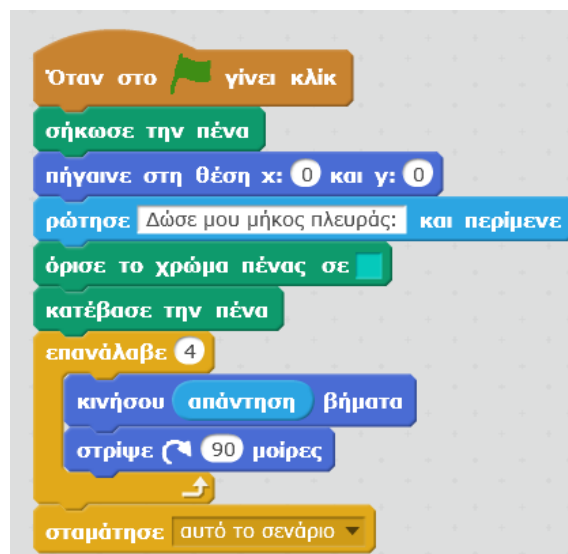
Τέλος επανάληψης

Αλγόριθμος: Σχεδίαση Τετραγώνου

(διάγραμμα ροής)



Πως υλοποιήσαμε αυτόν τον αλγόριθμο στο scratch



Παράδειγμα -2 (ψευδοκώδικας)

Αλγόριθμος: Μετρητής

$i = 0$

Για $i = 1$ μέχρι 10 επανάλαβε:

Εμφάνισε: i

$i \leftarrow i + 1$

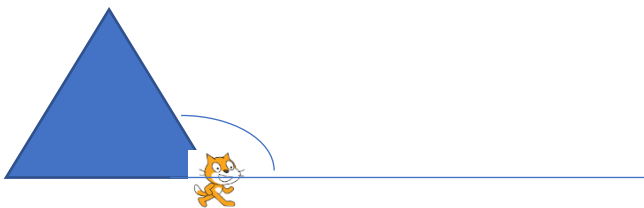
Τέλος Επανάληψης

Αλγόριθμος: Μετρητής

Παράδειγμα -3



Έστω ότι θέλουμε να προγραμματίσουμε τον scratchόγατο να σχεδιάσει ένα **ισόπλευρο τρίγωνο**. Με την εντολή **κινήσου** μετακινείτε προς τα δεξιά όσα βήματα έχουμε ορίσει. Τι στροφή θα πρέπει να κάνει όταν φτάσει στο σημείο που βλέπετε παρακάτω; Και πόσες μοίρες θα είναι η γωνία που θα στρίψει;

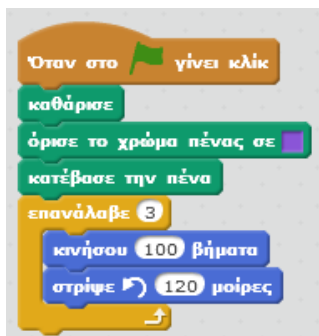


Απάντηση:

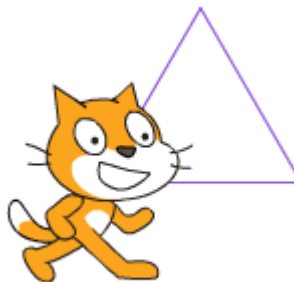
Εφόσον το τρίγωνο είναι ισόπλευρο όλες οι πλευρές του είναι ίσες και επίσης και όλες οι γωνίες του είναι ίσες 60° η καθεμία. Η γωνία που φαίνεται παραπάνω σημειωμένη είναι παραπληρωματική με την εσωτερική γωνία του τριγώνου οπότε θα είναι $180^\circ - 60^\circ = 120^\circ$ και ο γάτος θα πρέπει να στρίψει αριστερά.

Για να σχεδιαστεί το ισόπλευρο τρίγωνο αρκεί να επαναλάβουμε 3 φορές τις εντολές κινήσου και στρίψε.

Το πρόγραμμα



και το αποτέλεσμα αφού το τρέξουμε



Άσκηση: Δημιουργούμε τις δικές μας εντολές στο scratch, μία για σχεδίαση τετραγώνου και μία για σχεδίαση ισόπλευρου τριγώνου. Στο τέλος θα τις συνδυάσουμε για να σχεδιάσουμε ένα σπιτάκι.



