



Τμήμα Ηλεκτρολόγων Μηχανικών
& Μηχανικών Υπολογιστών
**ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΕΛΟΠΟΝΝΗΣΟΥ**

Τεχνικές Προγραμματισμού Υπολογιστών

Εργαστηριακό μάθημα

Εγχειρίδιο Scratch 3

Οδηγός φοιτητών

Από το περιβάλλον εργασίας και τις βασικές εντολές έως τη δομημένη ανάπτυξη, τον έλεγχο και την αποσφαλμάτωση ολοκληρωμένων έργων.

Οι βασικές κατηγορίες εντολών

Κίνηση

θέση, x/y, κατεύθυνση

Όψεις

ενδυμασίες, μέγεθος, εμφάνιση

Ήχος

ήχοι, ένταση, εφέ

Συμβάντα

εκκίνηση και μηνύματα

Έλεγχος

συνθήκες, βρόχοι, κλώνοι

Αισθητήρες

πληκτρολόγιο, ποντίκι, επαφή

Τελεστές

μαθηματικά, λογική, κείμενο

Μεταβλητές

δεδομένα που αλλάζουν

Λίστες

πολλές τιμές σε σειρά

Οι Εντολές μου

διαδικασίες και παράμετροι

Κίνηση • Όψεις • Ήχος • Συμβάντα • Έλεγχος • Αισθητήρες • Τελεστές • Δεδομένα • Διαδικασίες

Διδάσκων: **Πεφάνης Ευάγγελος**

Για χρήση στο εργαστηριακό μάθημα και για αυτοτελή μελέτη.

Πώς να χρησιμοποιήσετε το εγχειρίδιο

1. Σκοπός

Το Scratch 3 είναι ένα περιβάλλον οπτικού προγραμματισμού στο οποίο ο κώδικας δημιουργείται με πλακίδια εντολών. Ο στόχος του εγχειριδίου δεν είναι απλώς να παρουσιάσει τις διαθέσιμες εντολές, αλλά να εξηγήσει πώς συνδυάζονται για να κατασκευάζονται σωστά, ευανάγνωστα και ελέγξιμα προγράμματα.

2. Προτεινόμενος τρόπος μελέτης

1. Διαβάστε πρώτα την έννοια: τι πρόβλημα λύνει η συγκεκριμένη κατηγορία εντολών.
2. Παρατηρήστε το παράδειγμα και εντοπίστε ποιες μεταβλητές, συνθήκες ή μηνύματα χρησιμοποιούνται.
3. Υλοποιήστε το παράδειγμα μόνοι σας στο Scratch 3, αντί να το αντιγράψετε μηχανικά.
4. Δοκιμάστε κανονικές και οριακές τιμές και εξηγήστε με λόγια γιατί το πρόγραμμα συμπεριφέρεται όπως παρατηρείτε.

Κεντρική ιδέα: Το Scratch δεν καταργεί την ανάγκη για αλγοριθμική σκέψη. Αφαιρεί μεγάλο μέρος της συντακτικής δυσκολίας ώστε να εστιάζουμε στη λογική, στη δομή και στη σωστή σχεδίαση.

3. Συμβάσεις

- Οι ονομασίες των εντολών δίνονται με την ελληνική ορολογία του Scratch 3. Μικρές λεκτικές διαφορές μπορεί να εμφανίζονται ανά έκδοση ή μετάφραση.
- Οι τιμές x και y αναφέρονται στις συντεταγμένες της σκηνής και οι κατευθύνσεις σε μοίρες.
- Με τον όρο «αντικείμενο» εννοούμε ένα sprite. Η Σκηνή είναι το φόντο και ο χώρος όπου εμφανίζεται το αποτέλεσμα.
- Κάθε παράδειγμα πρέπει να μπορεί να ξεκινήσει ξανά από καθαρή αρχική κατάσταση με την πράσινη σημαία.

ΠΕΡΙΕΧΟΜΕΝΑ

Χάρτης του εγχειριδίου

Κεφάλαιο	Περιεχόμενο
1	Εισαγωγή στο Scratch 3 και μοντέλο εκτέλεσης
2	Περιβάλλον εργασίας, έργα, αντικείμενα και σκηνή
3	Κίνηση, συντεταγμένες και κατεύθυνση
4	Όψεις, ενδυμασίες, υπόβαθρα και ήχος
5	Συμβάντα, μηνύματα και παράλληλη εκτέλεση
6	Έλεγχος ροής: συνθήκες, βρόχοι και κλώνοι
7	Αισθητήρες και αλληλεπίδραση
8	Τελεστές: αριθμοί, λογική και κείμενο
9	Μεταβλητές, λίστες και δεδομένα
10	Οι Εντολές μου: διαδικασίες και παράμετροι
11	Πένα και γεωμετρικός σχεδιασμός
12	Επεκτάσεις και φυσική υπολογιστική
13	Αποσφαλμάτωση, δοκιμές και απόδοση
14	Σχεδίαση και οργάνωση ολοκληρωμένου έργου
15	Ολοκληρωμένα παραδείγματα και τελικός έλεγχος
Παράρτημα	Γλωσσάρι, σύντομη αναφορά και πηγές

Για το εργαστήριο: Τα Κεφάλαια 1–10 αποτελούν τον βασικό κορμό. Τα Κεφάλαια 11–15

χρησιμοποιούνται όταν οι ασκήσεις απαιτούν σχεδίαση, μεγαλύτερη οργάνωση, επαναχρησιμοποίηση κώδικα ή συστηματικό έλεγχο.

ΚΕΦΑΛΑΙΟ 1 / ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ

Τι είναι ένα έργο Scratch

1. Τα τέσσερα βασικά συστατικά

Ένα έργο Scratch οργανώνεται γύρω από τη Σκηνή και τα αντικείμενα. Κάθε αντικείμενο μπορεί να έχει δικό του κώδικα, ενδυμασίες και ήχους. Το αποτέλεσμα προκύπτει από τη συνεργασία όλων αυτών των στοιχείων.

Στοιχείο	Ρόλος
Σκηνή	Ο χώρος προβολής. Μπορεί να αλλάζει υπόβαθρο και να έχει δικά της σενάρια.
Αντικείμενο (sprite)	Οντότητα που κινείται, αλλάζει όψη, παίζει ήχο, λαμβάνει μηνύματα και εκτελεί κώδικα.
Σενάριο	Συνδεδεμένη ακολουθία πλακιδίων που ξεκινά από συμβάν ή καλείται ως διαδικασία.
Δεδομένα	Μεταβλητές και λίστες που αποθηκεύουν την κατάσταση του έργου.

2. Οπτικός προγραμματισμός δεν σημαίνει «χωρίς κανόνες»

Τα πλακίδια κουμπώνουν μόνο σε επιτρεπτές θέσεις, αλλά ένα πρόγραμμα μπορεί παρ' όλα αυτά να είναι λογικά λανθασμένο. Διαίρεση με μηδέν, βρόχος που δεν τερματίζει ή δύο παράλληλα σενάρια που αλλάζουν την ίδια μεταβλητή μπορούν να δημιουργήσουν πρόβλημα χωρίς «συντακτικό λάθος».

Τι μαθαίνουμε: Ακολουθία, επιλογή, επανάληψη, μεταβλητές, διαδικασίες, γεγονότα και επικοινωνία μεταξύ τμημάτων του προγράμματος.

ΚΕΦΑΛΑΙΟ 1 / ΣΕΝΑΡΙΑ

Πώς εκτελείται ο κώδικας

Σενάρια, συμβάντα και παράλληλη εκτέλεση

όταν γίνει κλικ στην πράσινη σημαία

όρισε σκορ σε 0

επανάλαβε 10

κινήσου 10 βήματα

Κανόνες εκτέλεσης

1. Μέσα στο ίδιο σενάριο: από πάνω προς τα κάτω.
2. Βρόχος: επαναλαμβάνει το εσωτερικό του.
3. Πολλά σενάρια μπορούν να τρέχουν ταυτόχρονα.
4. Η θέση ενός σεναρίου στην οθόνη δεν δίνει προτεραιότητα.

Μέσα στο ίδιο σενάριο οι εντολές εκτελούνται διαδοχικά, ενώ διαφορετικά σενάρια μπορούν να είναι ενεργά παράλληλα.

1. Από πάνω προς τα κάτω

Μέσα σε ένα συγκεκριμένο σενάριο, οι εντολές εκτελούνται με τη σειρά που είναι συνδεδεμένες. Αν μια εντολή περιμένει, ζητά απάντηση ή εκτελεί βρόχο, το επόμενο πλακίδιο του ίδιου σεναρίου περιμένει να ολοκληρωθεί.

2. Πολλά σενάρια ταυτόχρονα

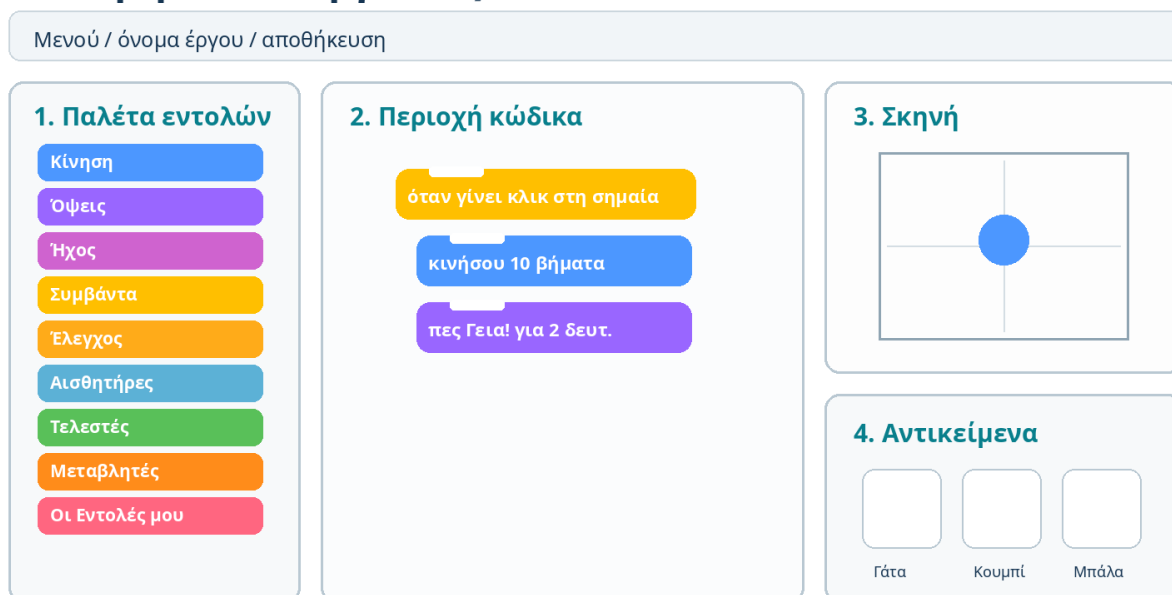
Δύο ή περισσότερα σενάρια μπορούν να είναι ενεργά την ίδια στιγμή. Αυτό είναι ισχυρό, αλλά απαιτεί προσοχή όταν τα σενάρια μοιράζονται τις ίδιες μεταβλητές ή αλλάζουν την ίδια κατάσταση.

Συχνό λάθος: Η χωρική θέση των σεναρίων στην περιοχή κώδικα δεν καθορίζει σειρά ή προτεραιότητα εκτέλεσης.

ΚΕΦΑΛΑΙΟ 2 / ΠΕΡΙΒΑΛΛΟΝ ΕΡΓΑΣΙΑΣ

Η οθόνη του Scratch 3

Το περιβάλλον εργασίας του Scratch 3



Επιλέγουμε πρώτα αντικείμενο και μετά γράφουμε τον κώδικά του.

Βασικές περιοχές του επεξεργαστή Scratch 3.

1. Επιλέγω αντικείμενο πριν γράψω κώδικα

Ο κώδικας που βλέπετε ανήκει στο αντικείμενο που είναι επιλεγμένο κάτω από τη σκηνή. Αν επιλέξετε άλλο αντικείμενο, αλλάζει και ο κώδικας που εμφανίζεται. Πριν προσθέσετε εντολή, βεβαιωθείτε ότι επεξεργάζεστε το σωστό sprite.

2. Οι τρεις καρτέλες

Καρτέλα	Τι περιέχει
Κώδικας	Παλέτα εντολών και περιοχή σεναρίων.
Ενδυμασίες	Γραφικός επεξεργαστής των εμφανίσεων του αντικειμένου.
Ήχοι	Ηχητικά αρχεία και βασικά εργαλεία επεξεργασίας ήχου.

Δημιουργία, αποθήκευση και οργάνωση έργου

1. Το αρχείο .sb3

Κάθε έργο Scratch 3 μπορεί να αποθηκευτεί ως αρχείο .sb3. Περιέχει αντικείμενα, υπόβαθρα, ήχους, μεταβλητές, λίστες και σενάρια. Για εργασίες μαθήματος, κάθε ανεξάρτητη άσκηση είναι καλύτερο να αποθηκεύεται σε ξεχωριστό αρχείο με σαφές όνομα.

2. Προτεινόμενη διαδικασία

1. Δημιουργήστε νέο έργο και δώστε από την αρχή περιγραφικό όνομα.
2. Αποθηκεύστε νωρίς, πριν ξεκινήσουν μεγάλες αλλαγές.
3. Χρησιμοποιείτε εκδόσεις όταν κάνετε σημαντική αναδόμηση, π.χ. calculator_v1.sb3, calculator_v2.sb3.
4. Πριν από την υποβολή, κλείστε και ανοίξτε ξανά το αρχείο.
5. Εκτελέστε την εφαρμογή από την πράσινη σημαία και όχι από τυχαία μεμονωμένα σενάρια.

Καλή πρακτική: Το έργο πρέπει να αρχικοποιείται μόνο του. Ο χρήστης δεν πρέπει να διορθώνει χειροκίνητα μεταβλητές ή θέσεις πριν ξεκινήσει.

3. Ονόματα που βοηθούν

Ασαφές	Καλύτερο
sprite1	Παρουσιαστής
a1	σκορ
x2	κάτω όριο
message1	έναρξη παιχνιδιού
block1	Έλεγχε είσοδο

Ιδιότητες αντικειμένου και Σκηνής

1. Ιδιότητες ενός sprite

- Θέση x και y: καθορίζει πού βρίσκεται στη σκηνή.
- Κατεύθυνση: καθορίζει προς τα πού είναι στραμμένο.
- Μέγεθος: ποσοστό σε σχέση με το αρχικό μέγεθος της ενδυμασίας.
- Ορατότητα: εμφανισμένο ή κρυμμένο.
- Στυλ περιστροφής: πλήρης περιστροφή, αριστερά-δεξιά ή χωρίς περιστροφή.

2. Τι διαφέρει στη Σκηνή

Η Σκηνή δεν κινείται όπως ένα sprite και γι' αυτό δεν διαθέτει εντολές Κίνησης. Έχει όμως υπόβαθρα και μπορεί να ανταποκρίνεται σε συμβάντα, να παίζει ήχους, να διαχειρίζεται δεδομένα και να στέλνει ή να λαμβάνει μηνύματα.

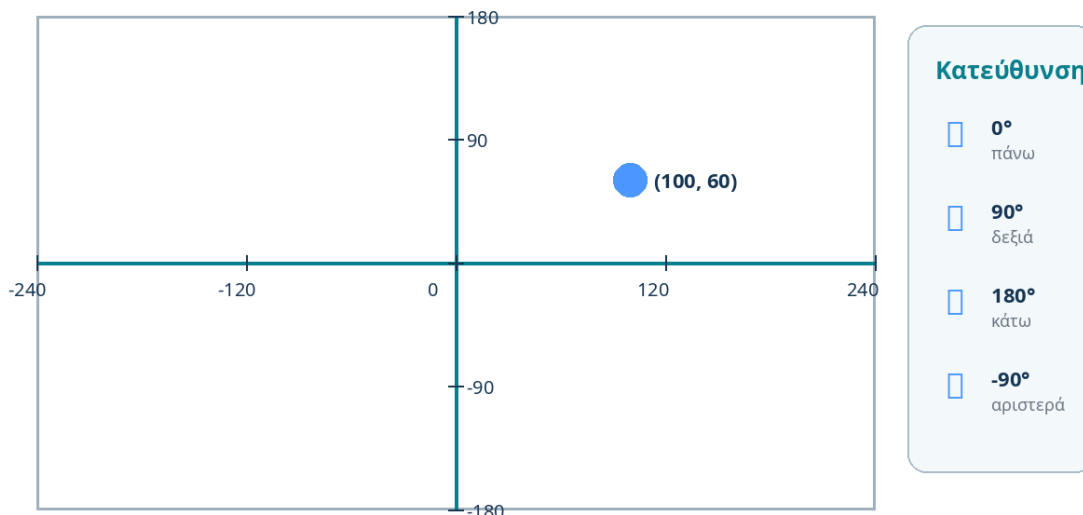
Πρακτική επιλογή: Χρησιμοποιήστε τη Σκηνή για γενικά συμβάντα και αλλαγές υποβάθρου. Χρησιμοποιήστε sprites για οντότητες με ξεχωριστή συμπεριφορά.

3. Το κέντρο ενός αντικειμένου

Οι συντεταγμένες ενός sprite αναφέρονται στο κέντρο περιστροφής της ενδυμασίας του. Αν η εικόνα περιστρέφεται παράξένα, ελέγξτε το κέντρο στην καρτέλα «Ενδυμασίες».

Το σύστημα συντεταγμένων

Συντεταγμένες και κατεύθυνση στη σκηνή



Η σκηνή εκτείνεται από $x = -240$ έως 240 και από $y = -180$ έως 180 .

1. Οριζόντια και κατακόρυφη θέση

Το x αυξάνεται προς τα δεξιά και μειώνεται προς τα αριστερά. Το y αυξάνεται προς τα πάνω και μειώνεται προς τα κάτω. Το σημείο $(0, 0)$ βρίσκεται στο κέντρο της σκηνής.

2. Κατεύθυνση

Στο Scratch, 90° σημαίνει δεξιά, 0° πάνω, -90° αριστερά και 180° κάτω. Η κατεύθυνση είναι ανεξάρτητη από τη θέση του αντικειμένου.

Έλεγχος: Για ακριβή τοποθέτηση χρησιμοποιήστε «πήγαινε σε $x:y$ ». Για κίνηση προς την τρέχουσα κατεύθυνση χρησιμοποιήστε «κινήσου ... βήματα».

ΚΕΦΑΛΑΙΟ 3 / ΚΙΝΗΣΗ

Οι βασικές εντολές Κίνησης

Εντολή / ιδέα	Πότε τη χρησιμοποιούμε
κινήσου N βήματα	Μετακίνηση προς την τρέχουσα κατεύθυνση.
στρίψε δεξιά/αριστερά N μοίρες	Αλλαγή προσανατολισμού.
πήγαινε σε $x:y$	Άμεση τοποθέτηση σε συγκεκριμένη θέση.
ολίσθησε ... δευτ. σε $x:y$	Ομαλή μετάβαση σε θέση.
άλλαξε x κατά / άλλαξε y κατά	Σχετική μετατόπιση σε έναν άξονα.
δείξε προς ...	Στρέφει το sprite προς ποντίκι ή άλλο αντικείμενο.
εάν αγγίζει όριο, αναπήδησε	Χρήσιμο σε απλά παιχνίδια κίνησης.

1. Απόλυτη και σχετική κίνηση

Η εντολή «πήγαινε σε $x:y$ » είναι απόλυτη: δεν εξαρτάται από την προηγούμενη θέση. Η «άλλαξε x κατά 10» είναι σχετική: προσθέτει 10 στην τρέχουσα τιμή του x .

2. Οριακές θέσεις

Το κέντρο ενός sprite μπορεί να βρίσκεται μέσα στη σκηνή ενώ μέρος της εικόνας του βγαίνει εκτός. Όταν το ακριβές όριο έχει σημασία, λάβετε υπόψη και τις διαστάσεις της ενδυμασίας.

ΚΕΦΑΛΑΙΟ 3 / ΠΡΟΤΥΠΑ ΚΙΝΗΣΗΣ

Κίνηση με πληκτρολόγιο και ομαλή συμπεριφορά

1. Συνεχής έλεγχος πλήκτρων

Σε παιχνίδια είναι συνηθισμένο να υπάρχει βρόχος «για πάντα» που ελέγχει αν είναι πατημένα τα βέλη. Έτσι το αντικείμενο κινείται όσο το πλήκτρο παραμένει πατημένο.

Έλεγχος	Ενέργεια
δεξί βέλος	άλλαξε x κατά +5
αριστερό βέλος	άλλαξε x κατά -5
πάνω βέλος	άλλαξε y κατά +5
κάτω βέλος	άλλαξε y κατά -5

2. Ταχύτητα ως μεταβλητή

Αν αντί για σταθερό 5 χρησιμοποιήσετε μεταβλητή «ταχύτητα», μπορείτε να αλλάζετε εύκολα τη συμπεριφορά χωρίς να διορθώνετε πολλές εντολές.

Σχεδιαστική αρχή: Οι αριθμοί που επηρεάζουν σημαντικά τη συμπεριφορά ενός έργου είναι συχνά καλύτερο να αποθηκεύονται σε περιγραφικές μεταβλητές αντί να επαναλαμβάνονται ως «μαγικοί αριθμοί».

ΚΕΦΑΛΑΙΟ 4 / ΟΨΕΙΣ

Ενδυμασίες, μέγεθος και ορατότητα

1. Ενδυμασίες

Ένα sprite μπορεί να διαθέτει πολλές ενδυμασίες. Με αλλαγή ενδυμασίας δημιουργούμε animation, διαφορετικές καταστάσεις ή οπτική ανατροφοδότηση. Η «επόμενη ενδυμασία» είναι κατάλληλη για κυκλικό animation, ενώ η «άλλαξε ενδυμασία σε ...» είναι ασφαλέστερη όταν απαιτείται συγκεκριμένη κατάσταση.

2. Εμφάνιση και απόκρυψη

Οι εντολές «εμφανίσου» και «εξαφανίσου» δεν διαγράφουν το αντικείμενο ούτε σταματούν τον κώδικά του. Ένα κρυμμένο sprite μπορεί να συνεχίσει να εκτελεί σενάρια και να αλλάζει μεταβλητές.

3. Επίπεδα εμφάνισης

Οι εντολές μετακίνησης σε μπροστινά ή πίσω επίπεδα καθορίζουν ποια αντικείμενα καλύπτουν άλλα. Αυτό είναι σημαντικό σε μενού, κουμπιά και πολυεπίπεδα γραφικά.

Συχνή παρεξήγηση: Η αλλαγή μεγέθους δεν αλλάζει τις συντεταγμένες του κέντρου, επηρεάζει όμως τον χώρο που καταλαμβάνει οπτικά το αντικείμενο και τις επαφές του.

Επεξεργαστής ενδυμασιών και υπόβαθρα

1. Διανυσματική και bitmap επεξεργασία

Στη διανυσματική μορφή τα σχήματα παραμένουν καθαρά όταν αλλάζει το μέγεθος και μπορούν να επεξεργάζονται ως ξεχωριστά αντικείμενα. Στη bitmap μορφή η εικόνα επεξεργάζεται ως πλέγμα εικονοστοιχείων.

2. Κέντρο περιστροφής

Το κέντρο της ενδυμασίας καθορίζει το σημείο γύρω από το οποίο περιστρέφεται το sprite και το σημείο στο οποίο αντιστοιχούν οι συντεταγμένες x και y. Για βέλη, οχήματα ή δείκτες, η σωστή τοποθέτηση του κέντρου είναι κρίσιμη.

3. Υπόβαθρα

Τα υπόβαθρα ανήκουν στη Σκηνή. Η αλλαγή υποβάθρου μπορεί να χρησιμοποιηθεί ως αλλαγή οθόνης: αρχικό μενού, παιχνίδι, οδηγίες, τέλος.

Οργάνωση: Αν χρησιμοποιείτε υπόβαθρα ως «οθόνες», συνδυάστε τα με μεταβλητή κατάστασης ή μηνύματα ώστε τα αντικείμενα να αντιδρούν στη σωστή φάση.

Ήχος, ένταση και χρονισμός

Εντολή	Συμπεριφορά
Ξεκινά ήχο ...	Η αναπαραγωγή ξεκινά και το σενάριο συνεχίζει αμέσως.
παίξε ήχο ... μέχρι τέλους	Το σενάριο περιμένει να ολοκληρωθεί ο ήχος πριν συνεχίσει.

1. Πότε χρησιμοποιούμε κάθε μορφή

Για μουσική υπόκρουση ή εφέ που δεν πρέπει να καθυστερούν το παιχνίδι χρησιμοποιούμε συνήθως την πρώτη μορφή. Για αφήγηση ή διαδοχικούς ήχους που πρέπει να ακουστούν με σειρά, χρησιμοποιούμε την αναμονή μέχρι το τέλος.

2. Ένταση και ηχητικά εφέ

Η ένταση μπορεί να οριστεί ή να αλλάξει σταδιακά. Τα ηχητικά εφέ μεταβάλλουν τον τρόπο αναπαραγωγής και μπορούν να επανέλθουν με καθαρισμό των εφέ.

Χρονισμός: Όταν ένα ηχητικό μήνυμα πρέπει να προηγείται της επόμενης ενέργειας, χρησιμοποιήστε εντολή που περιμένει. Διαφορετικά η εφαρμογή μπορεί να συνεχίσει πριν ολοκληρωθεί η αφήγηση.

Πώς ξεκινά ένα σενάριο

Τα συμβάντα είναι τα «σημεία εκκίνησης» του Scratch. Ένα σενάριο δεν εκτελείται επειδή απλώς υπάρχει στην περιοχή κώδικα: πρέπει να ενεργοποιηθεί από ένα συμβάν, όπως η πράσινη σημαία, το πάτημα ενός πλήκτρου, ένα κλικ ή η λήψη ενός μηνύματος.

1. Συνηθισμένα συμβάντα

Συμβάν	Χρήση
όταν γίνει κλικ στην πράσινη σημαία	Γενική αρχικοποίηση και έναρξη έργου.
όταν πατηθεί πλήκτρο ...	Άμεσος χειρισμός από το πληκτρολόγιο.
όταν γίνει κλικ σε αυτό το αντικείμενο	Κουμπιά, επιλογές, διαδραστικά στοιχεία.
όταν το υπόβαθρο αλλάξει σε ...	Ενέργειες που συνδέονται με αλλαγή οθόνης.
όταν λάβω ...	Συντονισμός αντικειμένων μέσω μηνυμάτων.

2. Η πράσινη σημαία ως «επανεκκίνηση»

Στο εκπαιδευτικό έργο είναι χρήσιμο να αντιμετωπίζουμε την πράσινη σημαία ως πλήρη επανεκκίνηση. Θέσεις, ενδυμασίες, σκορ, μετρητές, κρυφές μεταβλητές και κατάσταση παιχνιδιού πρέπει να αποκτούν γνωστές αρχικές τιμές. Έτσι η δεύτερη εκτέλεση δεν εξαρτάται από ό,τι συνέβη στην πρώτη.

Έλεγχος ποιότητας: Πατήστε την πράσινη σημαία, αλλάξτε αρκετά την κατάσταση του έργου και πατήστε την ξανά. Αν το αποτέλεσμα δεν επιστρέφει σε καθαρή αρχική κατάσταση, λείπει κάποια αρχικοποίηση.

ΚΕΦΑΛΑΙΟ 5 / ΜΗΝΥΜΑΤΑ

Επικοινωνία μεταξύ αντικειμένων

Τα μηνύματα επιτρέπουν σε ένα αντικείμενο να ζητά από άλλα αντικείμενα να εκτελέσουν συγκεκριμένη συμπεριφορά, χωρίς να χρειάζεται να γνωρίζει πώς είναι υλοποιημένη. Αυτό μειώνει την εξάρτηση ανάμεσα στα sprites και βοηθά να οργανώσουμε το έργο σε σαφείς ρόλους.

1. «μετάδωσε» και «μετάδωσε ... και περίμενε»

Εντολή	Τι συμβαίνει
μετάδωσε [μήνυμα]	Στέλνει το μήνυμα και το τρέχον σενάριο συνεχίζει αμέσως.
μετάδωσε [μήνυμα] και περίμενε	Περιμένει να ολοκληρωθούν τα σενάρια που ενεργοποιήθηκαν από το μήνυμα και μετά συνεχίζει.

2. Παράδειγμα οργάνωσης

1. Ο χρήστης πατά το κουμπί «Έναρξη».
2. Το κουμπί μεταδίδει «έναρξη παιχνιδιού».
3. Ο Παίκτης μετακινείται στην αρχική θέση.
4. Ο Ελεγκτής μηδενίζει σκορ και χρόνο.
5. Η Σκηνή αλλάζει υπόβαθρο.

Κάθε αντικείμενο εκτελεί μόνο τη δική του ευθύνη. Το κουμπί δεν χρειάζεται να αλλάζει το σκορ, να μετακινεί τον παίκτη και να αλλάζει υπόβαθρο μόνο του.

Ονοματοδοσία: Προτιμήστε μηνύματα όπως «έναρξη γύρου», «τέλος παιχνιδιού», «νέα ερώτηση» αντί για «μήνυμα1», «μήνυμα2».

Παράλληλη εκτέλεση και κοινή κατάσταση

Όταν δύο σενάρια είναι ενεργά παράλληλα, η σειρά με την οποία μεταβάλλουν κοινά δεδομένα μπορεί να επηρεάσει το αποτέλεσμα. Στο Scratch αυτό εμφανίζεται συχνά όταν δύο κουμπιά πατηθούν γρήγορα, όταν πολλοί κλώνοι αλλάζουν το ίδιο σκορ ή όταν περισσότερα σενάρια αντιδρούν στο ίδιο μήνυμα.

1. Μεταβλητή κατάστασης ως «κλείδωμα»

Μια κοινή μεταβλητή, για παράδειγμα «απασχολημένος», μπορεί να δείχνει αν μια λειτουργία βρίσκεται ήδη σε εξέλιξη. Το πρώτο αποδεκτό πάτημα την κάνει 1. Όσο είναι 1, νέα πατήματα αγνοούνται. Όταν η λειτουργία ολοκληρωθεί, επιστρέφει σε 0.

Κατάσταση	Σημασία
απασχολημένος = 0	Μπορεί να ξεκινήσει νέα λειτουργία.
απασχολημένος = 1	Υπάρχει ενεργή λειτουργία· απορρίπτονται νέες εκκινήσεις.

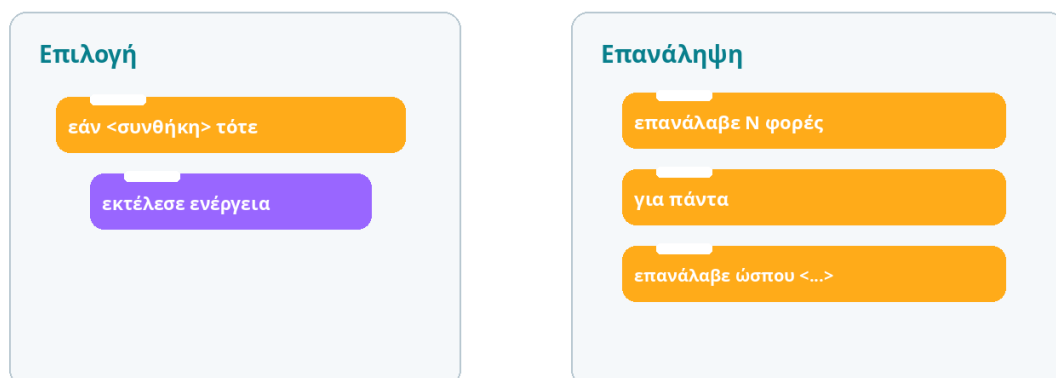
2. Μην χρησιμοποιείτε τυχαίες αναμονές για «συγχρονισμό»

Η προσθήκη «περίμενε 0.2 δευτερόλεπτα» για να ελπίζουμε ότι ένα άλλο σενάριο θα έχει ολοκληρωθεί είναι εύθραυστη λύση. Προτιμήστε «μετάδοση και περίμενε», σαφείς μεταβλητές κατάστασης ή ένα συγκεκριμένο μήνυμα ολοκλήρωσης.

Σκέψη προγραμματιστή: Ρωτήστε «ποιος επιτρέπεται να αλλάξει αυτή τη μεταβλητή και πότε;». Αν η απάντηση δεν είναι σαφής, η αρχιτεκτονική χρειάζεται απλοποίηση.

Συνθήκες και λογικές αποφάσεις

Συνθήκες, βρόχοι και Boolean λογική



Οι δομές ελέγχου καθορίζουν ποιος κλάδος εκτελείται και πόσες φορές επαναλαμβάνεται μια ενέργεια.

1. «εάν» ή «εάν ... αλλιώς»;

Χρησιμοποιήστε «εάν» όταν χρειάζεται ενέργεια μόνο στην αληθή περίπτωση. Χρησιμοποιήστε «εάν ... αλλιώς» όταν πρέπει να εκτελεστεί ακριβώς μία από δύο εναλλακτικές. Για περισσότερες περιπτώσεις μπορούμε να εμφωλεύσουμε συνθήκες, αλλά πρέπει να διατηρούμε ευανάγνωστη δομή.

Πρόβλημα	Κατάλληλη δομή
Αν ο παίκτης αγγίζει το κέρμα, αύξησε σκορ.	εάν
Αν η απάντηση είναι σωστή, δώσε πόντο, αλλιώς δείξε λάθος.	εάν ... αλλιώς
Αν $\Delta < 0$, αν $\Delta = 0$, αν $\Delta > 0$.	Εμφωλευμένες/διαδοχικές συνθήκες με αμοιβαία αποκλειόμενες περιπτώσεις.

ΚΕΦΑΛΑΙΟ 6 / ΛΟΓΙΚΗ

Συγκρίσεις και λογικοί τελεστές

Οι συνθήκες στο Scratch παράγουν αληθή ή ψευδή τιμή. Οι συγκρίσεις $<$, $=$ και $>$ συνδυάζονται με τους λογικούς τελεστές «και», «ή» και «όχι». Η σωστή διατύπωση της συνθήκης είναι συχνά σημαντικότερη από τις εντολές που βρίσκονται μέσα στον κλάδο.

1. Παραδείγματα

Ζητούμενο	Λογική έκφραση
Ηλικία από 18 έως 65	ηλικία ≥ 18 ΚΑΙ ηλικία ≤ 65
Πλήκτρο αριστερά ή A	πατημένο αριστερό βέλος Ή πατημένο A
Δεν αγγίζει τον τοίχο	ΟΧΙ (αγγίζει τοίχο)
Τιμή εκτός 1-100	τιμή < 1 Ή τιμή > 100

2. Προτεραιότητα στη σαφήνεια

Όταν μια συνθήκη γίνεται μεγάλη, είναι συχνά καλύτερο να τη χωρίσετε σε δύο βήματα ή να αποθηκεύσετε ένα ενδιάμεσο αποτέλεσμα σε περιγραφική μεταβλητή. Ο στόχος δεν είναι να χωρέσει όλη η λογική σε ένα πλακίδιο αλλά να μπορεί να ελεγχθεί.

Κανόνας: Η συνθήκη πρέπει να μπορεί να διαβαστεί ως πρόταση. Αν χρειάζεται μεγάλη προσπάθεια για να εξηγηθεί, πιθανότατα χρειάζεται απλούστευση.

ΚΕΦΑΛΑΙΟ 6 / ΕΠΑΝΑΛΗΨΗ

Βρόχοι και κριτήρια τερματισμού

Δομή	Πότε ταιριάζει
επανάλαβε N	Γνωρίζουμε εκ των προτέρων πόσες επαναλήψεις θέλουμε.
για πάντα	Η συμπεριφορά πρέπει να λειτουργεί όσο εκτελείται το έργο.
επανάλαβε ώσπου ...	Συνεχίζουμε μέχρι να γίνει αληθής μια συνθήκη.
περίμενε ώσπου ...	Δεν εκτελούμε σώμα βρόχου· απλώς αναμένουμε μια συνθήκη.

1. Βρόχος που τερματίζει

Σε «επανάλαβε ώσπου» πρέπει κάτι μέσα ή γύρω από τον βρόχο να μπορεί να αλλάξει τη συνθήκη. Αν, για παράδειγμα, περιμένουμε «ώσπου προσπάθειες = 3» αλλά δεν αυξάνουμε ποτέ τις προσπάθειες, το πρόγραμμα δεν θα τερματίσει.

2. «για πάντα» με λόγο ύπαρξης

Ο βρόχος «για πάντα» είναι κατάλληλος για συνεχή έλεγχο πληκτρολογίου, animation ή ανίχνευση συγκρούσεων. Δεν πρέπει να χρησιμοποιείται από συνήθεια σε κώδικα που θα μπορούσε να ολοκληρωθεί φυσιολογικά.

Δοκιμή τερματισμού: Πριν εκτελέσετε έναν βρόχο, εντοπίστε ποια μεταβλητή ή ποιο γεγονός μπορεί να κάνει τη συνθήκη εξόδου αληθή.

ΚΕΦΑΛΑΙΟ 6 / ΔΙΑΚΟΠΗ

Σταμάτημα σεναρίων και ασφαλής έξοδος

Το Scratch παρέχει εντολές διακοπής για το τρέχον σενάριο, για όλα τα σενάρια ή για άλλα σενάρια του ίδιου αντικειμένου. Η επιλογή πρέπει να γίνεται προσεκτικά: η γενική διακοπή μπορεί να σταματήσει και χρήσιμες λειτουργίες, όπως μουσική, χρονόμετρο ή διαχείριση μενού.

1. Προτιμήστε ελεγχόμενο τερματισμό

Σε πολλά έργα είναι καθαρότερο να αλλάζει μια μεταβλητή κατάσταση, π.χ. «παιχνίδι ενεργό = 0», και τα επιμέρους σενάρια να σταματούν φυσιολογικά όταν τη δουν, αντί να διακόπτονται απροειδοποίητα.

2. Τερματισμός μετά από σφάλμα

Αν μια είσοδος καθιστά αδύνατη την πράξη, όπως διαίρεση με μηδέν, το σενάριο πρέπει να εμφανίζει σαφές μήνυμα και να αποφεύγει τον μη επιτρεπτό υπολογισμό. Η ασφάλεια προκύπτει από τη δομή των κλάδων, όχι από την ελπίδα ότι ο χρήστης δεν θα δώσει την τιμή.

ΚΕΦΑΛΑΙΟ 6 / ΚΛΩΝΟΙ

Πολλά παρόμοια αντικείμενα χωρίς αντιγραφή

Οι κλώνοι δημιουργούν προσωρινά αντίγραφα ενός sprite κατά την εκτέλεση. Είναι ιδιαίτερα χρήσιμοι για σφαίρες, εχθρούς, αστέρια, αντικείμενα που πέφτουν ή επαναλαμβανόμενα στοιχεία διεπαφής.

1. Κύκλος ζωής κλώνου

1. Το αρχικό sprite δημιουργεί έναν κλώνο.
2. Ο νέος κλώνος ενεργοποιεί το συμβάν «όταν ξεκινήσω ως κλώνος».
3. Ο κλώνος αρχικοποιεί τη δική του θέση ή ιδιότητες.
4. Εκτελεί τη συμπεριφορά του.
5. Όταν δεν χρειάζεται πλέον, εκτελεί «διάγραψε αυτόν τον κλώνο».

2. Τοπικές και κοινές μεταβλητές

Μεταβλητή «μόνο για αυτό το αντικείμενο» μπορεί να έχει διαφορετική τιμή σε κάθε κλώνο. Μεταβλητή «για όλα τα αντικείμενα» είναι κοινή και επομένως όλοι οι κλώνοι βλέπουν και αλλάζουν την ίδια τιμή.

Συχνό σφάλμα: Κλώνοι που δεν διαγράφονται συσσωρεύονται και αυξάνουν την υπολογιστική επιβάρυνση. Προβλέψτε σαφή συνθήκη λήξης.

ΚΕΦΑΛΑΙΟ 7 / ΑΙΣΘΗΤΗΡΕΣ

Πώς το έργο «παρατηρεί» τον χρήστη και το περιβάλλον

Οι Αισθητήρες παρέχουν πληροφορίες αντί να εκτελούν από μόνοι τους μια ενέργεια. Οι τιμές τους χρησιμοποιούνται μέσα σε συνθήκες, υπολογισμούς και μεταβλητές. Έτσι ένα έργο μπορεί να αντιδρά σε πλήκτρα, ποντίκι, επαφές, χρώματα, χρόνο και απαντήσεις χρήστη.

1. Συνηθισμένες πηγές εισόδου

Αισθητήρας	Παράδειγμα χρήσης
πατήθηκε πλήκτρο ...;	Χειρισμός παιχνιδιού.
ποντίκι x / ποντίκι y	Ακολουθήση ή στόχευση.
αγγίζει ...;	Συγκρούσεις και ζώνες ενεργοποίησης.
αγγίζει χρώμα ...;	Λαβύρινθος ή έλεγχος ορίων.
απόσταση από ...	Συμπεριφορά με βάση εγγύτητα.
χρονόμετρο	Χρονικά όρια ή μετρήσεις διάρκειας.
ρώτησε ... και περίμενε / απάντηση	Είσοδος κειμένου ή αριθμών από τον χρήστη.

ΚΕΦΑΛΑΙΟ 7 / ΕΙΣΟΔΟΣ ΧΡΗΣΤΗ

Ερωτήσεις, απάντηση και αποθήκευση

Η εντολή «ρώτησε ... και περίμενε» εμφανίζει ερώτηση και σταματά το συγκεκριμένο σενάριο μέχρι ο χρήστης να απαντήσει. Η τιμή βρίσκεται στον αισθητήρα «απάντηση». Επειδή η επόμενη ερώτηση αντικαθιστά αυτή την τιμή, πρέπει να την αντιγράφουμε αμέσως στη σωστή μεταβλητή.

1. Σωστό πρότυπο

1. Ρώτησε «Δώσε τον πρώτο αριθμό».
2. Όρισε α = απάντηση.
3. Ρώτησε «Δώσε τον δεύτερο αριθμό».
4. Όρισε β = απάντηση.
5. Εκτέλεσε τον υπολογισμό με α και β.

2. Λανθασμένο πρότυπο

Αν γίνουν δύο ερωτήσεις στη σειρά και μετά αποθηκεύσουμε και τις δύο τιμές, η «απάντηση» περιέχει μόνο τη δεύτερη απάντηση. Αυτό οδηγεί σε δύο ίδιες τιμές και είναι δύσκολο να εντοπιστεί αν δεν γίνουν κατάλληλες δοκιμές.

Κανόνας: Μετά από κάθε «ρώτησε ... και περίμενε», αποθηκεύστε αμέσως την απάντηση πριν υποβληθεί επόμενη ερώτηση.

ΚΕΦΑΛΑΙΟ 7 / ΕΓΚΥΡΟΤΗΤΑ

Έλεγχος εισόδου πριν από τον υπολογισμό

Η είσοδος χρήστη είναι εξωτερικό δεδομένο και δεν πρέπει να θεωρείται αυτόματα έγκυρη. Ο έλεγχος γίνεται πριν αυξηθούν μετρητές, πριν αλλαχθεί η κατάσταση του έργου και πριν εκτελεστεί υπολογισμός που απαιτεί συγκεκριμένο τύπο τιμής.

1. Τρία επίπεδα ελέγχου

Επίπεδο	Παράδειγμα
Μορφή	Είναι αριθμός; Είναι ακέραιος; Έχει ακριβώς 4 χαρακτήρες;
Όρια	Βρίσκεται από 1 έως 100; Είναι μη αρνητικό;
Σημασιολογικός περιορισμός	Ο διαιρέτης δεν είναι 0; Η ποσότητα είναι πολλαπλάσιο του 10;

2. Επανάληψη μέχρι αποδοχή

Ένα πρακτικό μοτίβο είναι: θέσε «έγκυρο = 0», επανάλαβε ώσπου «έγκυρο = 1», ρώτησε, έλεγξε και μόνο όταν περάσουν όλοι οι έλεγχοι αποθήκευσε την αποδεκτή τιμή. Η μη αποδεκτή είσοδος ζητείται ξανά χωρίς να προχωρά η κύρια λογική.

Σειρά ενεργειών: Έλεγχος → αποδοχή → αποθήκευση → υπολογισμός.

ΚΕΦΑΛΑΙΟ 7 / ΧΡΟΝΟΣ ΚΑΙ ΠΕΡΙΒΑΛΛΟΝ

Χρονόμετρο, ένταση ήχου και ειδικές είσοδοι

Ο αισθητήρας χρονόμετρου είναι χρήσιμος όταν θέλουμε χρόνο που εξελίσσεται ανεξάρτητα από έναν συγκεκριμένο βρόχο. Μπορεί να μηδενιστεί και να χρησιμοποιηθεί για χρονικά όρια, καταγραφή επίδοσης ή εμφάνιση διάρκειας.

1. Μετρητής βρόχου ή χρονόμετρο;

Το «άλλαξε χρόνο κατά 1» μέσα σε βρόχο δεν ισοδυναμεί αναγκαστικά με ένα δευτερόλεπτο· η ταχύτητα εκτέλεσης εξαρτάται από το έργο. Για πραγματικό χρόνο προτιμήστε χρονόμετρο ή εντολές αναμονής με συνειδητή σχεδίαση.

2. Ειδικοί αισθητήρες

Ανάλογα με το περιβάλλον εκτέλεσης μπορούν να χρησιμοποιηθούν ένταση μικροφώνου, πληροφορίες ημερομηνίας/ώρας ή όνομα χρήστη. Σε εργαστηριακές ασκήσεις χρησιμοποιούμε τέτοιες πληροφορίες μόνο όταν είναι απαραίτητες και αποφεύγουμε να συλλέγουμε προσωπικά δεδομένα χωρίς λόγο.

ΚΕΦΑΛΑΙΟ 8 / ΤΕΛΕΣΤΕΣ

Αριθμητικοί υπολογισμοί

Οι Τελεστές είναι πλακίδια τιμών: τοποθετούνται μέσα σε άλλες εντολές και παράγουν αριθμό, κείμενο ή λογική τιμή. Για σύνθετο υπολογισμό είναι χρήσιμο να σπάμε τη μαθηματική έκφραση σε ενδιάμεσες μεταβλητές, ιδιαίτερα όταν θέλουμε να ελέγξουμε το αποτέλεσμα.

Ομάδα	Ενδεικτικές λειτουργίες
Βασική αριθμητική	+, -, ×, ÷
Σύγκριση	<, =, >
Λογική	και, ή, όχι
Ακέραιο υπόλοιπο	mod / ακέρ. υπόλ. ... δια ...
Στρογγυλοποίηση	στρογγυλοποίηση
Μαθηματικές συναρτήσεις	τετρ. ρίζα, απόλυτη τιμή, ημ, συν, λογάριθμοι κ.ά.

1. Παρενθέσεις με τη δομή των πλακιδίων

Στο Scratch οι εμφωλευμένες θέσεις των πλακιδίων λειτουργούν όπως οι παρενθέσεις σε έναν μαθηματικό τύπο. Για παράδειγμα, το $(\alpha + \beta) / 2$ απαιτεί πρώτα το άθροισμα να τοποθετηθεί μέσα στον αριθμητή της διαίρεσης.

Πρακτικός έλεγχος: Υπολογίστε χειροκίνητα ένα μικρό παράδειγμα και συγκρίνετέ το με το Scratch. Αν διαφέρει, ελέγξτε πρώτα τη δομή της αριθμητικής έκφρασης.

Τυχαίοι αριθμοί και ελεγχόμενες δοκιμές

Η εντολή τυχαίας επιλογής είναι χρήσιμη για παιχνίδια, προσομοιώσεις και διαφορετικές ερωτήσεις. Η τυχαιότητα όμως δυσκολεύει τη δοκιμή, επειδή κάθε εκτέλεση μπορεί να ακολουθεί διαφορετική διαδρομή.

1. Δοκιμή με γνωστή τιμή

Κατά την ανάπτυξη ενός παιχνιδιού μαντεψιάς μπορείτε προσωρινά να ορίσετε «μυστικός = 73» ώστε να ελέγξετε με συγκεκριμένες εισόδους την κατεύθυνση των υποδείξεων και τον μετρητή. Πριν από την τελική έκδοση επαναφέρετε την τυχαία επιλογή.

2. Ακέραιες και δεκαδικές τιμές

Όταν τα όρια της τυχαίας επιλογής είναι ακέραια, χρησιμοποιείται συχνά για τυχαίους ακεραίους. Αν το πρόβλημα απαιτεί ειδική κατανομή ή δεκαδικά βήματα, είναι προτιμότερο να παράγουμε ακέραιη τυχαία τιμή και να τη μετασχηματίζουμε με σαφή τρόπο.

ΚΕΦΑΛΑΙΟ 8 / ΚΕΙΜΕΝΟ

Συμβολοσειρές και επεξεργασία χαρακτήρων

Το Scratch χειρίζεται κείμενο με τελεστές όπως «ένωσε», «γράμμα ... του ...», «μήκος του» και «... περιλαμβάνει ...». Αυτοί οι τελεστές είναι αρκετοί για πολλούς ελέγχους μορφής, κωδικούς, ονόματα, απλά πρωτόκολλα μηνυμάτων και επεξεργασία χαρακτήρων.

Ενέργεια	Παράδειγμα
Συνένωση	ένωσε «Σκορ: » με σκορ
Χαρακτήρας σε θέση	γράμμα 3 του κειμένου
Μήκος	μήκος του κειμένου
Έλεγχος περιεχομένου	«0123456789» περιλαμβάνει χαρακτήρα;

1. Αρχικά μηδενικά

Σε μια αριθμομηχανή, το σύμβολο της πράξης είναι κείμενο, ενώ οι δύο τιμές που συμμετέχουν στον υπολογισμό είναι αριθμοί. Ξεχωρίζουμε λοιπόν τι αποτελεί μήνυμα ή επιλογή και τι πρέπει να μετατραπεί σε αριθμητική τιμή.

Διάκριση: Ένας αριθμός αναπαριστά ποσότητα. Ένα μήνυμα, ένα σύμβολο πράξης ή μια ετικέτα μπορεί να περιέχει ψηφία ή σύμβολα αλλά να παραμένει κείμενο.

ΚΕΦΑΛΑΙΟ 8 / ΓΕΩΜΕΤΡΙΑ

Τριγωνομετρικές συναρτήσεις και γωνίες

Οι συναρτήσεις ημίτονο και συνημίτονο του Scratch χρησιμοποιούν μοίρες. Αυτό επιτρέπει να υπολογίζουμε σημεία κύκλων και άλλων σχημάτων χωρίς να προσεγγίζουμε την ακτίνα με μήκος βήματος.

1. Παραμετρικό σημείο κύκλου

Για κύκλο με κέντρο (cx, cy), ακτίνα r και γωνία θ μπορούμε να χρησιμοποιήσουμε:

Τύποι: $x = cx + r \cdot \sin(\theta)$ και $y = cy + r \cdot \cos(\theta)$.

Αυξάνοντας τη γωνία από 0° έως 360° και μετακινώντας το αντικείμενο στα διαδοχικά σημεία, η Πένα σχεδιάζει μια πολύ καλή προσέγγιση της περιφέρειας.

2. Εξωτερική γωνία κανονικού πολυγώνου

Για κανονικό πολύγωνο με n πλευρές, η στροφή μετά από κάθε πλευρά είναι $360^\circ / n$. Για τετράγωνο έχουμε 90° , για ισόπλευρο τρίγωνο 120° και για εξάγωνο 60° .

ΚΕΦΑΛΑΙΟ 9 / ΔΕΔΟΜΕΝΑ

Μεταβλητές και κατάσταση προγράμματος

Μεταβλητές, λίστες και δεδομένα

Μεταβλητή = μία τιμή

όρισε σκορ σε 0

άλλαξε σκορ κατά 1

σκορ = 12

Λίστα = πολλές τιμές

1. Άννα

2. Μάριος

3. Ελένη

4. Νίκος

Οι μεταβλητές περιγράφουν την τρέχουσα κατάσταση, ενώ οι λίστες αποθηκεύουν ακολουθίες τιμών.

1. Τι είναι μεταβλητή

Μεταβλητή είναι ένα όνομα που αντιστοιχεί σε τιμή η οποία μπορεί να αλλάζει κατά την εκτέλεση. Σκορ, προσπάθειες, θέση σε ερώτηση, κάτω όριο, ταχύτητα και κατάσταση παιχνιδιού είναι παραδείγματα διαφορετικών ρόλων μεταβλητής.

2. Αρχικοποίηση

Κάθε μεταβλητή που επηρεάζει το αποτέλεσμα πρέπει να αποκτά γνωστή αρχική τιμή. Η σωστή αρχικοποίηση αποτρέπει «παλιές» τιμές από προηγούμενη εκτέλεση και κάνει τις δοκιμές επαναλήψιμες.

Ονόματα: Προτιμήστε «προσπάθειες», «τρέχουσα ερώτηση», «μέγιστο σκορ» αντί για x , $a1$, $temp2$, εκτός αν πρόκειται πράγματι για συντεταγμένη ή πολύ τοπική βοηθητική τιμή.

ΚΕΦΑΛΑΙΟ 9 / ΕΜΒΕΛΕΙΑ

Μεταβλητές για όλα ή μόνο για ένα αντικείμενο

Κατά τη δημιουργία μιας μεταβλητής επιλέγουμε αν θα είναι διαθέσιμη «για όλα τα αντικείμενα» ή μόνο για το επιλεγμένο αντικείμενο. Η επιλογή καθορίζει ποιος μπορεί να τη χρησιμοποιεί και, στους κλώνους, αν η τιμή είναι κοινή ή ξεχωριστή.

Τύπος	Κατάλληλη χρήση
Για όλα τα αντικείμενα	Κοινό σκορ, κατάσταση παιχνιδιού, κοινή σημαία συγχρονισμού.

Τύπος	Κατάλληλη χρήση
Μόνο για αυτό το αντικείμενο	Εσωτερική θέση/κατάσταση ενός sprite, ιδιότητα που δεν χρειάζεται να γνωρίζουν τα άλλα.

1. Περιορίστε το κοινό κράτος

Όσο περισσότερα αντικείμενα αλλάζουν μια κοινή μεταβλητή, τόσο δυσκολότερο είναι να βρούμε ποιος δημιούργησε λάθος τιμή. Κρατήστε τις εσωτερικές βοηθητικές μεταβλητές τοπικές όταν δεν υπάρχει λόγος να είναι κοινές.

2. Εμφάνιση στη σκηνή

Η προβολή μιας μεταβλητής είναι χρήσιμη για σκορ, χρόνο ή προσωρινή αποσφαλμάτωση. Οι βοηθητικές μεταβλητές συνήθως κρύβονται στην τελική έκδοση ώστε ο χρήστης να βλέπει μόνο πληροφορίες που έχουν νόημα.

ΚΕΦΑΛΑΙΟ 9 / ΠΡΟΤΥΠΑ

Μετρητές, αθροιστές και ακρότατα

Πολλά προβλήματα χρησιμοποιούν επαναλαμβανόμενα πρότυπα ενημέρωσης μεταβλητών. Η αναγνώριση του προτύπου βοηθά να σχεδιάζουμε γρηγορότερα και να ελέγχουμε πιο συστηματικά τον κώδικα.

Πρότυπο	Αρχική τιμή	Ενημέρωση
Μετρητής	0	άλλαξε μετρητή κατά 1
Αθροιστής	0	άλλαξε άθροισμα κατά νέα τιμή
Γινόμενο	1	όρισε γινόμενο = γινόμενο × νέα τιμή
Ελάχιστο	Πρώτη έγκυρη τιμή	αν νέα < ελάχιστο, όρισε ελάχιστο = νέα
Μέγιστο	Πρώτη έγκυρη τιμή	αν νέα > μέγιστο, όρισε μέγιστο = νέα

Προσοχή: Η αρχική τιμή για ελάχιστο ή μέγιστο πρέπει να επιλέγεται με τρόπο συμβατό με τα δεδομένα. Συχνά η ασφαλέστερη λύση είναι να χρησιμοποιείται η πρώτη έγκυρη τιμή.

ΚΕΦΑΛΑΙΟ 9 / ΛΙΣΤΕΣ

Ακολουθίες δεδομένων

Η λίστα αποθηκεύει πολλές τιμές με σειρά. Κάθε στοιχείο έχει θέση, με πρώτη θέση το 1. Μπορούμε να προσθέτουμε, να διαγράφουμε, να αντικαθιστούμε και να διαβάζουμε στοιχεία, καθώς και να ελέγχουμε το μήκος της λίστας.

1. Παραδείγματα χρήσης

- Ιστορικό έγκυρων μαντεψιών στο παιχνίδι «Μάντεψε τον αριθμό».
- Ονόματα παικτών και βαθμολογίες.
- Ιστορικό μετρήσεων ή τιμών.
- Κατάλογος αντικειμένων που απομένουν να εμφανιστούν.

2. Παράλληλες λίστες

Αν η ερώτηση στη θέση i αντιστοιχεί στην απάντηση στη θέση i , δύο λίστες μπορούν να λειτουργούν ως απλός πίνακας εγγραφών. Η ίδια μεταβλητή «θέση» χρησιμοποιείται για ανάγνωση και από τις δύο.

Δείκτες: Στο Scratch η πρώτη θέση μιας λίστας είναι 1. Ελέγχετε πάντα ότι ο δείκτης βρίσκεται από 1 έως «μήκος λίστας».

ΚΕΦΑΛΑΙΟ 9 / ΛΙΣΤΕΣ

Διάσχιση, αναζήτηση και ενημέρωση

1. Διάσχιση όλων των στοιχείων

1. Θέσε θέση = 1.
2. Επανάλαβε όσο θέση $\leq$ μήκος λίστας.
3. Χρησιμοποίησε το στοιχείο στη θέση.
4. Άλλαξε θέση κατά 1.

Αυτό το μοτίβο είναι η βάση για άθροισμα λίστας, αναζήτηση, μέτρηση στοιχείων που ικανοποιούν μια συνθήκη ή μαζική μετατροπή.

2. Αναζήτηση με πρόωρη διακοπή

Αν θέλουμε να βρούμε μόνο αν υπάρχει μια τιμή, μπορούμε να σταματήσουμε όταν βρεθεί. Χρησιμοποιούμε μεταβλητή «βρέθηκε» ή συνθήκη τερματισμού ώστε να μην εξετάζονται άσκοπα τα υπόλοιπα στοιχεία.

3. Μεταβολή λίστας μέσα σε βρόχο

Όταν διαγράφονται στοιχεία ενώ διατρέχουμε τη λίστα, οι επόμενες θέσεις μετακινούνται. Χρειάζεται προσοχή ώστε να μη «χαθεί» το στοιχείο που μετακινήθηκε στη θέση που μόλις εξετάσαμε.

ΚΕΦΑΛΑΙΟ 9 / ΔΕΔΟΜΕΝΑ

Ανθεκτική είσοδος και συνέπεια δεδομένων

Η ποιότητα ενός προγράμματος φαίνεται στις οριακές και μη αποδεκτές εισόδους. Η βασική λογική δεν πρέπει να ξεκινά μέχρι να είναι βέβαιο ότι τα δεδομένα ικανοποιούν τις προδιαγραφές.

1. Σειρά ελέγχου

1. Έλεγχος κενής ή μη κατάλληλης μορφής.
2. Έλεγχος ακεραιότητας ή μήκους, όταν απαιτείται.
3. Έλεγχος αριθμητικών ορίων.
4. Έλεγχος ειδικού περιορισμού του προβλήματος.
5. Αποθήκευση και χρήση της τιμής.

2. Μην αλλάζετε πραγματικά δεδομένα όσο δοκιμάζετε υποψήφιες λύσεις

Σε έναν υπολογισμό, όπως η επίλυση εξίσωσης ή η αριθμομηχανή, κρατήστε ξεχωριστές βοηθητικές μεταβλητές για τα ενδιάμεσα αποτελέσματα. Η τελική μεταβλητή αποτελέσματος ενημερώνεται μόνο όταν έχουν ολοκληρωθεί οι απαραίτητοι έλεγχοι.

Αρχή ασφαλούς υπολογισμού: έλεγξε πρώτα τις προϋποθέσεις, εκτέλεσε τον υπολογισμό και μετά εμφάνισε ή αποθήκευσε το αποτέλεσμα.

Σύντομη αναφορά σε μεταβλητές νέφους

Σε υποστηριζόμενα διαδικτυακά έργα, οι μεταβλητές νέφους μπορούν να αποθηκεύουν αριθμητικές τιμές στον εξυπηρετητή και να είναι κοινές μεταξύ εκτελέσεων. Δεν είναι απαραίτητες για τις βασικές εργαστηριακές ασκήσεις και έχουν διαφορετικούς περιορισμούς από τις τοπικές μεταβλητές.

1. Πότε έχουν νόημα

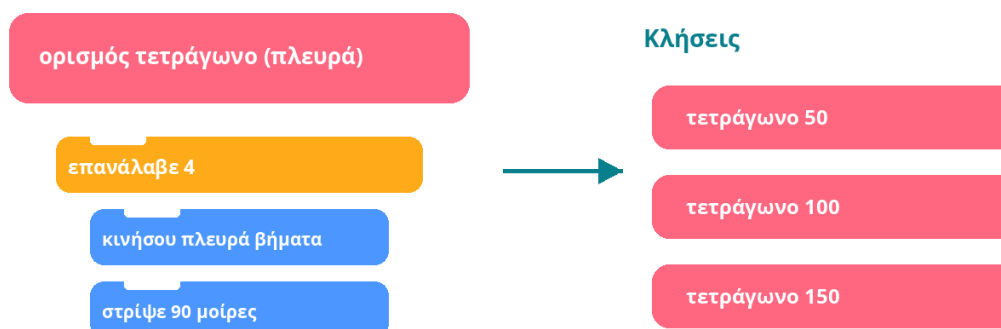
- Κοινός αριθμητικός μετρητής ή απλός πίνακας επιδόσεων.
- Έργο που έχει σχεδιαστεί ρητά για διαδικτυακή χρήση και λαμβάνει υπόψη ταυτόχρονες ενημερώσεις.

2. Τι να αποφεύγετε

Μην αποθηκεύετε προσωπικά ή ευαίσθητα δεδομένα. Μην βασίζετε μια τοπική εργαστηριακή άσκηση σε λειτουργία νέφους όταν δεν είναι απαραίτητη. Για τη βασική εκμάθηση μεταβλητών, οι κανονικές μεταβλητές είναι απλούστερες και προβλέψιμες.

Διαδικασίες: δίνουμε όνομα σε μια λειτουργία

Δικές μας εντολές και παράμετροι



Μια δική μας εντολή συγκεντρώνει ένα επαναχρησιμοποιήσιμο τμήμα λογικής και μπορεί να δέχεται παραμέτρους.

Οι «Εντολές μου» επιτρέπουν να δημιουργούμε διαδικασίες. Αντί να επαναλαμβάνουμε ένα μεγάλο σύνολο πλακιδίων, του δίνουμε όνομα και το καλούμε όπου χρειάζεται. Αυτό βελτιώνει την αναγνωσιμότητα, την επαναχρησιμοποίηση και τη συντήρηση.

1. Πότε δημιουργούμε διαδικασία

- Όταν το ίδιο μοτίβο εντολών επαναλαμβάνεται.
- Όταν ένα μεγάλο σενάριο περιέχει σαφές υποέργο, π.χ. «Έλεγε είσοδο».
- Όταν θέλουμε να δοκιμάσουμε μια λειτουργία ανεξάρτητα.
- Όταν μια έννοια έχει φυσικό όνομα, π.χ. «Σχεδίασε πολύγωνο».

Στόχος: Το κύριο σενάριο να διαβάζεται ως περιγραφή της λύσης και οι λεπτομέρειες να βρίσκονται σε μικρότερες διαδικασίες.

Μία διαδικασία, πολλές τιμές

Μια παράμετρος είναι τιμή που δίνεται στη διαδικασία κατά την κλήση. Για παράδειγμα, αντί για «Τετράγωνο50» και «Τετράγωνο100» δημιουργούμε «Τετράγωνο (πλευρά)» και περνάμε 50 ή 100 ανάλογα με την ανάγκη.

1. Καλά ονόματα παραμέτρων

Ασαφές	Περιγραφικό
x	πλευρά
n	πλήθος πλευρών
v	ταχύτητα
a	ελάχιστο
b	μέγιστο

2. Παράμετρος δεν είναι κοινή μεταβλητή

Η παράμετρος υπάρχει ως είσοδος της συγκεκριμένης κλήσης και εμφανίζεται ως ροζ πλακίδιο μέσα στον ορισμό. Δεν χρειάζεται να δημιουργήσουμε ξεχωριστή πορτοκαλί μεταβλητή μόνο και μόνο για να μεταφέρουμε την τιμή.

3. Γενίκευση

Όταν μια σταθερά αντικαθίσταται από παράμετρο, η διαδικασία γίνεται πιο γενική. Αυτή είναι βασική αρχή προγραμματισμού: απομονώνουμε ό,τι αλλάζει από τη λογική που παραμένει ίδια.

Αποσύνθεση ενός προβλήματος σε διαδικασίες

Ένα μεγάλο έργο γίνεται ευκολότερο όταν χωριστεί σε μικρές λειτουργίες με σαφείς ευθύνες. Η διάσπαση πρέπει να ακολουθεί τη λογική του προβλήματος και όχι απλώς να δημιουργεί πολλές εντολές χωρίς λόγο.

1. Ενδεικτική ροή

1. Αρχικοποίησε έργο.
2. Διάβασε και έλεγξε στοιχεία.
3. Εκτέλεσε τον βασικό υπολογισμό ή γύρο.
4. Εμφάνισε αποτέλεσμα.
5. Ολοκλήρωσε ή επανάφερε την εφαρμογή σε ασφαλή κατάσταση.

2. Παράδειγμα ονομάτων

Κύριο έργο	Υποδιαδικασίες
Μάντεψε τον αριθμό	Διάβασε μαντεψιά, Έλεγε όρια, Δώσε υπόδειξη, Εμφάνισε προσπάθειες
Επίλυση εξίσωσης	Διάβασε συντελεστές, Υπολόγισε διακρίνουσα, Έλεγε περίπτωση, Εμφάνισε ρίζες
Συστάδες κύκλων	Σχεδίασε κύκλο, Μετακινήσου χωρίς γραμμή, Σχεδίασε συστάδα

«Εκτέλεση χωρίς ανανέωση οθόνης»

Κατά τη δημιουργία μιας δικής μας εντολής υπάρχει επιλογή εκτέλεσης χωρίς ανανέωση οθόνης. Όταν είναι ενεργή, το Scratch προσπαθεί να εκτελέσει τη διαδικασία πολύ γρήγορα πριν ανανεώσει την εμφάνιση. Είναι χρήσιμη σε ορισμένους καθαρά υπολογιστικούς βρόχους, αλλά δεν ταιριάζει σε κάθε διαδικασία.

1. Κατάλληλες περιπτώσεις

- Γρήγορος υπολογισμός πολλών ενδιάμεσων τιμών.
- Γεωμετρική διαδικασία όπου θέλουμε να εμφανιστεί άμεσα το τελικό σχέδιο, εφόσον δεν περιλαμβάνει αναμονές ή αλληλεπίδραση.

2. Μη κατάλληλες περιπτώσεις

- Διαδικασίες που περιλαμβάνουν «ρώτησε ... και περίμενε».
- Διαδικασίες με σκόπιμες αναμονές, αφήγηση ή animation που πρέπει να είναι ορατό βήμα προς βήμα.

Προσοχή: Μην ενεργοποιείτε την επιλογή από συνήθεια. Χρησιμοποιήστε την μόνο όταν γνωρίζετε γιατί βελτιώνει τη συγκεκριμένη διαδικασία.

Δοκιμή και συντήρηση διαδικασιών

1. Δοκιμάστε μικρά κομμάτια

Μια διαδικασία είναι ευκολότερο να ελεγχθεί όταν έχει σαφή είσοδο και αναμενόμενο αποτέλεσμα. Για παράδειγμα, δοκιμάστε «Τετράγωνο (πλευρά)» με 20, 50 και 100 πριν τη χρησιμοποιήσετε μέσα σε μεγαλύτερη σύνθεση.

2. Μην κρύβετε σφάλματα μέσα σε «μαύρο κουτί»

Η χρήση διαδικασιών δεν σημαίνει ότι αγνοούμε το εσωτερικό τους. Όταν υπάρχει σφάλμα, ελέγχουμε πρώτα τις παραμέτρους που δόθηκαν και μετά την εσωτερική λογική της διαδικασίας.

3. Μικρές και συνεκτικές ευθύνες

Μια εντολή με όνομα «Κάνε τα πάντα» που διαβάζει δεδομένα, υπολογίζει, αλλάζει υπόβαθρο, παίζει ήχο και διαγράφει κλώνους είναι δύσκολη στη συντήρηση. Προτιμήστε λίγες, σαφείς ευθύνες ανά διαδικασία.

Έλεγχος ονόματος: Αν δεν μπορείτε να δώσετε στη διαδικασία ένα σύντομο και σαφές ρήμα με αντικείμενο, ίσως εκτελεί περισσότερα από όσα πρέπει.

Σχεδίαση με κίνηση αντικειμένου

Η Πένα μετατρέπει την κίνηση σε σχεδίαση

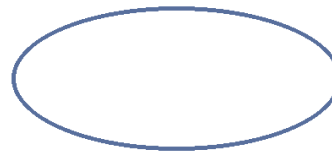
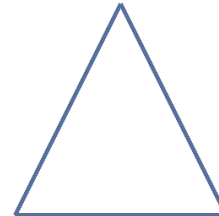
καθάρισε όλα

κατέβασε πένα

κινήσου 100 βήματα

στρίψε 90 μοίρες

σήκωσε πένα



Η Πένα μετατρέπει τη διαδρομή ενός αντικειμένου σε γραμμή επάνω στη σκηνή.

Η επέκταση «Πένα» επιτρέπει στο sprite να αφήνει ίχνος καθώς κινείται. Η γεωμετρία του σχεδίου προκύπτει από τον συνδυασμό θέσης, κατεύθυνσης, κίνησης και κατάστασης της πέννας.

1. Βασικές εντολές

Εντολή / ιδέα	Λειτουργία
καθάρισε όλα	Διαγράφει τα ίχνη και τις σφραγίδες της Πέννας.
κατέβασε πένα	Οι επόμενες μετακινήσεις σχεδιάζουν.
σήκωσε πένα	Οι επόμενες μετακινήσεις δεν αφήνουν γραμμή.
όρισε/άλλαξε μέγεθος πέννας	Ρυθμίζει το πάχος.
όρισε/άλλαξε χρώμα πέννας	Ρυθμίζει την εμφάνιση της γραμμής.
σφραγίδα	Αποτυπώνει την τρέχουσα εμφάνιση του sprite στη σκηνή.

2. Αρχική κατάσταση

Στην πράσινη σημαία είναι καλή πρακτική να καθαρίζουμε τη σκηνή, να σηκώνουμε την πένα, να τοποθετούμε το αντικείμενο στην αφετηρία και μόνο μετά να ξεκινά η σχεδίαση.

ΚΕΦΑΛΑΙΟ 11 / ΠΟΛΥΓΩΝΑ

Κανονικά σχήματα με επανάληψη

Για κανονικό πολύγωνο με n πλευρές ίσου μήκους, επαναλαμβάνουμε n φορές: κίνηση κατά το μήκος της πλευράς και στροφή κατά την εξωτερική γωνία $360^\circ/n$. Μετά την ολοκλήρωση, το αντικείμενο επιστρέφει στην αρχική κατεύθυνση.

Σχήμα	Πλευρές n	Στροφή $360^\circ/n$
Ισόπλευρο τρίγωνο	3	120°
Τετράγωνο	4	90°
Κανονικό πεντάγωνο	5	72°
Κανονικό εξαγώνο	6	60°
Κανονικό δωδεκάγωνο	12	30°

1. Παραμετρική διαδικασία

Μια γενική διαδικασία «Πολύγωνο (πλευρές) (μήκος)» μπορεί να χρησιμοποιεί τις δύο παραμέτρους και να υπολογίζει αυτόματα τη στροφή. Έτσι το ίδιο τμήμα κώδικα σχεδιάζει πολλά διαφορετικά σχήματα.

Γεωμετρική ιδέα: Το άθροισμα των εξωτερικών στροφών σε μία πλήρη περιφορά είναι 360° .

ΚΕΦΑΛΑΙΟ 11 / ΜΕΤΑΚΙΝΗΣΗ

Σχεδιάζουμε μόνο εκεί που πρέπει

Οι ανεπιθύμητες γραμμές εμφανίζονται όταν μετακινούμε το sprite από ένα σχήμα σε άλλο με κατεβασμένη πένα. Η λύση είναι να διαχωρίσουμε τη «μετακίνηση τοποθέτησης» από τη «μετακίνηση σχεδίασης».

1. Ασφαλές πρότυπο μετακίνησης

1. Σήκωσε πένα.
2. Πήγαινε στην επιθυμητή θέση ή μετακινήσου στο επόμενο κέντρο.
3. Ρύθμισε κατεύθυνση, αν χρειάζεται.
4. Κατέβασε πένα μόνο όταν αρχίζει η πραγματική περιφέρεια ή πλευρά.
5. Σήκωσε ξανά την πένα στο τέλος της διαδικασίας.

2. Σύμβαση αρχής και τέλους

Μια καλοσχεδιασμένη διαδικασία σχεδίασης καθορίζει πού αφήνει το αντικείμενο όταν τελειώσει. Για παράδειγμα, η διαδικασία «κύκλος (ακτίνα)» μπορεί να επιστρέφει στο αρχικό κέντρο με σηκωμένη πένα. Αυτό κάνει ασφαλή την επαναχρησιμοποίησή της.

Έλεγχος: Εκτελέστε τη διαδικασία δύο φορές από διαφορετικές θέσεις. Αν η δεύτερη σχεδίαση εξαρτάται από «τυχαία» τελική θέση της πρώτης, χρειάζεται σαφέστερη σύμβαση.

ΚΕΦΑΛΑΙΟ 11 / ΚΥΚΛΟΣ

Κύκλος με πραγματική παράμετρο ακτίνας

Η επανάληψη «κινήσου 1 βήμα, στρίψε 1° » δημιουργεί μια περίπου κυκλική διαδρομή, αλλά το 1 δεν είναι ακτίνα. Αν η παράμετρος πρέπει να σημαίνει πραγματική ακτίνα σε βήματα της σκηνής, χρησιμοποιούμε παραμετρικές συντεταγμένες.

1. Αλγόριθμος

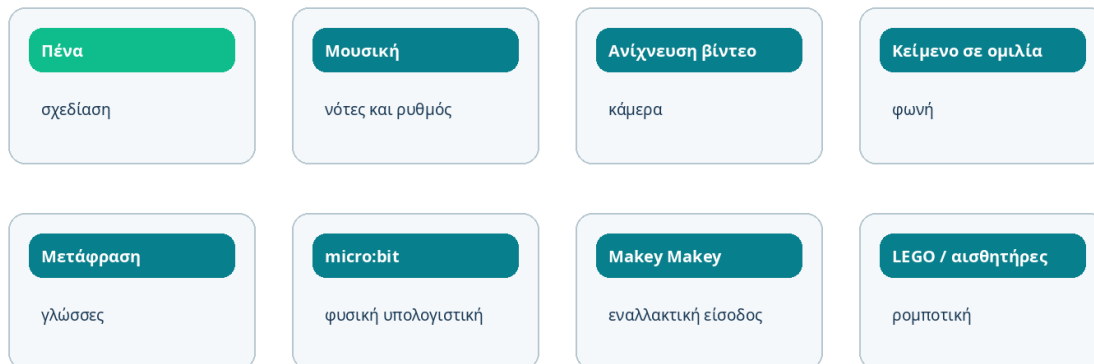
1. Αποθήκευσε το κέντρο $cx = \text{θέση } x$ και $cy = \text{θέση } y$.
2. Με σηκωμένη πένα πήγαινε στο σημείο $\theta = 0^\circ$.
3. Κατέβασε πένα.
4. Για θ από 1° έως 360° , πήγαινε στο $(cx + r \cdot \sin\theta, cy + r \cdot \cos\theta)$.
5. Σήκωσε πένα και επέστρεψε στο (cx, cy) , αν αυτή είναι η σύμβαση της διαδικασίας.

2. Ακρίβεια και κόστος

Τα 360 τμήματα δίνουν ομαλό αποτέλεσμα και είναι εύκολα στην κατανόηση. Για πολύ σύνθετα σχέδια μπορεί να μειωθεί το πλήθος των σημείων, με μικρή απώλεια ομαλότητας, ώστε να μειωθεί ο χρόνος σχεδίασης.

Προσθέτοντας νέες δυνατότητες

Επεκτάσεις: πρόσθετες δυνατότητες



Οι επεκτάσεις προσθέτουν νέες κατηγορίες πλακιδίων για λογισμικό, αισθητήρες ή εξωτερικές συσκευές. Από το κουμπί προσθήκης επέκτασης μπορούμε να ενεργοποιήσουμε επιπλέον κατηγορίες εντολών. Η ακριβής λίστα μπορεί να διαφέρει ανά έκδοση ή περιβάλλον, αλλά συνηθισμένα παραδείγματα είναι Πένα, Μουσική, Ανίχνευση Βίντεο, Μετατροπή Κειμένου σε Ομιλία, Μετάφραση και συνδέσεις με εκπαιδευτικό υλικό ή συσκευές.

1. Πότε χρησιμοποιούμε επέκταση

- Όταν η λειτουργία δεν ανήκει στον βασικό πυρήνα του Scratch.
- Όταν θέλουμε να συνδέσουμε έργο με κάμερα, ήχο, μουσική ή εκπαιδευτικό υλικό.
- Όταν μια εξωτερική συσκευή παρέχει αισθητήρες ή ενεργοποιητές που πρέπει να ενταχθούν στη λογική του έργου.

Αρχή σχεδίασης: Η επέκταση πρέπει να εξυπηρετεί τη λύση και όχι να προστίθεται μόνο επειδή είναι διαθέσιμη.

ΚΕΦΑΛΑΙΟ 12 / ΦΥΣΙΚΗ ΥΠΟΛΟΓΙΣΤΙΚΗ

Συσκευές, αισθητήρες και Scratch Link

Ορισμένες επεκτάσεις επικοινωνούν με εξωτερικές συσκευές ή εκπαιδευτικά κιτ. Ανάλογα με τη συσκευή και το λειτουργικό σύστημα μπορεί να απαιτείται υποστηρικτικό λογισμικό σύνδεσης, όπως το Scratch Link, και κατάλληλη ασύρματη ή ενσύρματη σύνδεση.

1. Μοντέλο εισόδου–επεξεργασίας–εξόδου

Στάδιο	Παράδειγμα
Είσοδος	Ανάγνωση πλήκτρου, κλίσης, απόστασης ή άλλου αισθητήρα.
Επεξεργασία	Έλεγχος ορίου, υπολογισμός, επιλογή κατάστασης.
Έξοδος	Κίνηση sprite, ήχος, μήνυμα ή εντολή προς ενεργοποιητή.

2. Απομόνωση της συσκευής

Σε μεγαλύτερο έργο είναι χρήσιμο οι εντολές που διαβάζουν τη συσκευή να βρίσκονται σε λίγες διαδικασίες. Το υπόλοιπο πρόγραμμα χρησιμοποιεί μεταβλητές όπως «θερμοκρασία», «κουμπί πατημένο» ή «κλίση x» και δεν εξαρτάται από τις λεπτομέρειες σύνδεσης.

ΚΕΦΑΛΑΙΟ 12 / ΑΞΙΟΠΙΣΤΙΑ

Σχεδίαση όταν η είσοδος δεν είναι τέλεια

Οι πραγματικοί αισθητήρες μπορεί να έχουν θόρυβο, καθυστέρηση ή τιμές που μεταβάλλονται γρήγορα γύρω από ένα όριο. Ένα πρόγραμμα που αντιδρά σε κάθε πολύ μικρή μεταβολή μπορεί να φαίνεται ασταθές.

1. Απλά μέτρα

- Χρησιμοποιήστε λογικό όριο και όχι υπερβολικά ευαίσθητη σύγκριση.
- Απαιτήστε η συνθήκη να παραμένει αληθής για μικρό χρονικό διάστημα όταν αυτό έχει νόημα.
- Αποθηκεύστε προηγούμενη κατάσταση ώστε να αντιδράτε στην αλλαγή και όχι σε κάθε επανάληψη.
- Προβλέψτε τι θα κάνει το έργο αν η συσκευή αποσυνδεθεί.

2. Προσομοίωση πριν από τη συσκευή

Κατά την ανάπτυξη μπορούμε να αντικαταστήσουμε προσωρινά τον αισθητήρα με πλήκτρα ή μεταβλητές. Έτσι ελέγχουμε πρώτα τη λογική και μετά προσθέτουμε την πραγματική συσκευή ως πηγή εισόδου.

ΚΕΦΑΛΑΙΟ 13 / ΑΠΟΣΦΑΛΜΑΤΩΣΗ

Από το σύμπτωμα στην αιτία

Κύκλος εντοπισμού σφάλματος



Η αποσφαλμάτωση είναι κύκλος παρατήρησης, υπόθεσης, μικρής αλλαγής και νέας δοκιμής. Σφάλμα δεν σημαίνει απαραίτητα ότι «δεν τρέχει τίποτα». Συχνά το έργο εκτελείται αλλά παράγει λάθος τιμή, ακολουθεί λάθος κλάδο ή διατηρεί παλιά κατάσταση. Η συστηματική αποσφαλμάτωση μειώνει τον χρόνο που χάνεται σε τυχαίες αλλαγές.

1. Κύκλος αποσφαλμάτωσης

1. Αναπαρήγαγε το πρόβλημα με συγκεκριμένη δοκιμή.

2. Κατέγραψε τι περίμενες και τι συνέβη.
3. Εντόπισε το μικρότερο τμήμα κώδικα που μπορεί να ευθύνεται.
4. Παρατήρησε σχετικές μεταβλητές ή πρόσθεσε προσωρινό μήνυμα.
5. Άλλαξε ένα πράγμα κάθε φορά.
6. Εκτέλεσε ξανά την ίδια δοκιμή και μετά κάνε δοκιμή παλινδρόμησης.

Μην διορθώνετε στα τυφλά: Αν κάνετε πολλές αλλαγές μαζί και το πρόβλημα εξαφανιστεί, δεν θα γνωρίζετε ποια αλλαγή το διόρθωσε, ούτε αν δημιούργησε νέο σφάλμα.

ΚΕΦΑΛΑΙΟ 13 / ΠΑΡΑΤΗΡΗΣΗ

Μεταβλητές, προσωρινά μηνύματα και μικρές δοκιμές

1. Εμφάνιση εσωτερικής κατάστασης

Κατά την ανάπτυξη μπορείτε να εμφανίζετε προσωρινά μεταβλητές όπως «κάτω όριο», «πάνω όριο», «θέση λίστας» ή «βρέθηκε». Αν μια τιμή αλλάζει σε λάθος στιγμή, η αιτία συχνά γίνεται αμέσως ορατή.

2. Προσωρινά μηνύματα

Μια εντολή «πες» μπορεί να δείξει ότι εκτελέστηκε συγκεκριμένος κλάδος ή ότι ελήφθη μήνυμα. Αφαιρέστε τα διαγνωστικά μηνύματα από την τελική έκδοση αν δεν έχουν νόημα για τον χρήστη.

3. Δοκιμή διαδικασίας ανεξάρτητα

Αν το συνολικό έργο είναι μεγάλο, δημιουργήστε προσωρινό μικρό σενάριο που καλεί μόνο τη διαδικασία που ελέγχετε με γνωστές τιμές. Είναι το αντίστοιχο ενός μικρού «πειράματος» μέσα στο έργο.

ΚΕΦΑΛΑΙΟ 13 / ΔΟΚΙΜΕΣ

Κανονικές, οριακές και μη αποδεκτές περιπτώσεις

Ένα πρόγραμμα δεν θεωρείται ελεγμένο επειδή λειτούργησε μία φορά. Χρειαζόμαστε σύνολο δοκιμών που καλύπτει τις διαφορετικές διαδρομές της λογικής.

Κατηγορία	Παράδειγμα
Κανονική	Τυπική τιμή στο μέσο των ορίων.
Κάτω όριο	Η μικρότερη αποδεκτή τιμή.
Άνω όριο	Η μεγαλύτερη αποδεκτή τιμή.
Ακριβώς έξω από όριο	Μία τιμή μικρότερη/μεγαλύτερη από το επιτρεπτό.
Ειδική περίπτωση	0 στον διαιρέτη, αρνητική διακρίνουσα, μη έγκυρη μαντεψιά.
Μη αποδεκτή μορφή	Κείμενο αντί αριθμού, δεκαδικός αντί ακεραίου, λάθος μήκος.

1. Πίνακας αναμενόμενων αποτελεσμάτων

Πριν από την εκτέλεση, γράψτε το αναμενόμενο αποτέλεσμα. Η δοκιμή έχει αξία όταν συγκρίνουμε πραγματικό με αναμενόμενο, όχι όταν απλώς παρατηρούμε αν «φαίνεται εντάξει».

Δέκα προβλήματα που αξίζει να ελέγχετε πρώτα

Σύμπτωμα	Πιθανός έλεγχος
Η δεύτερη εκτέλεση είναι λάθος	Λείπει αρχικοποίηση μεταβλητής/θέσης/λίστας.
Η απάντηση σε δύο ερωτήσεις είναι ίδια	Δεν αποθηκεύτηκε αμέσως η πρώτη «απάντηση».
Δύο ενέργειες ξεκινούν μαζί	Λείπει σημαία «απασχολημένος» ή σωστός συγχρονισμός.
Ο βρόχος δεν τελειώνει	Η συνθήκη τερματισμού δεν μπορεί να αλλάξει.
Το sprite αφήνει άσχετες γραμμές	Η πένα δεν σηκώθηκε πριν από μετακίνηση τοποθέτησης.
Το αντικείμενο περιστρέφεται παράξενα	Λάθος κέντρο ενδυμασίας ή στυλ περιστροφής.
Αριθμητική είσοδος διαβάζεται σαν κείμενο	Δεν έγινε η κατάλληλη μετατροπή πριν από τον υπολογισμό.
Λίστα δίνει λάθος στοιχείο	Λάθος δείκτης ή ξεχάστηκε ότι η πρώτη θέση είναι 1.
Παλιό αποτέλεσμα εμφανίζεται	Δεν μηδενίστηκε/κρύφτηκε βοηθητική μεταβλητή.
Το έργο επιβραδύνεται	Πάρα πολλοί κλώνοι, βαριοί βρόχοι ή υπερβολική σχεδίαση.

ΚΕΦΑΛΑΙΟ 13 / ΑΠΟΔΟΣΗ

Όταν το έργο γίνεται βαρύ

Το Scratch είναι σχεδιασμένο για διαδραστικά εκπαιδευτικά έργα, αλλά ένα κακοσχεδιασμένο πρόγραμμα μπορεί να εκτελεί πολύ περισσότερη εργασία από όση χρειάζεται. Η βελτιστοποίηση ξεκινά αφού πρώτα έχουμε σωστή λειτουργία και μετρήσιμο πρόβλημα.

1. Συνηθισμένες αιτίες

- Πολλά σενάρια «για πάντα» που κάνουν βαριές πράξεις σε κάθε κύκλο.
- Κλώνοι που δημιουργούνται γρηγορότερα από όσο διαγράφονται.
- Επαναλαμβανόμενη αναζήτηση ολόκληρης λίστας ενώ το αποτέλεσμα θα μπορούσε να αποθηκευτεί.
- Σχεδίαση με υπερβολικά πολλά σημεία χωρίς ανάγκη.
- Πολλές ταυτόχρονες αλλαγές γραφικών, εφέ ή ήχων.

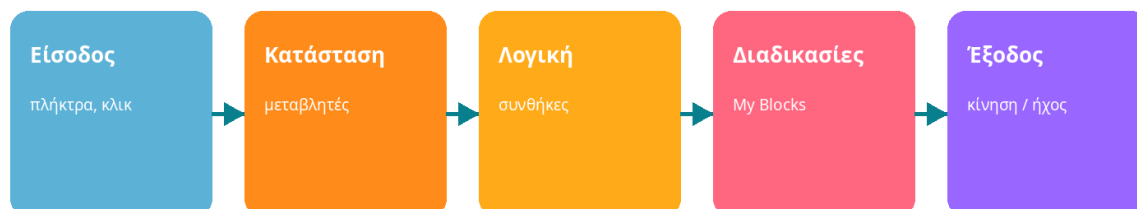
2. Βελτίωση χωρίς απώλεια σαφήνειας

Αφαιρέστε περιττές επαναλήψεις, κρατήστε ενδιάμεσα αποτελέσματα όταν είναι ασφαλές, διαγράψτε κλώνους εγκαίρως και εκτελείτε ακριβούς υπολογισμούς μόνο όταν άλλαξε η σχετική είσοδος.

Προτεραιότητα: Σωστό και κατανοητό πρόγραμμα πρώτα· βελτιστοποίηση μόνο όπου υπάρχει πραγματική ανάγκη.

Από την εκφώνηση στη δομή του έργου

Μια καθαρή αρχιτεκτονική έργου



Μια καθαρή ροή διαχωρίζει είσοδο, κατάσταση, λογική, διαδικασίες και παρουσίαση.

Πριν μεταφέρουμε μια εκφώνηση σε πλακίδια, είναι χρήσιμο να τη μετατρέψουμε σε μικρή προδιαγραφή: ποια δεδομένα δίνονται, ποιο αποτέλεσμα ζητείται, ποιες περιπτώσεις υπάρχουν και ποια στοιχεία πρέπει να θυμάται το πρόγραμμα.

1. Τέσσερις βασικές ερωτήσεις

1. Ποιες είναι οι είσοδοι;
2. Ποιες μεταβλητές περιγράφουν την κατάσταση;
3. Ποιοι κανόνες/υπολογισμοί μετατρέπουν την είσοδο σε αποτέλεσμα;
4. Πώς και πότε παρουσιάζεται το αποτέλεσμα στον χρήστη;

2. Διαχωρισμός ρόλων

Ακόμη και σε μικρό έργο είναι χρήσιμο να ξεχωρίζουμε τον κώδικα εισόδου από τον υπολογισμό και την παρουσίαση. Έτσι αλλάζουμε, για παράδειγμα, τον τρόπο εμφάνισης χωρίς να πειράζουμε τον αλγόριθμο.

Οργάνωση με καταστάσεις εφαρμογής

Κατάσταση εφαρμογής: ένα απλό μοντέλο



Παράδειγμα μετάβασης ανάμεσα σε Μενού, Παιχνίδι, Παύση και Τέλος.

Σε έργα με περισσότερες «οθόνες» είναι χρήσιμη μια μεταβλητή όπως «κατάσταση». Η τιμή της δηλώνει σε ποια φάση βρίσκεται η εφαρμογή και αποτρέπει ενέργειες που δεν έχουν νόημα στην τρέχουσα φάση.

1. Παράδειγμα

Κατάσταση	Επιτρεπτή συμπεριφορά
ΜΕΝΟΥ	Εμφάνιση κουμπιών, επιλογή έναρξης.
ΠΑΙΧΝΙΔΙ	Κίνηση, συγκρούσεις, σκορ.
ΠΑΥΣΗ	Πάγωμα βασικής δράσης, επιλογή συνέχισης.
ΤΕΛΟΣ	Εμφάνιση αποτελέσματος και επιλογή επανεκκίνησης.

2. Μεταβάσεις

Η μετάβαση πρέπει να γίνεται σε σαφές σημείο: ένα κουμπί, ένα μήνυμα ή μια συνθήκη. Όταν αλλάζει η κατάσταση, τα αντικείμενα μπορούν να προσαρμόζουν εμφάνιση και συμπεριφορά μέσω μηνύματος ή ελέγχου της κοινής μεταβλητής.

Ονόματα, σχόλια και διάταξη κώδικα

1. Ονόματα που εξηγούν τον ρόλο

Το όνομα «προσπάθειες» μεταφέρει περισσότερη πληροφορία από το «x». Το «Υπολόγισε ρίζες» ή «Σχεδίασε κύκλο» είναι σαφέστερο από το «block3». Η καλή ονοματοδοσία μειώνει την ανάγκη για σχόλια.

2. Σχόλια για το «γιατί»

Ένα χρήσιμο σχόλιο εξηγεί πρόθεση ή μη προφανή περιορισμό, π.χ. «Αποκλείουμε την προηγούμενη πρόταση ώστε το διάστημα να μικραίνει αυστηρά». Σχόλιο «αυξάνει το σκορ κατά 1» δίπλα σε ακριβώς αυτή την εντολή δεν προσθέτει πληροφορία.

3. Χωρική οργάνωση

- Κρατήστε σχετικά σενάρια κοντά μεταξύ τους.

- Αποφύγετε τεράστιες «στήλες» όταν μπορούν να γίνουν διαδικασίες.
- Αφαιρέστε αποσυνδεδεμένα πειραματικά σενάρια πριν από την υποβολή ή κρατήστε τα σαφώς σχολιασμένα.
- Μην αφήνετε ενεργά δύο διαφορετικά κύρια σενάρια που ανταγωνίζονται για την ίδια αρχικοποίηση.

ΚΕΦΑΛΑΙΟ 14 / ΑΛΓΟΡΙΘΜΟΣ

Ψευδοκώδικας πριν από τα πλακίδια

Ο ψευδοκώδικας είναι σύντομη περιγραφή της λογικής σε φυσική γλώσσα με δομή αλγορίθμου. Δεν χρειάζεται να μοιάζει με συγκεκριμένη γλώσσα προγραμματισμού. Βοηθά να λύσουμε πρώτα το πρόβλημα και μετά να επιλέξουμε τα κατάλληλα πλακίδια.

1. Παράδειγμα: μέγιστο τριών αριθμών

Διάβασε α, β, γ. Θέσε μέγιστο = α. Αν $\beta > \text{μέγιστο}$, θέσε μέγιστο = β. Αν $\gamma > \text{μέγιστο}$, θέσε μέγιστο = γ. Εμφάνισε μέγιστο.

2. Μεταφορά στο Scratch

Κάθε βήμα αντιστοιχεί σε γνωστή κατηγορία: ερώτηση/απάντηση στους Αισθητήρες, ανάθεση στις Μεταβλητές, σύγκριση στους Τελεστές, επιλογή στον Έλεγχο και παρουσίαση στις Όψεις.

Στρατηγική: Αν δεν μπορείτε να περιγράψετε τη λύση με σαφή βήματα χωρίς Scratch, το πρόβλημα συνήθως δεν είναι η εύρεση του σωστού πλακιδίου αλλά η αλγοριθμική σχεδίαση.

ΚΕΦΑΛΑΙΟ 14 / ΑΝΑΠΤΥΞΗ

Μικρές εκδόσεις αντί για μία μεγάλη υλοποίηση

Η σταδιακή ανάπτυξη μειώνει ταυτόχρονα τα πιθανά αίτια ενός σφάλματος. Κάθε μικρή έκδοση πρέπει να λειτουργεί πριν προσθέσουμε το επόμενο χαρακτηριστικό.

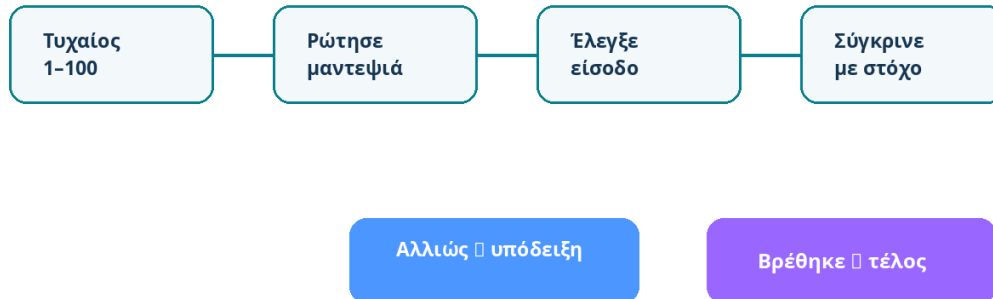
1. Δημιούργησε τη βασική σκηνή και αρχικοποίηση.
2. Υλοποίησε μία μόνο βασική διαδρομή με σταθερές δοκιμαστικές τιμές.
3. Πρόσθεσε είσοδο χρήστη και έλεγχο εγκυρότητας.
4. Πρόσθεσε τις εναλλακτικές περιπτώσεις.
5. Πρόσθεσε γραφικά, ήχο και βελτιώσεις διεπαφής.
6. Κάνε πλήρη πίνακα δοκιμών και καθάρισε τον κώδικα.

1. Γιατί όχι όλα μαζί;

Αν γραφικά, είσοδος, αλγόριθμος και μηνύματα προστεθούν ταυτόχρονα, ένα λάθος μπορεί να προέρχεται από πολλά σημεία. Με μικρά βήματα γνωρίζουμε ποια αλλαγή εισήγαγε τη νέα συμπεριφορά.

Παιχνίδι «Μάντεψε τον αριθμό»

Παράδειγμα: «Μάντεψε τον αριθμό»



Ροή ενός παιχνιδιού μαντεψιάς με έγκυρη είσοδο, υποδείξεις και μετρητή προσπαθειών.

1. Προδιαγραφή

- Ο υπολογιστής επιλέγει μία φορά μυστικό ακέραιο 1-100.
- Κάθε έγκυρη μαντεψιά μετρά ως προσπάθεια.
- Άκυρη είσοδος ζητείται ξανά χωρίς αύξηση μετρητή.
- Δίνεται υπόδειξη μεγαλύτερος/μικρότερος μέχρι την επιτυχία.

2. Δομή λύσης

1. Αρχικοποίησε μυστικό, προσπάθειες και μαντεψιά.
2. Επανάλαβε ώσπου μαντεψιά = μυστικός.
3. Διάβασε έγκυρο ακέραιο 1-100.
4. Αύξησε προσπάθειες.
5. Αν η μαντεψιά είναι μικρότερη/μεγαλύτερη, δώσε την κατάλληλη υπόδειξη.
6. Μετά τον βρόχο, εμφάνισε επιτυχία και προσπάθειες.

Κρίσιμο σημείο: Ο μυστικός αριθμός επιλέγεται έξω από τον βρόχο. Αν επιλέγεται ξανά μετά από κάθε μαντεψιά, ο στόχος αλλάζει.

Αριθμομηχανή με κουμπιά και μηνύματα

Η αριθμομηχανή είναι καλό παράδειγμα διαχωρισμού διεπαφής από υπολογισμό. Τέσσερα κουμπιά επιλέγουν πράξη και ένα αντικείμενο «Υπολογιστής» αναλαμβάνει τις ερωτήσεις και τον υπολογισμό.

1. Ρόλοι

Στοιχείο	Ευθύνη
Κουμπί +	Αν η εφαρμογή είναι ελεύθερη, δεσμεύει και μεταδίδει «πρόσθεση».
Κουμπί - / × / ÷	Ίδια λογική με διαφορετικό μήνυμα.

Στοιχείο	Ευθύνη
Υπολογιστής	Ζητά α και β, ελέγχει ειδικές περιπτώσεις, εκτελεί πράξη, εμφανίζει αποτέλεσμα, αποδεσμεύει.

2. Ασφαλής διαίρεση

Ο έλεγχος $\beta = 0$ γίνεται πριν από τη διαίρεση. Σε σφάλμα εμφανίζεται μήνυμα και η εφαρμογή επιστρέφει σε «απασχολημένος = 0» ώστε να μπορεί να γίνει νέα πράξη.

3. Γιατί τα κουμπιά δεν ρωτούν αριθμούς;

Αν κάθε κουμπί υλοποιούσε δικές του ερωτήσεις, θα είχαμε τέσσερα σχεδόν ίδια σενάρια. Η κεντρική διαδικασία μειώνει την επανάληψη και τον κίνδυνο ασυνέπειας.

ΚΕΦΑΛΑΙΟ 15 / ΠΑΡΑΔΕΙΓΜΑ

Επίλυση εξίσωσης με διακλαδώσεις

Η επίλυση εξίσωσης δείχνει πώς συνδυάζονται είσοδος, μεταβλητές, μαθηματικοί τελεστές και διακλαδώσεις. Πριν από κάθε τύπο ελέγχουμε αν οι προϋποθέσεις του επιτρέπουν τον υπολογισμό.

1. Δεδομένα

Δημιουργήστε λίστα «ερωτήσεις» και λίστα «σωστές απαντήσεις» ίδιου μήκους. Η ερώτηση στη θέση i αντιστοιχεί στην απάντηση στη θέση i .

2. Αλγόριθμος

1. Θέσε σκορ = 0 και θέση = 1.
2. Όσο θέση $\leq$ μήκος ερωτήσεων, εμφάνισε/ρώτησε το στοιχείο «θέση» της λίστας ερωτήσεων.
3. Σύγκρινε «απάντηση» με το στοιχείο της ίδιας θέσης στη λίστα σωστών απαντήσεων.
4. Αν είναι σωστή, αύξησε σκορ.
5. Αύξησε θέση και συνέχισε.
6. Στο τέλος εμφάνισε σκορ και πλήθος ερωτήσεων.

3. Επέκταση

Μπορείτε να προσθέσετε τρίτη λίστα με ανατροφοδότηση ή κατηγορία δυσκολίας. Αν θέλετε τυχαία σειρά χωρίς επανάληψη, χρειάζεται επιπλέον δομή ώστε μια ερώτηση που χρησιμοποιήθηκε να μη ξαναεπιλεγεί.

ΚΕΦΑΛΑΙΟ 15 / ΠΑΡΑΔΕΙΓΜΑ

Συστάδες κύκλων με παραμετρική διαδικασία

Στη σχεδίαση συστάδων κύκλων, η ίδια διαδικασία «κύκλος (ακτίνα)» καλείται από διαφορετικά κέντρα. Η θέση εκκίνησης και η παράμετρος αλλάζουν, ενώ ο ορισμός της διαδικασίας παραμένει ίδιος.

1. Έναρξη

Στην πράσινη σημαία: κατάσταση = ΜΕΝΟΥ, αλλάζει το υπόβαθρο, εμφανίζονται τα κουμπιά μενού και κρύβονται τα στοιχεία της δραστηριότητας.

2. Μετάβαση

Το κουμπί «Έναρξη» μεταδίδει «έναρξη δραστηριότητας». Ο ελεγκτής θέτει κατάσταση = ΠΑΙΧΝΙΔΙ, αρχικοποιεί δεδομένα και μεταδίδει μήνυμα ώστε τα σχετικά sprites να εμφανιστούν.

3. Τέλος

Όταν ολοκληρωθεί η δραστηριότητα, μεταδίδεται «τέλος», η κατάσταση γίνεται ΤΕΛΟΣ και εμφανίζεται η οθόνη αποτελέσματος. Έτσι οι κανόνες μετάβασης είναι συγκεντρωμένοι και ελέγξιμοι.

ΚΕΦΑΛΑΙΟ 15 / ΠΑΡΑΔΕΙΓΜΑ

Παραμετρική σχεδίαση με Πένα

Συνδυάστε τις έννοιες διαδικασίας, παραμέτρου, επανάληψης και γεωμετρίας σε ένα μικρό «εργαστήριο σχημάτων».

1. Διαδικασίες

Διαδικασία	Παράμετροι	Αποτέλεσμα
Πολύγωνο	πλευρές, μήκος	Σχεδιάζει κανονικό πολύγωνο.
Κύκλος	ακτίνα	Σχεδιάζει περιφέρεια με πραγματική ακτίνα.
Μετακινήσου	νέο x, νέο y	Με σηκωμένη πένα αλλάζει θέση χωρίς γραμμή.

2. Σύνθεση

Το κύριο σενάριο καθαρίζει τη σκηνή, καλεί τις διαδικασίες με διαφορετικές παραμέτρους και μετακινεί την αφετηρία ανάμεσα στα σχήματα. Οι διαδικασίες δεν χρειάζεται να γνωρίζουν τη συνολική σύνθεση.

Μάθημα: Η παραμετρική διαδικασία δεν είναι απλώς «λιγότερα πλακίδια». Είναι τρόπος να περιγράψουμε γενική λύση που λειτουργεί για πολλές τιμές.

ΚΕΦΑΛΑΙΟ 15 / ΥΠΟΒΟΛΗ

Τελικός τεχνικός έλεγχος ενός έργου

Πριν παραδοθεί ένα .sb3, ελέγξτε τόσο τη λειτουργία όσο και την καθαρότητα του έργου. Ένα πρόγραμμα που δουλεύει μόνο επειδή ο δημιουργός γνωρίζει τη σωστή σειρά κλικ δεν είναι ολοκληρωμένο παραδοτέο.

Checklist

- Το έργο ξεκινά σωστά από την πράσινη σημαία.
- Όλες οι μεταβλητές και λίστες αρχικοποιούνται όπως απαιτείται.
- Οι μη αποδεκτές είσοδοι δεν προχωρούν τη λογική.
- Οι οριακές περιπτώσεις έχουν δοκιμαστεί με γνωστό αναμενόμενο αποτέλεσμα.
- Τα μηνύματα προς τον χρήστη είναι σαφή και στα ελληνικά όπου απαιτείται.
- Τα sprites και οι μεταβλητές έχουν περιγραφικά ονόματα.
- Οι βοηθητικές μεταβλητές που δεν χρειάζονται είναι κρυμμένες.
- Δεν υπάρχουν ανεπιθύμητα ενεργά πειραματικά σενάρια.
- Οι κλώνοι διαγράφονται όταν τελειώνει ο ρόλος τους.
- Το αρχείο αποθηκεύεται, κλείνει, ανοίγει ξανά και εκτελείται χωρίς πρόβλημα.

Τελική δοκιμή: Εκτελέστε το έργο σαν να το ανοίγει κάποιος που δεν γνωρίζει πώς το κατασκευάσατε. Οι απαραίτητες ενέργειες πρέπει να είναι αυτονόητες ή να εξηγούνται στην οθόνη.

Οι βασικές κατηγορίες πλακιδίων

Οι βασικές κατηγορίες εντολών

Κίνηση

θέση, x/y, κατεύθυνση

Όψεις

ενδυμασίες, μέγεθος, εμφάνιση

Ήχος

ήχοι, ένταση, εφέ

Συμβάντα

εκκίνηση και μηνύματα

Έλεγχος

συνθήκες, βρόχοι, κλώνοι

Αισθητήρες

πληκτρολόγιο, ποντίκι, επαφή

Τελεστές

μαθηματικά, λογική, κείμενο

Μεταβλητές

δεδομένα που αλλάζουν

Λίστες

πολλές τιμές σε σειρά

Οι Εντολές μου

διαδικασίες και παράμετροι

Οι κατηγορίες χρώματος βοηθούν να εντοπίζουμε γρήγορα τον ρόλο κάθε πλακιδίου.

Κατηγορία	Ερώτηση που απαντά
Κίνηση	Πού βρίσκεται και προς τα πού κινείται το sprite;
Όψεις	Πώς φαίνεται και τι εμφανίζει;
Ήχος	Τι ακούγεται και πότε;
Συμβάντα	Τι ξεκινά τη συμπεριφορά;
Έλεγχος	Ποια διαδρομή/επανάληψη εκτελείται;
Αισθητήρες	Τι πληροφορία λαμβάνουμε από χρήστη/περιβάλλον;
Τελεστές	Πώς υπολογίζουμε, συγκρίνουμε ή επεξεργαζόμαστε κείμενο;
Μεταβλητές / Λίστες	Τι πρέπει να θυμάται το πρόγραμμα;
Οι Εντολές μου	Ποια επαναχρησιμοποιήσιμη λειτουργία θέλουμε να ονομάσουμε;

ΠΑΡΑΡΤΗΜΑ Α / ΣΧΗΜΑΤΑ ΠΛΑΚΙΔΙΩΝ

Πώς το σχήμα δείχνει τον ρόλο

Το σχήμα ενός πλακιδίου αποτελεί μέρος της «σύνταξης» του Scratch. Μας δείχνει πού μπορεί να τοποθετηθεί και τι είδους αποτέλεσμα παράγει.

Τύπος	Χαρακτηριστικό	Παράδειγμα ρόλου
Πλακίδιο καπέλο	Στρογγυλεμένη κορυφή· αρχίζει σενάριο.	όταν γίνει κλικ στην πράσινη σημαία
Πλακίδιο στοίβας	Συνδέεται πάνω/κάτω με άλλες εντολές.	κινήσου, όρισε, πες
Πλακίδιο C	Περιβάλλει άλλα πλακίδια.	επανάλαβε, εάν, για πάντα
Πλακίδιο τιμής	Στρογγυλό/οβάλ· μπαίνει σε πεδίο τιμής.	θέση x, σκορ, μήκος λίστας
Λογικό πλακίδιο	Εξαγωνικό· παράγει αληθές/ψευδές.	$x > 10$, αγγίζει;
Τερματικό πλακίδιο	Δεν δέχεται συνέχεια κάτω του.	σταμάτησε ..., διάγραψε κλώνο

Χρήσιμο: Αν ένα πλακίδιο «δεν κουμπώνει», ελέγξτε αν προσπαθείτε να βάλετε εντολή εκεί όπου απαιτείται τιμή ή λογική συνθήκη, ή το αντίστροφο.

ΠΑΡΑΡΤΗΜΑ Β / ΓΛΩΣΣΑΡΙ

Όροι Α–Κ

Όρος	Σύντομη εξήγηση
Αισθητήρας	Πλακίδιο που παρέχει πληροφορία από χρήστη, σκηνή ή περιβάλλον.
Αλγόριθμος	Πεπερασμένη και σαφής ακολουθία βημάτων για την επίλυση προβλήματος.
Αναδρομή	Διαδικασία που καλεί τον εαυτό της. Χρησιμοποιείται με προσοχή και απαιτεί συνθήκη τερματισμού.
Αρχικοποίηση	Ανάθεση γνωστών αρχικών τιμών πριν από την κύρια εκτέλεση.
Αντικείμενο / sprite	Προγραμματιζόμενη οντότητα της σκηνής.
Βρόχος	Δομή που επαναλαμβάνει εντολές.
Διαδικασία	Δική μας εντολή που συγκεντρώνει επαναχρησιμοποιήσιμη λειτουργία.
Ενδυμασία	Μία γραφική εμφάνιση ενός sprite.
Κατάσταση	Το σύνολο τιμών που περιγράφουν τι συμβαίνει τώρα στο έργο.
Κλώνος	Προσωρινό αντίγραφο sprite που δημιουργείται κατά την εκτέλεση.
Κριτήριο τερματισμού	Συνθήκη που επιτρέπει σε βρόχο/διαδικασία να ολοκληρωθεί.

ΠΑΡΑΡΤΗΜΑ Β / ΓΛΩΣΣΑΡΙ

Όροι Λ–Ω

Όρος	Σύντομη εξήγηση
Λίστα	Διατεταγμένη συλλογή πολλών τιμών.
Λογική τιμή	Αληθές ή ψευδές αποτέλεσμα συνθήκης.
Μεταβλητή	Ονομασμένη θέση που αποθηκεύει τιμή η οποία μπορεί να αλλάξει.
Μήνυμα	Ονομασμένο συμβάν που μεταδίδεται για συντονισμό σεναρίων.
Παράμετρος	Τιμή που δίνεται σε διαδικασία κατά την κλήση.
Σενάριο	Συνδεδεμένη ακολουθία πλακιδίων που εκτελείται ως μονάδα.
Σκηνή	Ο χώρος παρουσίασης του έργου και το αντικείμενο που διαχειρίζεται υπόβαθρο.
Συμβάν	Γεγονός που ξεκινά σενάριο.
Συνθήκη	Λογική έκφραση που καθορίζει απόφαση ή τερματισμό.
Τελεστής	Πλακίδιο που υπολογίζει αριθμό, κείμενο ή λογική τιμή.
Υπόβαθρο	Η γραφική εμφάνιση της Σκηνής.

ΠΑΡΑΡΤΗΜΑ Γ / ΣΥΝΤΟΜΟΣ ΟΔΗΓΟΣ

Δέκα συνήθειες που βελτιώνουν κάθε έργο

1. Αρχικοποιήστε πάντα το έργο από την πράσινη σημαία.
2. Αποθηκεύετε αμέσως κάθε απάντηση χρήστη που θα χρειαστεί αργότερα.
3. Ελέγχετε τις εισόδους πριν αλλάξετε μετρητές ή πραγματικά δεδομένα.
4. Χρησιμοποιείτε περιγραφικά ονόματα σε μεταβλητές, μηνύματα, sprites και διαδικασίες.
5. Μετατρέπετε επαναλαμβανόμενα ή νοηματικά τμήματα κώδικα σε «Οι Εντολές μου».
6. Χρησιμοποιείτε παραμέτρους αντί για πολλές σχεδόν ίδιες διαδικασίες.
7. Διατηρείτε σαφή κριτήρια τερματισμού σε κάθε βρόχο.
8. Δοκιμάζετε κανονικές, οριακές και μη αποδεκτές τιμές.
9. Κάνετε μία αλλαγή τη φορά όταν αποσφαλματώνετε.
10. Πριν από την υποβολή ανοίγετε ξανά το .sb3 και το εκτελείτε από την αρχή σαν νέος χρήστης.

Στόχος του μαθήματος: Να μπορείτε να εξηγείτε όχι μόνο ποια πλακίδια χρησιμοποιήσατε, αλλά και γιατί η δομή του αλγορίθμου είναι σωστή.

ΠΑΡΑΡΤΗΜΑ Δ / ΠΗΓΕΣ

Επίσημη τεκμηρίωση και περαιτέρω μελέτη

Για αλλαγές στο περιβάλλον, νέες επεκτάσεις και πρόσθετο εκπαιδευτικό υλικό, προτιμάτε τις επίσημες πηγές του Scratch και της Scratch Foundation. Το παρόν εγχειρίδιο εστιάζει στις σταθερές έννοιες προγραμματισμού και στη χρήση τους μέσα στο Scratch 3.

Πηγή	Τι προσφέρει
Scratch : scratch.mit.edu	Περιβάλλον δημιουργίας, κοινότητα, παραδείγματα έργων και βασική βοήθεια.
Scratch Foundation : scratchfoundation.org	Εκπαιδευτικό υλικό, οδηγοί και δραστηριότητες.
Scratch Help / Tutorials	Σύντομες οδηγίες για βασικές δυνατότητες του περιβάλλοντος.
Scratch : My Blocks / Custom Blocks	Υλικό για διαδικασίες, παραμέτρους και δομημένη επαναχρησιμοποίηση.

Πώς να διαβάσετε ξένα έργα

Η μελέτη ενός υπάρχοντος έργου είναι χρήσιμη όταν προσπαθείτε να καταλάβετε τη δομή του: ποια είναι τα sprites, ποια συμβάντα ξεκινούν σενάρια, ποια δεδομένα είναι κοινά και πώς επικοινωνούν τα μέρη. Μην περιορίζετε στην αντιγραφή πλακιδίων· προσπαθήστε να εξηγήσετε τη λογική με δικά σας λόγια.

ΠΑΡΑΡΤΗΜΑ Ε / ΑΥΤΟΑΞΙΟΛΟΓΗΣΗ

Ερωτήσεις κατανόησης

Οι παρακάτω ερωτήσεις μπορούν να χρησιμοποιηθούν ως σύντομος έλεγχος πριν από εργαστηριακή εξέταση ή νέα εργασία.

1. Γιατί η πράσινη σημαία πρέπει να επαναφέρει γνωστή αρχική κατάσταση;
2. Ποια είναι η διαφορά ανάμεσα σε «μετάδωσε» και «μετάδωσε και περίμενε»;
3. Πότε επιλέγουμε «εάν» και πότε «εάν ... αλλιώς»;
4. Ποια μεταβολή πρέπει να συμβαίνει ώστε ένας βρόχος «επανάλαβε ώσπου» να τερματίζει;
5. Γιατί πρέπει να αντιγράφουμε αμέσως την «απάντηση» σε μεταβλητή;
6. Ποια είναι η διαφορά ανάμεσα σε μεταβλητή για όλα τα αντικείμενα και μόνο για ένα αντικείμενο;
7. Γιατί πρέπει να ξεχωρίζουμε ένα σύμβολο πράξης από τους αριθμούς που συμμετέχουν στον υπολογισμό;
8. Ποιο πλεονέκτημα δίνει μια διαδικασία με παράμετρο;
9. Γιατί σηκώνουμε την Πένα πριν από μετακίνηση τοποθέτησης;
10. Ποιες τρεις κατηγορίες δοκιμών πρέπει να περιλαμβάνει τουλάχιστον ένας τελικός έλεγχος;

Προτεινόμενη χρήση: Απαντήστε χωρίς να κοιτάξετε το εγχειρίδιο και μετά επιστρέψτε στα αντίστοιχα κεφάλαια μόνο για τις ερωτήσεις στις οποίες δεν μπορείτε να αιτιολογήσετε την απάντηση.

Τι πρέπει να μπορείτε να κάνετε μετά τη μελέτη

Μετά την ολοκλήρωση του εγχειριδίου, ο στόχος δεν είναι η απομνημόνευση όλων των πλακιδίων. Είναι η ικανότητα να μετατρέπετε μια προδιαγραφή σε αλγόριθμο, να οργανώνετε το έργο σε κατανοητά μέρη, να ελέγχετε τις εισόδους και τις οριακές περιπτώσεις και να τεκμηριώνετε γιατί η λύση λειτουργεί.

Βασικές δεξιότητες

- Χρήση σκηνής, sprites, ενδυμασιών, ήχων και συντεταγμένων.
- Σχεδίαση με συμβάντα, μηνύματα, συνθήκες και βρόχους.
- Αποθήκευση κατάστασης σε μεταβλητές και δομημένη χρήση λιστών.
- Έλεγχος εγκυρότητας δεδομένων και χειρισμός ειδικών περιπτώσεων.
- Δημιουργία διαδικασιών με παραμέτρους και επαναχρησιμοποίηση κώδικα.
- Ανάπτυξη γεωμετρικών σχεδίων με Πένα.
- Συστηματική αποσφαλμάτωση και δοκιμή με αναμενόμενα αποτελέσματα.
- Οργάνωση μεγαλύτερων έργων με σαφείς ρόλους και καταστάσεις.

Επίλογος: Ένα καλό έργο Scratch δεν χαρακτηρίζεται από τον αριθμό των πλακιδίων, αλλά από τη σαφήνεια της σκέψης, την ορθότητα της λογικής και την ευκολία με την οποία μπορεί να ελεγχθεί και να επεκταθεί.