

Υλοποίηση

Κάθε χημικό στοιχείο έχει ένα πεδίο candidate το οποίο αν είναι yes σημαίνει ότι μπορεί να είναι αυτό που ψάχνουμε. Στο πρώτο στάδιο ο σκοπός του προγράμματος είναι να βρει ποια από αυτά είναι no ώστε να τα αποκλείσει.

Στο δεύτερο στάδιο, οι αποθήκες και τα φρεάτια έχουν και αυτά ένα πεδίο candidate που αν είναι yes σημαίνει ότι υπάρχει μόλυνση. Σε αυτό το στάδιο αποκλείουμε όσα έχουν candidate no και σχηματίζουμε τα διάφορα μονοπάτια που δημιουργούνται από τις πιθανές μολυσμένες αποθήκες μέχρι το σταθμό ελέγχου. Τέλος ξεκινάμε από το σταθμό ελέγχου και το πρόγραμμα αν βρει διασταύρωση με παραπάνω από ένα φρεάτιο που είναι μολυσμένο, κάνει ερώτηση στο χρήστη και ανάλογα με την απάντηση συνεχίζει στο κατάλληλο μονοπάτι.

Δημιουργήθηκαν η παρακάτω defunction:

- ask-question : Χρησιμοποιήθηκε σε πολλές περιπτώσεις ερωτήσεων αν ο χρήστης δεν απαντήσει με μια από τις προβλεπόμενες επιλογές τότε του ξαναγίνεται η ερώτηση.



Ομάδα:

Ανδρέας Αγαπητός

AEM: 2530

Παντελής Κυριακίδης

AEM: 2551

Δημιουργήθηκαν οι εξής defrules:

- asking_for_metrics : Από εδώ ξεκινάει το πρόγραμμα δίνοντας στο χρήστη την δυνατότητα να επιλέξει σε ποιες μετρήσεις θα δώσει τιμές και στη συνέχεια για κάθε επιλογή του ζητάει να δώσει την τιμή.
- isStrongAcid : Στη περίπτωση που επιλέξει το PH τσεκάρει αν ανήκει το PH στα όρια της κλάσης strongAcid. Αν όχι τότε κάνει το candidate απ'όλα τα στοιχεία που είναι strong acid no.
- Παρόμοια λειτουργούν το isWeakAcid, isStrongBase, isWeakBase και isOil. Επίσης με παρόμοιο τρόπο αλλά όχι με την μέτρηση PH λειτουργούν τα checkSoluble, checkSmell, checkSpGravity, checkColour, checkRadio, checkSpectro.
- checkWarehouse : Αφού βρήκαμε τα πιθανά μολυσμένα στοιχεία τότε ελέγχουμε σε ποιες αποθήκες υπάρχουν και σε αυτές που έχουν πιθανά μολυσμένα στοιχεία τις κάνουμε υποψήφιες.
- checkManhole : Παρόμοια με το checkWarehouse με την διαφορά ότι βλέπουμε το φρεάτιο με τι συνδέεται από τα αριστερά (previous) και ανάλογα το θέτουμε πιθανό μολυσμένο ή όχι.
- StartStation : Ξεκινάει η διαδικασία ελέγχου από το σταθμό, αφού έχουν σχεδιαστεί τα πιθανά μονοπάτια. Με την βοήθεια του assert δηλώνουμε το φρεάτιο που βρισκόμαστε.
- FindPollution : Αφού έχουμε το μολυσμένο φρεάτιο που συνδέεται με το σταθμό ελέγχου συνεχίζουμε το μονοπάτι μέχρι να βρούμε διασταύρωση όπως αναφέραμε στην αρχή. Εκεί γίνεται ερώτηση και διαλέγουμε το επόμενο φρεάτιο ανάλογα την απάντηση. Πάλι με την βοήθεια του assert κρατάμε το φρεάτιο που βρισκόμαστε.
- FindTheElemenInWarehouse : Αφού καταλήξουμε σε φρεάτιο που το previous είναι αποθήκη τότε ελέγχουμε μέσα σε αυτήν τα στοιχεία που είναι candidate=yes και τα προβάλλουμε στο χρήστη ως πιθανά μαζί με τους κινδύνους που μπορεί να έχουν.

Σημειώση:

- Το `specific_gravity` είναι σημειωμένο στις τιμές 0.9 , 1.0 και 1.1 λόγω έλλειψης ακριβής πληροφορίας. Στη περίπτωση εισαγωγής νεών στοιχείων με τιμή πχ 0.95 θα το δεχτεί γιατί έχουμε εύρος από [0.9,1.1]