

ΣΧΕΔΙΑΣΗ & ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ (ΗΥ 420)
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ, ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2014-2015
ΔΙΔΑΣΚΩΝ: ΧΡΗΣΤΟΣ Δ. ΑΝΤΩΝΟΠΟΥΛΟΣ

ΕΝΔΕΙΚΤΙΚΕΣ ΑΣΚΗΣΕΙΣ^{1,2}

Μοντέλα Διεργασίας Ανάπτυξης Λογισμικού

Άσκηση 1

Εάν υποθέσουμε την ανάπτυξη ενός συστήματος όπου οι πελάτες δεν είναι σίγουροι για αυτό που θέλουν, οι απαιτήσεις είναι συχνά κακώς ορισμένες. Ποιο από τα παρακάτω θα ήταν ένα κατάλληλο μοντέλο ανάπτυξης για αυτή την περίπτωση;

- α. Μοντέλο Δημιουργίας Πρωτοτύπου
- β. Μοντέλο Καταρράκτη
- γ. Μοντέλο V
- δ. Σπειροειδές Μοντέλο

Άσκηση 2

Η ομάδα εργασίας ,που αναπτύσσει ένα νέο σύστημα, είναι εξειδικευμένη σε ένα τομέα. Αν και το αναπτυσσόμενο έργο είναι αρκετά μεγάλο, δεν αναμένεται να απέχει πολύ από εφαρμογές που έχουν υλοποιηθεί από την ίδια ομάδα στο παρελθόν. Ποιο από τα παρακάτω θα ήταν ένα κατάλληλο μοντέλο ανάπτυξης για αυτή την περίπτωση;

- α. Μοντέλο Δημιουργίας Πρωτοτύπου
- β. Μοντέλο Καταρράκτη
- γ. Μοντέλο V
- δ. Σπειροειδές Μοντέλο

Προβλήματα – Λάθη

Άσκηση 1

Υποθέστε ότι αναπτύσσεται ένα σύστημα βιβλιοθήκης. Το σύστημα αποτελείται από

-
- 1 Το φυλλάδιο περιλαμβάνει τις εκφωνήσεις των ασκήσεων που επιλύθηκαν στο μάθημα. Οι ασκήσεις έχουν προέλθει από το βασικό εγχειρίδιο του μαθήματος, το βιβλίο “Theory and Problems of Software Engineering”, υλικό από τη διδασκαλία του μαθήματος σε προηγούμενα έτη, υλικό από τα θέματα εξετάσεων προηγούμενων ετών, καθώς και από τη διδασκαλία αντίστοιχων μαθημάτων στην Ελλάδα και το εξωτερικό.
 - 2 Θερμές ευχαριστίες στον φοιτητή Μιχάλη Τριανταφυλλίδη, ο οποίος ανέλαβε εθελοντικά τη δακτυλογράφηση μεγάλου μέρους του φυλλαδίου.

τρία κύρια υποσυστήματα: ένα που διαχειρίζεται τις συναλλαγές εισόδου-εξόδου, ένα που διαχειρίζεται τους καταλόγους και ένα που διαχειρίζεται αναφορές. Κατά τη διάρκεια της ανάπτυξης του συστήματος αρκετά προβλήματα εμφανίζονται. Προσδιορίστε τα προβλήματα ως σφάλματα, ελαττώματα ή αστοχίες.

- Στο κώδικα που υπολογίζει το ποσό των καθυστερημένων επιστροφών, η μεταβλητή *fine_total* δεν είναι αρχικοποιημένη.
- Όταν ένας βιβλιοθηκάριος προσπαθεί να καταχωρήσει ένα νέο τίτλο βιβλίου στον κατάλογο, το σύστημα κλείνει.
- Ο συγγραφέας των απαιτήσεων δεν είναι ενήμερος ότι η κάρτα βιβλιοθήκης δεν είναι απαραίτητη για συναλλαγή εισόδου.
- Στο κείμενο προδιαγραφών, το πρόστιμο για τις καθυστερημένες επιστροφές έχει οριστεί ως 0.25\$ την ημέρα με μέγιστο ποσό τα 15\$. Ο κώδικας για την υπολογισμό του προστίμου για τις καθυστερημένες επιστροφές δεν κάνει έλεγχο για μέγιστο ποσό.
- Κάθε βράδυ στις 23:00, το σύστημα της βιβλιοθήκης υποτίθεται ότι πρέπει να κρατήσει ένα αντίγραφο των καθημερινών συναλλαγών. Το αντίγραφο για το βράδυ της Τρίτης δεν δημιουργείται.

Απαιτήσεις

Άσκηση 1

Τα βενζινάδικα αμερικανικού τύπου είναι υπό μία έννοια βενζινάδικα self-service. Ο οδηγός βάζει μόνος του βενζίνη και είτε πληρώνει με πιστωτική κάρτα απευθείας στην αντλία, είτε προπληρώνει σε ταμία ο οποίος και ενεργοποιεί την αντλία ώστε να αγοραστεί το συγκεκριμένο ποσό καυσίμων. Σε αυτή την περίπτωση, ο ταμίας μπορεί στο τέλος της συναλλαγής να δώσει ρέστα εφόσον τελικά δεν αγοράστηκε το ποσό καυσίμων που είχε προπληρωθεί. Σε ένα τέτοιο βενζινάδικο συνήθως υπάρχει και ένα μικρό κατάστημα με είδη αυτοκινήτου (στο οποίο βρίσκεται και ο ταμίας). Το σύστημα πρέπει να αλληλεπιδράσει με άλλα αυτόματα συστήματα. Για παράδειγμα, προκειμένου να είναι δυνατές οι πληρωμές με πιστωτική, το σύστημα πρέπει να συνομιλεί με το υπολογιστικό σύστημα της τράπεζας. Το τελευταίο ελέγχει ότι ο πελάτης δεν έχει εξαντλήσει το πιστωτικό του όριο, χρεώνει την κάρτα και πληρώνει τελικά το βενζινάδικο.

Χαρακτηρίστε τις επόμενες προτάσεις ως ΣΩΣΤΕΣ αν ικανοποιούν τις προδιαγραφές για το αμερικανικό σύστημα βενζινάδικου και ΛΑΘΟΣ αν δε τις ικανοποιούν.

- Πόση τεκμηρίωση (documentation) απαιτείται να παράγει η ομάδα ανάπτυξης.
- Το επίπεδο εξοικείωσης που θα είναι απαραίτητο για τους ταμίες ώστε να χρησιμοποιήσουν το σύστημα αποδοτικά.
- Τους περιορισμούς που οι νέοι πελάτες, που πληρώνουν για βενζίνη, πρέπει να γνωρίζουν, δηλαδή πως να χρησιμοποιήσουν το σύστημα με απλές οδηγίες αναρτημένες στις αντλίες καυσίμων.
- Το μέγιστο κόστος του συστήματος.
- Τον προσδιορισμό του υλικού (hardware) που είναι απαραίτητο προκειμένου να υλοποιηθεί η διεπαφή μεταξύ του ταμείου και των αντλιών βενζίνης.
- Τη διαμόρφωση των δεδομένων που ανταλλάσσονται μεταξύ του ταμείου και των αντλιών βενζίνης.

- Πόση και τι είδους συντήρηση θα πραγματοποιείται στο σύστημα.

Άσκηση 2

Ποια από τα παρακάτω αποσπάσματα θα μπορούσαν να θεωρηθούν σωστές προδιαγραφές;

- “Μόλις ολοκληρωθεί η διαδικασία πληρωμής, το σύστημα θα πρέπει να αποκριθεί με τον ακόλουθο τρόπο : Εάν ο χρήστης έχει πληρώσει στον ταμία απευθείας, ή πλήρωσε στην αντλία αλλά δεν επιθυμεί απόδειξη, τότε επιστρέφει στην αρχική κατάσταση.
- “Θα πρέπει να διατηρείται μία καταγραφή για κάθε ταμία. Κάθε καταγραφή θα πρέπει να αποθηκεύει το όνομα, το επίθετο, και το ID του εργαζομένου. Οι καταγραφές θα πρέπει να διατηρούνται σε μία συνδεδεμένη λίστα.”
- “Μετά την επιλογή του τρόπου πληρωμής από την πλευρά του χρήστη, το σύστημα θα πρέπει να ελέγξει εάν η είσοδος είναι έγκυρη (για παράδειγμα αν επιλογή είναι ένας αριθμός μεταξύ ένα και τρία) “.

Άσκηση 3

Ποια από τα ακόλουθα είναι παραδείγματα από έγκυρες μη λειτουργικές απαιτήσεις?

- “Η οθόνη θα πρέπει να ανανεώνεται σε λιγότερο από 3 δευτερόλεπτα αφότου ο χρήστης έχει επιλέξει τρόπο πληρωμής.”
- “Όταν αγοράζονται ανταλλακτικά αυτοκινήτου, ο κατάλογος αποθεμάτων θα πρέπει να ανανεώνεται. Ένα προειδοποιητικό μήνυμα θα πρέπει να προβάλλεται εάν το πλήθος των αποθεμάτων είναι κάτω από ένα προκαθορισμένο αριθμό.
- “Ο χρήστης πρέπει να τοποθετήσει στη θέση του το ακροφύσιο όταν τελειώσει την άντληση καυσίμων.

Άσκηση 4

Πραγματοποιείται συνάντηση για την εξαγωγή απαιτήσεων ώστε να σιγουρευτούμε εάν οι απαιτήσεις περιγράφουν επαρκώς το σύστημα που θα δημιουργηθεί. Για τις επόμενες ερωτήσεις εξετάστε λεπτομερώς το δεδομένο απόσπασμα από τις απαιτήσεις και αποφασίστε εάν η εκάστοτε απαίτηση είναι επαρκής ή όχι. Εάν θα έπρεπε να επαναδιατυπωθεί, επισημάνετε όλους τους λόγους για τους οποίους αυτό θα έπρεπε να συμβεί (π.χ. είναι ανακριβής, είναι διφορούμενη ή ασυνεπής, δεν είναι ρεαλιστική, δε μπορεί να πιστοποιηθεί κλπ).

- Αφού ολοκληρωθεί η διαδικασία της πληρωμής, οι σχετικές πληροφορίες θα πρέπει να επισυνάπτονται σε ένα αρχείο καταγραφών.
- Το σύστημα θα πρέπει να είναι έτσι κατασκευασμένο ώστε να μπορεί εύκολα να προστεθεί νέα λειτουργικότητα στο μέλλον.
- Το κόστος του καυσίμου υπολογίζεται ως η τιμή ανά γαλόνι για τον τύπο του καυσίμου που αγοράστηκε, πολλαπλασιαζόμενη με τον αριθμό των γαλονιών που αγοράστηκαν.
- Το σύστημα πρέπει να είναι εύκολο να χρησιμοποιηθεί από νέους πελάτες.
- Το σύστημα πρέπει να είναι διαθέσιμο 24 ώρες την ημέρα, 7 μέρες την εβδομάδα.

Διαγράμματα Ροής Δεδομένων

Άσκηση 1

Το παρακάτω κείμενο περιγράφει τη λειτουργία μιας μονάδας ελέγχου αίματος σε σταθμό αιμοδοσίας. Φτιάξτε, με βάση το παρακάτω κείμενο:

- Ένα διάγραμμα ροής δεδομένων επιπέδου 0 το οποίο περιγράφει το σύστημα.
- Ένα περισσότερο εκλεπτυσμένο διάγραμμα ροής δεδομένων το οποίο περιλαμβάνει όλες τις πληροφορίες σχετικά με τη ροή των δεδομένων στο σύστημα. Θεωρήστε ότι και οι μονάδες αίματος είναι ένα είδος «δεδομένων».

Φροντίστε τα διαγράμματα να περιλαμβάνουν όλη την απαιτούμενη προσδιοριστική πληροφορία (επεξηγηματικά ονόματα στις διεργασίες, εξωτερικές οντότητες, ροές κλπ), αρίθμηση κ.ο.κ.

Η τράπεζα αίματος πραγματοποιεί αιμοληψίες σε εθελοντές αιμοδότες. Την επόμενη της αιμοδοσίας η τράπεζα αίματος ελέγχει όλες τις μονάδες αίματος ως προς την ομάδα αίματος και τυχόν μολυσματικούς παράγοντες. Τα αποτελέσματα στέλνονται στο γραφείο διαχείρισης (μια άλλη μονάδα του κέντρου αιμοδοσίας). Για κάθε ελεγχθείσα μονάδα αίματος συμπληρώνεται φόρμα που περιλαμβάνει τον μοναδικό κωδικό αριθμό της μονάδας, την ομάδα αίματος, την ημερομηνία αιμοληψίας και τα αποτελέσματα του ελέγχου. Αν ο έλεγχος αποκαλύψει μολυσματικούς παράγοντες, η μονάδα αίματος καταστρέφεται και το γεγονός σημειώνεται στη φόρμα.

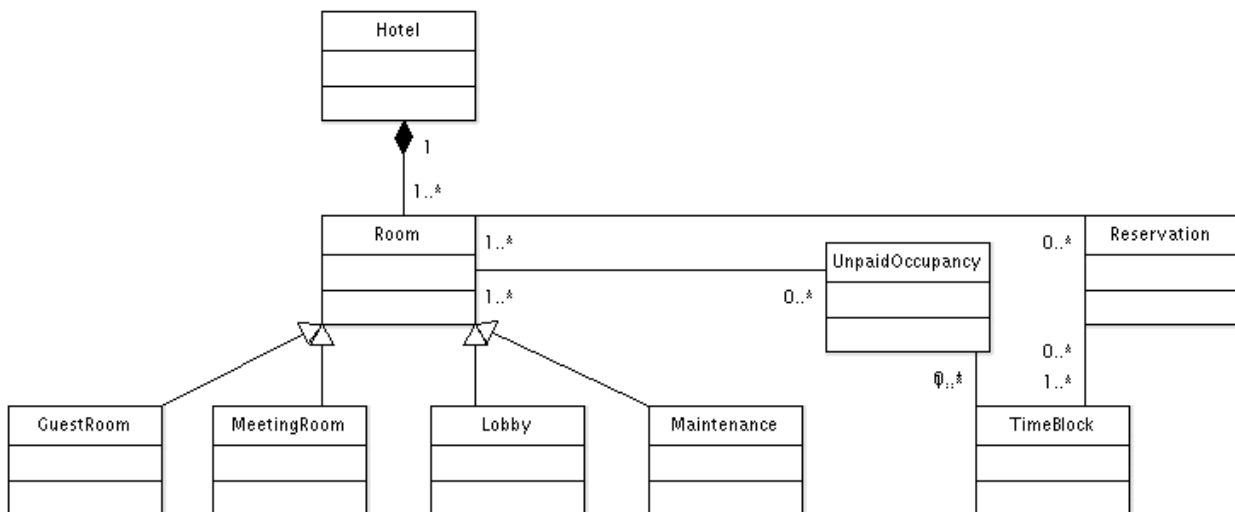
Οι μονάδες αίματος έχουν περιορισμένη ζωή. Κάθε μέρα η τράπεζα αίματος λαμβάνει από το γραφείο διαχείρισης μια λίστα κωδικών μονάδων αίματος που έχουν λήξει. Οι αντίστοιχες μονάδες καταστρέφονται και η λίστα επιστρέφεται στο γραφείο διαχείρισης με την επισήμανση ότι οι ληγμένες μονάδες αίματος έχουν καταστραφεί.

Η τράπεζα αίματος διαθέτει αίμα στα νοσοκομεία που το ζητούν. Τα αιτήματα των νοσοκομείων λαμβάνονται από το γραφείο διαχείρισης και αφορούν συνήθως συγκεκριμένη ομάδα αίματος. Η τράπεζα αίματος λαμβάνει από το γραφείο διαχείρισης λίστα με τις συγκεκριμένες ανάγκες κάθε νοσοκομείου. Η λίστα τυπώνεται σε τριπλότυπο. Όταν πραγματοποιηθεί η διάθεση αίματος η τράπεζα αίματος επιστρέφει ένα υπογεγραμμένο αντίγραφο στο γραφείο διαχείρισης. Ένα αντίγραφο συνοδεύει τις μονάδες αίματος στο νοσοκομείο. Το τρίτο αντίγραφο διατηρείται για ένα έτος στην τράπεζα αίματος.

UML: Διαγράμματα Κλάσεων

Άσκηση 1

Αναλύστε το ακόλουθο διάγραμμα κλάσεων, εξηγώντας σε βάθος αυτό που μοντελοποιεί. Υπάρχουν λάθη σε αυτή τη σχεδίαση; Αν ναι, διορθώστε τα



Άσκηση 2

Σχεδιάστε ένα διάγραμμα κλάσεων που αναπαριστά ένα βιβλίο, το οποίο ορίζεται από την ακόλουθη περιγραφή: “Ένα βιβλίο συντίθεται από έναν αριθμό μερών, που με τη σειρά τους αποτελούνται από έναν αριθμό κεφαλαίων. Τα κεφάλαια αποτελούνται από έναν αριθμό ενότητων.” Εστιάστε μόνο στις κλάσεις και στις συσχετίσεις. Επιπλέον στο παραπάνω διάγραμμα προσθέστε τις πληθικότητες.

- Επεκτείνετε το διάγραμμα κλάσεων της προηγούμενης άσκησης συμπεριλαμβάνοντας τις εξής ιδιότητες:
- Το βιβλίο περιλαμβάνει έναν εκδότη, μία ημερομηνία κυκλοφορίας και έναν αριθμό ISBN.
- Ένα μέρος περιλαμβάνει έναν τίτλο και έναν αριθμό.
- Ένα κεφάλαιο περιλαμβάνει έναν τίτλο, έναν αριθμό και μία περίληψη.
- Μία ενότητα περιλαμβάνει έναν τίτλο και έναν αριθμό.
- Θεωρείστε το διάγραμμα του προηγούμενου ερωτήματος. Παρατηρείστε ότι το Μέρος, το Κεφάλαιο και η Ενότητα, όλα συμπεριλαμβάνουν έναν τίτλο και έναν αριθμό ως γνωρίσματα. Προσθέστε μία κλάση και μία σχέση κληρονομικότητας ώστε να εξάγετε τα δύο γνωρίσματα σε μία νέα κλάση.

Άσκηση 3

Σχεδιάστε ένα διάγραμμα κλάσεων που αναπαριστά τη σχέση μεταξύ παιδιών και γονέων. Να λάβετε υπόψιν σας ότι ένα άτομο μπορεί να έχει και γονέα και παιδί. Προσδιορίστε στις σχέσεις τους ρόλους και τους λόγους πληθικότητας.

Άσκηση 4

Παραθέτουμε μέρος των απαιτήσεων εφαρμογής ενός e-bookshop:

- Ο πελάτης πρέπει να μπορεί να βλέπει τους τίτλους των βιβλίων και τα στοιχεία του κάθε βιβλίου, να παραγγέλνει βιβλία και να εγγράφεται σε ηλεκτρονική λίστα αλληλογραφίας για να ενημερώνεται για νέες παραλαβές.
- Όταν ένας πελάτης εγγράφεται στην ηλεκτρονική λίστα αλληλογραφίας θα του ζητείται να παρέχει τα στοιχεία του (όνομα, διεύθυνση, e-mail). Ένας αρμόδιος υπάλληλος, ο υπεύθυνος προώθησης, θα αποστέλλει e-mail με τις σχετικές πληροφορίες σε τακτά χρονικά διαστήματα.
- Ο πελάτης θα μπορεί να επιλέγει βιβλία για αγορά και τα στοιχεία τους θα τοποθετούνται σε ηλεκτρονικό καλάθι. Επιπλέον για κάθε νέα επιλογή βιβλίου θα εμφανίζεται η συνολική τιμή αγοράς των περιεχομένων του καλαθιού. Όταν ολοκληρώνεται η επιλογή βιβλίων ο πελάτης θα μπορεί να ορίζει μέθοδο αποστολής και να πληρώνει ηλεκτρονικά για την παραγγελία με χρήση πιστωτικής κάρτας.
- Θα γίνεται ηλεκτρονική διαχείριση αποθήκης. Η αποθήκη θα ενημερώνεται όταν νέα βιβλία παραλαμβάνονται από τους εκδοτικούς οίκους καθώς και όταν βιβλία αποστέλλονται σε πελάτες. Επιπλέον θα παραγγέλλονται νέα αντίτυπα από όσα βιβλία κοντεύουν να εξαντληθούν (ο αριθμός αντιτύπων βρίσκεται κάτω από ένα ορισμένο όριο). Η νέες παραγγελίες θα επικυρώνονται από τον αρμόδιο διαχειριστή αποθήκης που θα αλληλεπιδρά με το σύστημα και θα υπογράφει και τις σχετικές αναφορές κίνησης αποθήκης.
- Τέλος ο υπεύθυνος πωλήσεων θα προσδιορίζει τις τιμές των βιβλίων και θα καθορίζει την πολιτική των ειδικών προσφορών σε συχνούς πελάτες με στόχο την προώθηση των πωλήσεων.

Μετά από μια πρώτη αξιολόγηση των παραπάνω έχουν βρεθεί οι παρακάτω υποψήφιος κλάσεις:

- *Καρτέλα πελάτη*: Τα στοιχεία του πελάτη
- *Βιβλίο*: Τα στοιχεία του βιβλίου
- *Παραγγελία*: Αναφέρεται στο σύνολο των βιβλίων που παραγγέλνει κάποιος πελάτης. Μπορεί να περιλαμβάνει ένα ή περισσότερα αντίγραφα από το ίδιο ή διαφορετικά βιβλία. Περιέχει μια ή περισσότερες γραμμές παραγγελίας και πληροφορίες για τη διεύθυνση που θα αποσταλεί.
- *Γραμμή παραγγελίας*: Αναφέρεται σε μια γραμμή στην παραγγελία του πελάτη, για παράδειγμα στην παραγγελία ενός αριθμού αντιγράφων από ένα συγκεκριμένο βιβλίο. Μια παραγγελία περιέχει μία ή περισσότερες γραμμές παραγγελίας.
- *Παραγγελία σε αναμονή (backorder)*: Αυτό είναι το μέρος της παραγγελίας που δεν μπορεί να ικανοποιηθεί με τα υπάρχοντα αποθέματα βιβλίων στην αποθήκη. Το μέρος αυτό της παραγγελίας ικανοποιείται όταν παραληφθούν τα αντίστοιχα βιβλία στην αποθήκη.
- *Παραγγελία σε εκτέλεση*: Το μέρος της παραγγελίας που μπορεί να ικανοποιηθεί και αποστέλλεται άμεσα στον πελάτη.
- *Αποθήκη*: Περιγράφει τα βιβλία που είναι αποθηκευμένα στην αποθήκη. Περιέχει καρτέλες βιβλίου
- *Καρτέλα βιβλίου*: Περιγράφει τις πληροφορίες σχετικά με την αποθήκευση ενός βιβλίου, όπως αριθμός αντιτύπων που βρίσκονται στην αποθήκη και κριτήρια παραγγελίας επιπρόσθετων αντιτύπων.
- *Πιστωτική κάρτα*: Περιέχει πληροφορίες σχετικά με την πιστωτική κάρτα του

- πελάτη.
- α) Με βάση τις απαιτήσεις που περιγράφηκαν παραπάνω προσδιορίστε τους χειριστές (actors) του συστήματος.
- β) Με βάση τις κλάσεις του συστήματος που αναφέρθηκαν παραπάνω σχεδιάστε το διάγραμμα κλάσεων όπου θα απεικονίζονται η πληθικότητα και οι συσχετίσεις μεταξύ των κλάσεων.

Άσκηση 5

Σχεδιάστε το διάγραμμα κλάσεων (UML) για το πληροφοριακό σύστημα ενός ταξιδιωτικού γραφείου. Φροντίστε να περιλάβετε τις βασικές κλάσεις, συσχετισμούς, ρόλους και πολλαπλότητες. Οι απαιτήσεις περιγράφονται στις ακόλουθες γραμμές:

Το σύστημα χειρίζεται πληροφορίες πελατών, εκδρομών/δραστηριοτήτων, κρατήσεων και τυχόν παραπόνων. Οι πελάτες οργανώνουν μόνοι τους τη δομή των ταξιδιών τους. Κάθε πελάτης μπορεί να κάνει μια κράτηση ή να υποβάλλει παράπονο σχετικό με κάποια προηγούμενη κράτηση. Κάθε κράτηση αφορά έναν πελάτη και μια εκδρομή/δραστηριότητα. Επίσης αφορά συγκεκριμένη χρονική περίοδο. Πολλαπλές εκδρομές/δραστηριότητες του ίδιου πελάτη μπορούν να συνδυαστούν σε ένα ταξίδι. Οι εκδρομές/δραστηριότητες μπορούν να είναι δύο ειδών: πεζές ή να απαιτούν κάποιο μέσο μεταφοράς. Στη δεύτερη περίπτωση επισημαίνεται και το χρησιμοποιούμενο μέσο μεταφοράς, το οποίο μπορεί να είναι λεωφορείο, τρένο, πλοίο ή αεροπλάνο.

Άσκηση 6

Τα Χριστούγεννα είναι μόλις 3 μήνες μακριά και ο Αγ. Βασίλης προετοιμάζεται. Φτιάξτε ένα διάγραμμα κλάσεων για το πληροφοριακό σύστημα του Αγ. Βασίλη που περιγράφεται παρακάτω. Συμπεριλάβετε τις πολλαπλότητες και τους ρόλους στις σχέσεις μεταξύ κλάσεων.

Μια ομάδα ξωτικών κατασκευάζει παιχνίδια. Κάθε ξωτικό μπορεί να φτιάξει πολλαπλά παιχνίδια. Υπάρχουν τρία ήδη παιχνιδιών: τρενάκια, κούκλες και PS3. Κάθε παιδί μπορεί να ζητήσει έως 3 παιχνίδια. Όλα τα παιδιά δικαιούνται δώρο (ναι... ακόμα και τα κακά). Τα περισσότερα παίρνουν 1 παιχνίδι σαν δώρο, κάποια όμως μπορεί να πάρουν δύο.

Άσκηση 7

Η εταιρεία ανάπτυξης λογισμικού στην οποία ανήκετε ανέλαβε τη δημιουργία ενός συστήματος λογισμικού για νοσοκομείο. Το νοσοκομείο διαθέτει αρκετά εξειδικευμένα τμήματα: Καρδιολογικό, Γυναικολογικό, Ορθοπαιδικό, Παιδιατρικό, ΩΡΛ. Επίσης, το Τμήμα Επειγόντων Περιστατικών (ΤΕΠ) είναι ένα ακόμα Τμήμα. Ένας γιατρός σχετίζεται κάθε στιγμή με ένα Τμήμα, αλλά μπορεί ταυτόχρονα να είναι μέλος και του ΤΕΠ. Κάθε γιατρός προσφέρει υπηρεσίες στο ΤΕΠ συγκεκριμένη μέρα και ώρα κάθε εβδομάδα.

Στον χώρο υποδοχής λαμβάνονται τα στοιχεία του ασθενή και καταγράφεται. Εάν είναι η πρώτη επίσκεψη του ασθενή στο νοσοκομείο δημιουργείται και ένας μοναδικός

αριθμός μητρώου που τον χαρακτηρίζει. Επιπλέον, προεισπράττονται τυχόν νοσήλια. Όταν τα παραπάνω ρυθμιστούν, ανατίθεται ένας γιατρός ο οποίος θα εξετάσει τον ασθενή.

Ο γιατρός μπορεί να ζητήσει από τον ασθενή εξετάσεις. Ο ασθενής επισκέπτεται τα εργαστήρια του νοσοκομείου για να εκτελέσει τις εξετάσεις. Όταν οι εξετάσεις ολοκληρωθούν, οι σχετικές αναφορές δίνονται στον ασθενή, προκειμένου να τις παρουσιάσει στον γιατρό. Πληρωμές σχετικά με τις εξετάσεις επίσης γίνονται στον χώρο υποδοχής. Αφού μελετήσει τα αποτελέσματα, ο γιατρός συνταγογραφεί φαρμακευτική αγωγή, ή ζητά περαιτέρω εξετάσεις, ή αποφασίζει να εισάγει τον ασθενή σε κάποια από τις κλινικές του νοσοκομείου. Ο ασθενής εισάγεται στην κλινική ενός εξειδικευμένου τμήματος (αν διαθέτει) με την εντολή του γιατρού. Ο αριθμός των κρεβατιών στην κλινική είναι περιορισμένος. Αν δεν υπάρχει διαθέσιμο κρεβάτι, η εισαγωγή μπορεί να προγραμματιστεί για μεταγενέστερη χρονική στιγμή. Αν ένας ασθενής, μετά την εισαγωγή του, χρειαστεί επέμβαση, τότε προγραμματίζεται μέρα και ώρα από τον γιατρό που θα εκτελέσει την επέμβαση.

Μετά την ολοκλήρωση της θεραπείας, ο ασθενής παίρνει εξιτήριο μετά από εισήγηση του γιατρού και αφού εξοφλήσει το κόστος νοσηλείας στον χώρο υποδοχής. Με την εξόφληση, η υποδοχή εκδίδει το εξιτήριο του ασθενή.

α) Σχεδιάστε το διάγραμμα κλάσεων για το σύστημα λογισμικού που περιγράφηκε παραπάνω. Περιγράψτε όλες τις κλάσεις, τις σχέσεις μεταξύ τους και την πολλαπλότητα των σχέσεων. Δεν απαιτείται να περιγράψετε τις μεθόδους και τα χαρακτηριστικά (attributes) κάθε κλάσης.

β) Επικεντρωθείτε στη διαδικασία ανάθεσης μοναδικού αριθμού μητρώου σε κάθε ασθενή. Ποιο πρότυπο σχεδίασης λογισμικού θα εφαρμόζατε ώστε να εγγυηθείτε ότι κάθε ασθενής θα λαμβάνει μοναδικό αριθμό μητρώου; Δώστε τον ψευδοκώδικα (ή κανονικό κώδικα αν επιθυμείτε) της υλοποίησης της αντίστοιχης κλάσης / κλάσεων.

γ) Σχεδιάστε το διάγραμμα ακολουθίας για τη διαδικασία της εξέτασης ασθενή.

UML: Διαγράμματα Περιπτώσεων Χρήσης

Άσκηση 1

Actors ενός Α.Τ.Μ.

- Θεωρείστε ένα σύστημα Α.Τ.Μ. Προσδιορίστε τουλάχιστον τρεις actors που αλληλεπιδρούν με αυτό το σύστημα.
- Μπορεί να το σύστημα υπό κάποια θεώρηση να αναπαρασταθεί ως actor? Δικαιολογήστε την απάντησή σας.

Άσκηση 2

Αυτόματος πωλητής εισιτηρίων: Σχεδιάστε ένα διάγραμμα περιπτώσεων χρήστη για μια αυτόματη μηχανή πώλησης εισιτηρίων τρένου. Το σύστημα συμπεριλαμβάνει 2 actors : έναν ταξιδιώτη που αγοράζει διαφορετικούς τύπους εισιτηρίων, και κεντρικό υπολογιστή, που διατηρεί βάση δεδομένων για το κόστος των εισιτηρίων. Οι

περιπτώσεις χρήσης πρέπει να περιλαμβάνουν οπωσδήποτε τις ακόλουθες: BuyOneWayTicket, BuyWeeklyCard, BuyMonthlyCard, UpdateTariff. Επίσης να συμπεριλάβετε τις ακόλουθες περιπτώσεις εξαιρέσεων: Time-Out (π.χ. ο ταξιδιώτης έκανε πολλή ώρα να εισάγει το σωστό ποσό πληρωμής), TransactionAborted (π.χ. ο ταξιδιώτης ζήτησε ακύρωση της συναλλαγής πριν αυτή ολοκληρωθεί) , DistributorOutOfChange και DistributorOutOfPaper.

Άσκηση 3

Φτιάξτε ένα διάγραμμα περιπτώσεων χρήσης το οποίο περιγράφει το ακόλουθο σενάριο:

Οι εγγραφές σε ένα πανεπιστήμιο διαχειρίζονται από τον προϊστάμενο της Γραμματείας. Ένας φοιτητής κάνει αίτηση εγγραφής στο πανεπιστήμιο. Στα πλαίσια της εγγραφής του επιλέγει και τα μαθήματα που θα παρακολουθήσει. Στην ειδική περίπτωση που ένας φοιτητής είναι αλλοδαπός, θα πρέπει, στα πλαίσια της διαδικασίας εγγραφής, να πραγματοποιηθεί επιπλέον έλεγχος εγκυρότητας των στοιχείων του. Από τη στιγμή που ένας φοιτητής εγγραφεί, έχει τη δυνατότητα να εγγράψει και άλλα μέλη της οικογένειάς του στο πανεπιστήμιο.

Άσκηση 4

Φτιάξτε ένα διάγραμμα περιπτώσεων χρήσης για το βενζινάδικο αμερικανικού τύπου που περιγράφηκε νωρίτερα.

UML: Διαγράμματα Ακολουθίας

Άσκηση 1

Πυρκαγιά στην αποθήκη εμπορευμάτων. Σχεδιάστε ένα διάγραμμα ακολουθίας για το σενάριο WareHouseOnFire που περιγράφεται στον ακόλουθο πίνακα. Να συμπεριλάβετε τα αντικείμενα bob, alice, john, friend, και στιγμιότυπα άλλων κλάσεων που ενδεχομένως να χρειαστείτε. Απεικονίστε μόνο τα 5 πρώτα μηνύματα που στέλνονται.

Όνομα Σεναρίου	WareHouseOnFire
Στιγμιότυπα actors που συμμετέχουν	Bob , alice : Field Officer john : Dispatcher
Ακολουθία Γεγονότων	1. Ο Bob, οδηγώντας στο κεντρικό δρόμο μέσα στο περιπολικό, παρατηρεί ότι βγαίνει καπνός από μία αποθήκη. Η βοηθός του, η Alice, ενεργοποιεί τη λειτουργία “Report Emergency” από το laptop της, το “FRIEND” 2. Η Alice εισάγει τη διεύθυνση του κτιρίου, μία σύντομη περιγραφή της ακριβούς τοποθεσίας της φωτιάς (π.χ. βορειοδυτική γωνία κτηρίου) και ένα επίπεδο εκτιμώμενου

	<p>κινδύνου. Εκτός από πυροσβεστικές μονάδες, ζητά και κάποια ασθενοφόρα δεδομένου ότι η περιοχή έχει αρκετή κίνηση. Επιβεβαιώνει την εισαγωγή της και περιμένει για επιβεβαίωση.</p> <p>3. Ο John, ο συντονιστής (dispatcher), ειδοποιείται για το συμβάν από ένα ηχητικό σήμα που παράγεται από το σταθμό εργασίας του. Εξετάζει τις πληροφορίες που καταχώρησε η Alice και επιβεβαιώνει την αναφορά. Δεσμεύει μία πυροσβεστική μονάδα και 4. ασθενοφόρα και στέλνει τον εκτιμώμενο χρόνο άφιξης τους (ETA) στην Alice.</p> <p>5. Η Αλίκη λαμβάνει την επιβεβαίωση και τους χρόνους άφιξης των ασθενοφόρων.</p>
--	---

Άσκηση 2

Κάθε φορά που διεξάγεται ενδιάμεση πρόοδος σε κάποιο μάθημα γίνονται τα ακόλουθα: Ο καθηγητής αρχικά ενημερώνει τους φοιτητές για την ημερομηνία και την ύλη της προόδου. Κατόπιν ετοιμάζει τα θέματα (μαζί με ενδεικτικές λύσεις) και τα δίνει στη γραμματεία για αναπαραγωγή. Την προκαθορισμένη μέρα και ώρα τα θέματα δίνονται στους φοιτητές. Οι φοιτητές, αφού απαντήσουν, επιστρέφουν τα γραπτά στον διδάσκοντα. Αυτός με τη σειρά του τα δίνει στους βοηθούς του μαθήματος, μαζί με τις ενδεικτικές λύσεις. Οι βοηθοί διορθώνουν τα γραπτά και τα δίνουν πάλι στον διδάσκοντα. Ο τελευταίος, καταγράφει τους βαθμούς και επιστρέφει τα διορθωμένα γραπτά στους φοιτητές.

Φτιάξτε ένα διάγραμμα ακολουθίας το οποίο αναπαριστά την παραπάνω διαδικασία. Θα πρέπει στο διάγραμμα να είναι σαφή τα χρονικά διαστήματα συμμετοχής κάθε παίκτη στη διαδικασία, η λειτουργία που πραγματοποιείται σε κάθε αλληλεπίδραση και τυχόν ορίσματά της.

UML: Διαγράμματα καταστάσεων

Άσκηση 1

Σχεδιάστε διάγραμμα καταστάσεων που περιγράφει την αλληλεπίδραση μεταξύ ενός πελάτη, που προσπαθεί να αγοράσει ένα συγκεκριμένο CD μουσικής με μετρητά, και του υπαλλήλου σε ένα κατάστημα πωλήσεων CD. Προσπαθήστε να καλύψετε όλες τις πιθανότητες.

Άσκηση 2

Ένα απλό σύστημα διαχείρισης φούρνου μικροκυμάτων περιλαμβάνει τα παρακάτω πλήκτρα:

- Πλήκτρα έναρξης-ακύρωσης λειτουργίας (start-cancel)
- Πλήκτρα καθορισμού χρόνου λειτουργίας (timer και αριθμητικά πλήκτρα)

- Πλήκτρο καθορισμού ισχύος μικροκυμάτων (πλήρης ισχύς ή μισή ισχύς).
- Το άνοιγμα της πόρτας απενεργοποιεί τη συσκευή (δεν είναι δυνατόν να αρχίσει να λειτουργεί και αν βρίσκεται ήδη σε λειτουργία τότε η λειτουργία διακόπτεται). Αρχικά το σύστημα βρίσκεται σε κατάσταση αναμονής.
- Οι πιθανές εισόδους του χρήστη προς το σύστημα περιγράφονται στον παρακάτω πίνακα:

Είσοδος	Περιγραφή
Μισή ισχύς	Έχει τοποθετηθεί το πλήκτρο καθορισμού της ισχύος στο μισό
Πλήρης ισχύς	Έχει τοποθετηθεί το πλήκτρο καθορισμού της ισχύος στο 100%
Timer	Έχει πιεστεί το πλήκτρο timer
Αριθμός	Ο χρήστης έχει πιάσει ένα αριθμητικό πλήκτρο
Άνοιγμα πόρτας	Έχει ανοίξει η πόρτα του φούρνου μικροκυμάτων
Κλείσιμο πόρτας	Έχει κλείσει η πόρτα του φούρνου μικροκυμάτων
Start	Έχει πιεστεί το πλήκτρο έναρξης λειτουργίας
Cancel	Έχει πιεστεί το πλήκτρο ακύρωσης λειτουργίας
End	Έχει ολοκληρωθεί ο χρόνος λειτουργίας που ζητήθηκε

Για ευκολία υποθέτουμε ότι η γενική σειρά των ενεργειών για χρήση του φούρνου μικροκυμάτων είναι:

- Αρχικά η συσκευή βρίσκεται σε κατάσταση αναμονής.
- Επιλέγεται η ισχύς λειτουργίας. Αυτή μπορεί να είναι πλήρης ή μισή ισχύς.
- Επιλέγεται ο καθορισμός του χρόνου λειτουργίας με πίεση του πλήκτρου timer.
- Άνοιγμα της πόρτας για είσοδο του φαγητού και κλείσιμο πόρτας. Για όσο διάστημα η πόρτα είναι ανοιχτή η συσκευή είναι απενεργοποιημένη. Όταν κλείσει η πόρτα η συσκευή ενεργοποιείται και πάλι. Αν το φαγητό είναι ήδη μέσα στη συσκευή τότε απλώς κλείνει η πόρτα.
- Πίεση του πλήκτρου start και ο φούρνος λειτουργεί για τον καθορισμένο χρόνο εφόσον η πόρτα είναι κλειστή. Αν στο μεταξύ ανοίξει η πόρτα η συσκευή απενεργοποιείται και πρέπει να ξανακλείσει η πόρτα και να πιεστεί ξανά το πλήκτρο start για να ξαναρχίσει η συσκευή να λειτουργεί.
- Η συσκευή λειτουργεί για τον καθορισμένο χρόνο και κατόπιν επιστρέφει στην κατάσταση αναμονής. Αν όσο λειτουργεί πιεστεί το πλήκτρο cancel τότε η λειτουργία διακόπτεται και η συσκευή επιστρέφει σε κατάσταση αναμονής

Να προσδιοριστούν οι πιθανές καταστάσεις και να κατασκευαστεί το διάγραμμα μετάβασης για το παραπάνω σύστημα λαμβάνοντας υπ' όψιν ότι:

- α) η συσκευή μπορεί να ενεργοποιηθεί μόνο όταν βρίσκεται σε κατάσταση καθορισμού χρόνου ή σε κατάσταση απενεργοποίησης
- β) η συσκευή μπορεί να απενεργοποιηθεί μόνο όταν είναι σε κατάσταση καθορισμού χρόνου ή σε κατάσταση λειτουργίας

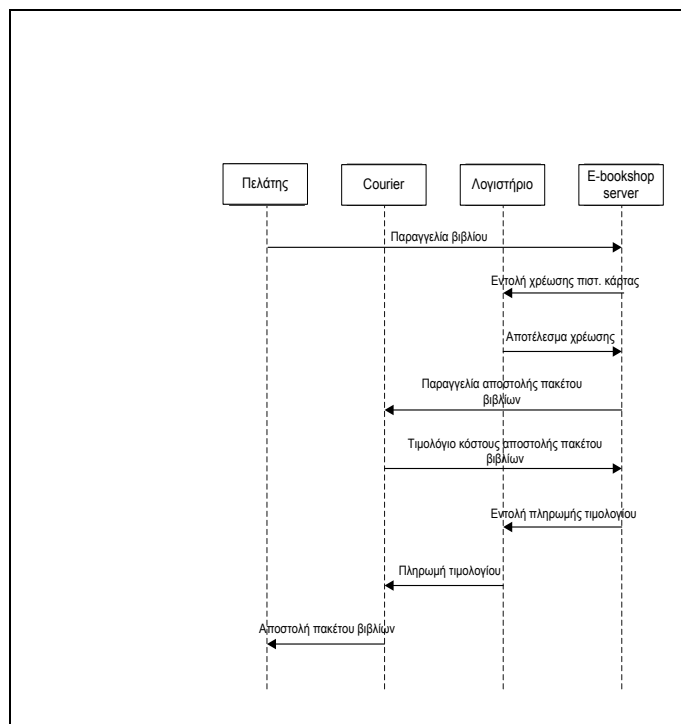
Άσκηση 3

Φτιάξτε ένα διάγραμμα καταστάσεων σε UML για ένα απλό σύστημα κλιματισμού αυτοκινήτου. Ο εξωτερικός αέρας μπορεί να ζεσταθεί – με την ενεργοποίηση ενός θερμαντικού σώματος – ή να ψυχθεί – με την ενεργοποίηση ενός ψυκτικού σώματος – πριν να μπει στην καμπίνα του αυτοκινήτου, ώστε να επιτευχθεί η επιθυμητή θερμοκρασία που έχει καθορίσει ο οδηγός. Όσο η θερμοκρασία παραμένει η επιθυμητή, τόσο το ψυκτικό όσο και το θερμαντικό σώμα είναι απενεργοποιημένα. Κάθε φορά που η θερμοκρασία αποκλίνει από την επιθυμητή ενεργοποιείται και πάλι το θερμαντικό ή το ψυκτικό σώμα.

UML: Διαγράμματα Συνεργασίας

Άσκηση 1

Δίνεται το παρακάτω διάγραμμα αλληλουχίας για μια περίπτωση χρήσης ενός συστήματος ηλεκτρονικού βιβλιοπωλείου:



Κατασκευάστε το διάγραμμα συνεργασίας για την παραπάνω περίπτωση χρήσης.

UML: Διαγράμματα Δραστηριοτήτων

Άσκηση 1

Κατασκευάστε το διάγραμμα δραστηριοτήτων για το παρακάτω σενάριο καταστήματος ενοικίασης DVD.

Μπαίνοντας στο κατάστημα ο πελάτης, κατευθύνεται στα ράφια όπου είναι τοποθετημένα τα DVD ή τα παιχνίδια και μετά από αναζήτηση επιλέγει ένα ή περισσότερα τα οποία προσκομίζει στο ταμείο για ενοικίαση. Εναλλακτικά, μπορεί να πραγματοποιήσει παρόμοια αναζήτηση ηλεκτρονικά, σε τέσσερα τερματικά που υπάρχουν διαθέσιμα σε μία πλευρά του καταστήματος, καταχωρώντας την επιλογή του σε μία κράτηση στον κωδικό του, την οποία στη συνέχεια ζητάει για ενοικίαση από τον ταμία.

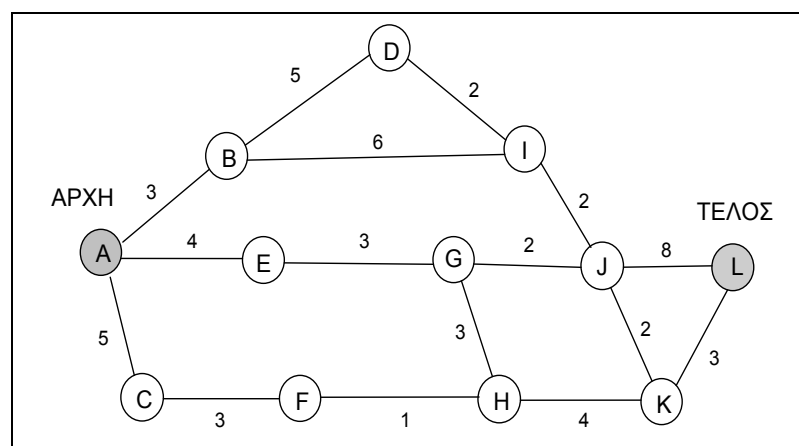
Ο ταμίας ξεκινάει την ενοικίαση, αφού ζητήσει τον κωδικό του πελάτη, τον οποίο εισάγει στο σύστημα για επιβεβαίωση. Εισάγει ένα-ένα τα προϊόντα που έχει επιλέξει ο πελάτης, και το σύστημα εμφανίζει τον τίτλο, της αξία, και το μέγιστο σε ημέρες ενοικίασης του καθενός, εμφανίζοντας προοδευτικά την συνολική αξία.

Τελειώνοντας, την εισαγωγή της ενοικίασης, ενημερώνει τον πελάτη για την αξία και του ζητάει επιβεβαίωση. Ο πελάτης ανταποκρίνεται καταβάλλοντας το τίμημα της αξίας της ενοικίασης. Ο ταμίας του παραδίδει τα προϊόντα της ενοικίασης, τα ρέστα, καθώς και μία αναλυτική εκτυπωμένη κατάσταση της ενοικίασης. Έτσι, τελειώνει και καταχωρείται η ενοικίαση στο σύστημα, ενημερώνοντας ταυτόχρονα και το ταμείο καθώς και την αποθήκη του καταστήματος. Σημειώνεται ότι ο πελάτης έχει τη δυνατότητα να πληρώσει με μετρητά ή με πιστωτική κάρτα. Επίσης, ο πελάτης έχει τη δυνατότητα να έχει προπιστώσει κάποιο ποσό στο λογαριασμό του στο DVD-club και να επιλέξει πληρωμή με αντίστοιχη μείωση του ποσού αυτού.

Χρονοπρογραμματισμός και κοστολόγηση έργου

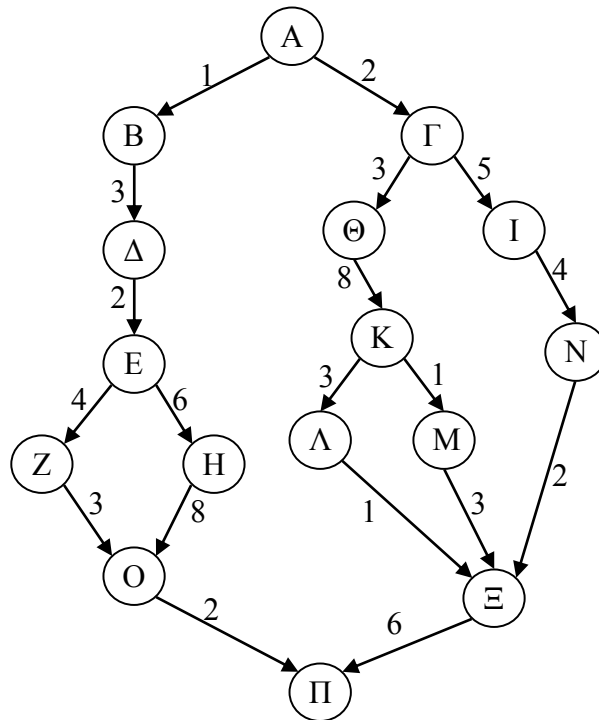
Άσκηση 1

Σας δίνεται ο παρακάτω γράφος δραστηριοτήτων για ένα έργο ανάπτυξης λογισμικού. Για κάθε δραστηριότητα υπολογίστε τον νωρίτερο δυνατό χρόνο εκκίνησης, τον μεταγενέστερο δυνατό χρόνο εκκίνησης και τον αδρανή χρόνο. Επιπλέον βρείτε το κρίσιμο μονοπάτι και υπολογίστε τη χρονική διάρκεια του έργου.



Άσκηση 2

Σας δίνεται ο παρακάτω γράφος εκτέλεσης δραστηριοτήτων.



- Υπολογίστε το νωρίτερο δυνατό χρόνο εκκίνησης κάθε δραστηριότητας, το μεταγενέστερο δυνατό χρόνο εκκίνησης κάθε δραστηριότητας, το κρίσιμο μονοπάτι, το διαθέσιμο χρονικό περιθώριο (slack) κάθε δραστηριότητας και τη χρονική διάρκεια του έργου
- Υποθέστε ότι έχετε 4 προγραμματιστές και ότι κάθε δραστηριότητα απαιτεί έναν από αυτούς καθ' όλη τη διάρκειά της. Υποθέστε επίσης ότι υπάρχει τρόπος ανάθεσης των προγραμματιστών σε δραστηριότητες ούτως ώστε το έργο να μην καθυστερήσει. Ποιο είναι το ποσοστό αξιοποίησης των προγραμματιστών σας (ώρες που αξιοποιείτε τους προγραμματιστές σας προς ώρες που τους πληρώνετε) στη διάρκεια του έργου;

Άσκηση 3

Η ανάλυση απαιτήσεων για τη βάση δεδομένων ενός συστήματος αυτοματισμού γραφείου έδειξε ότι αποτελείται από 4 τμήματα με τα ακόλουθα εκτιμώμενα μεγέθη:

Εισαγωγή δεδομένων	0.6 KDSI
Ανανέωση δεδομένων	0.6 KDSI
Ερωτήματα	0.8 KDSI
Δημιουργία αναφορών	1.0 KDSI

Το έργο εκτιμήθηκε ως οργανικό (συνεπώς $a=3.2$ και $b=1.05$). Ο υπεύθυνος του έργου βαθμολόγησε τις ιδιαιτερότητές του ως εξής:

Χαρακτηριστικό	Επίπεδο	Πολλαπλασιαστής
Πολυπλοκότητα	Υψηλή	1.15
Αποθήκευση	Υψηλή	1.06
Εμπειρία	Χαμηλή	1.13

Ικανότητα Προγραμματιστών Χαμηλή 1.17

Οι υπόλοιποι παράγοντες έχουν ονομαστική συνεισφορά.

Υπολογίστε:

- Την προσπάθεια που απαιτείται για τη διεκπεραίωση του έργου.
- Το χρονικό διάστημα που απαιτείται για την ολοκλήρωση του έργου.
- Τον αριθμό των εργαζομένων που πρέπει να απασχοληθούν στο έργο.
- Μια τεχνική για την εκτίμηση της πολυπλοκότητας του έργου είναι αυτή των βαθμών πολυπλοκότητας (function points). Σύμφωνα με την τεχνική αυτή, η πολυπλοκότητα του έργου καθορίζεται από 5 παράγοντες και καθήκον του διαχειριστή του έργου είναι να αναθέσει ένα βάρος σε καθέναν από αυτούς τους παράγοντες. Το άθροισμα των βαρών είναι ο συνολικός αριθμός βαθμών πολυπλοκότητας του έργου, ο οποίος μπορεί να πολλαπλασιαστεί με μια εκτίμηση των KDSI που απαιτούνται ανά βαθμό πολυπλοκότητας ώστε να προσεγγισθεί το μέγεθος του έργου. Έστω ότι απαιτούνται 70 DSI ανά βαθμό πολυπλοκότητας, και οι 5 παράγοντες για ένα έργο έχουν βαθμολογηθεί ως εξής:

<u>Χαρακτηριστικό</u>	<u>Βάρος</u>
Αριθμός εισόδων	4
Αριθμός εξόδων	5
Αριθμός ερωτημάτων	4
Αριθμός αρχείων	10
Αριθμός διεπαφών	7

Υπολογίστε το μέγεθος του συγκεκριμένου έργου.

Άσκηση 4

Εκτιμήσετε την προσπάθεια που απαιτείται για την ολοκλήρωση ενός έργου του οποίου το μέγεθος υπολογίζεται σε 50000 γραμμές κώδικα:

α) Σύμφωνα με το μοντέλο Walston/Felix.

β) Σύμφωνα με το μοντέλο Bailey and Basili.

γ) Χρησιμοποιώντας την τεχνική Delphi, τις εκτιμήσεις από τα (α) και (β) και μια επιπλέον εκτίμηση ειδικού που υπολογίζει την προσπάθεια σε 400 ανθρωπομήνες.

δ) Για καθεμιά από τις παραπάνω εκτιμήσεις, ποια θα είναι η χρονική διάρκεια του έργου αν εργάζεται σε αυτό μια ομάδα 12 ατόμων; Θεωρήστε ότι και τα 12 άτομα μπορούν να εργάζονται αποδοτικά παράλληλα και ανεξάρτητα σε όλη τη διάρκεια του έργου.

Μετρικές Λογισμικού

Άσκηση 1

Σας δίνεται το ακόλουθο τμήμα κώδικα.

```
class X {
public:
```

```

void A() {}
};

class Y {
public:
void B() {}
};

class Z {
public:
void C() {}
};

class Q : public X, protected Y,
private Z {
public:
void Test() {}
};

class R : public Q {
public:
void Test2() {}
};

```

- Δώστε τον πίνακα ορατότητας κάθε μεθόδου από άλλες κλάσεις.
- Υπολογίστε την ορατότητα κάθε μεθόδου ($V(Mm,i)$). Υπενθυμίζεται ότι η ορατότητα κάθε μεθόδου υπολογίζεται ως ο αριθμός των κλάσεων (πέραν αυτής στην οποία ορίζεται) από τις οποίες είναι προσπελάσιμη, δια του αριθμού των κλάσεων του προγράμματος (πέραν αυτής στην οποία ορίζεται).
- Υπολογίστε το βαθμό απόκρυψης μεθόδων (MHF) για το πρόγραμμα. Υπενθυμίζεται ότι ο βαθμός απόκρυψης μεθόδων είναι ίσος με το άθροισμα των όρων $(1 - V(Mm,i))$ για κάθε μέθοδο m κάθε κλάσης i , προς το συνολικό αριθμό μεθόδων.
- Ο βαθμός πολυμορφισμού (PF) δίνεται από τον ακόλουθο τύπο:

$$PF = \frac{\sum_{i=1}^{TC} M_o(C_i)}{\sum_{i=1}^{TC} (M_n(C_i) DC(C_i))}$$

όπου $M_o(C_i)$ είναι ο αριθμός των μεθόδων που γίνονται override στην κλάση C_i , $M_n(C_i)$ ο αριθμός των νέων μεθόδων που ορίζονται στην κλάση C_i , $DC(C_i)$ ο αριθμός των απογόνων της κλάσης C_i στο δέντρο ιεραρχίας και TC ο αριθμός των κλάσεων στο σύστημα. Ποια είναι η φυσική σημασία του παρανομαστή; Ποια είναι η φυσική σημασία του βαθμού πολυμορφισμού (PF);

Άσκηση 2

Σας δίδονται οι δύο παρακάτω κλάσεις. Υπολογίστε το σύνολο απόκρισης (Response Set – RS) της καθεμιάς

<pre>public class A { private B _aB; public void methodA1() { return _aB.methodB1(); } public void methodA2(C aC) { return aC.methodC1(); } }</pre>	<pre>public class A { private B _aB; public void methodA1() { return _aB.methodB1(); } public void methodA2() { return _aB.methodB1(); } }</pre>
---	--

Άσκηση 3

Σας δίνεται ο παρακάτω κώδικας. Υπολογίστε:

- Τον ζυγισμένο αριθμό μεθόδων ανά κλάση (Weighted Methods per Class – WMC)
- Βάθος του δέντρου κληρονομικότητας (Depth of Inheritance Tree – DIT)
- Αριθμό παιδιών (Number of Children – NOC)
- Συσχέτιση μεταξύ κλάσεων αντικειμένων (Coupling between Object Classes – CBO)
- Έλλειψη συνοχής στις μεθόδους (Lack of Cohesion in Methods – LCOM)

```

class person{
    char* name;
    char* ssn;
public:
    person(){
        name = new char[NAMELENGTH];
        ssn = new char[SSNLENGTH];
    }
    ~person(){delete name; delete ssn;}
    void addName(char* newname){strcpy(name, newname);}
    void addSsn(char* newssn){strcpy(ssn, newssn);}
    char* getName(){return name;}
    void virtual display(){cout << "the person's name is" << name;}
};

class student public person {
    float gpa;
public:
    void addGpa(float newgpa){gpa = newgpa;}
    void display(){cout<<"the student's name is" << getName() <<
        " and gpa is " << gpa;}
};

class personlist {
    person* list[MAX];
    int listIndex;
public:
    personlist(){listIndex = 0;}
    void addPerson(char* newname, char*newssn){
        list[listIndex]=new person;
        list[listIndex]->addName(newname);
        list[listIndex]->addSsn(newssn);
        listIndex++;
    }
    void addStudent(char* newname, char* newssn, float gpa){
        student* temp = new student;
        temp->addName(newname); temp->addSsn(newssn);
        temp->addGpa(newgpa);list[listIndex++]=temp;}
    void display() {
        int j;
        for (j=0; j<listIndex; j++) list[j]->display();}
};

```

Άσκηση 4

Σας δίνεται ο παρακάτω κώδικας:

```

public class A {
    private int f1, f2, f3, f4, f5;

    public void method1 () {
        f1 = f2 * 3;
    }

    public void method2 () {
        f2 = f3 / 4;
    }

    public void method3 () {
        f4 = f4 + f5;
    }
}

```

α) Υπολογίστε τη μετρική LCOM1 για τον κώδικα. Θυμηθείτε ότι $LCOM1 = \max(|P|$

- $|Q|, 0)$, όπου P είναι το σύνολο των ζευγαριών μεθόδων που προσπελαύνουν διακριτά μεταξύ τους σύνολα χαρακτηριστικών, ενώ Q είναι το σύνολο των ζευγαριών μεθόδων που δεν προσπελαύνουν διακριτά μεταξύ τους σύνολα χαρακτηριστικών.

β) Τι είναι συνοχή; Είναι επιθυμητή υψηλή ή χαμηλή συνοχή κατά τη διαδικασία σχεδίασης του λογισμικού;

Άσκηση 5

Ο βαθμός πολυμορφισμού (μία από τις μετρικές Mood) υπολογίζεται από την ακόλουθη σχέση:

$$PF = \frac{\sum_{i=1}^{TC} M_o(C_i)}{\sum_{i=1}^{TC} (M_n(C_i)DC(C_i))}$$

α) Περιγράψτε τους όρους της σχέσης και εξηγήστε τη φυσική της σημασία.

β) Υπολογίστε το βαθμό πολυμορφισμού για τον ακόλουθο κώδικα.

```
class BasicComponent {
protected:
    char *_name;
    Widget _w;
    BasicComponent( const char *);
public:
    virtual ~BasicComponent();
    virtual void manage();
    virtual void unmanage();
    const Widget baseWidget();
};

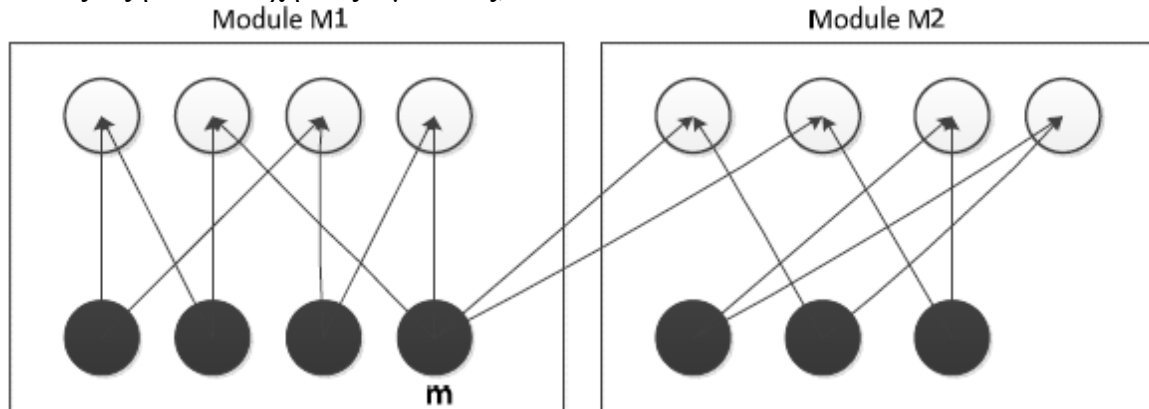
class UIComponent : public BasicComponent {
private:
    static void widgetDestroyedCallback
        ( Widget, XtPointer, XtPointer );
protected:
    UIComponent ( const char * );
    void installDestroyHandler();
    virtual void widgetDestroyed();
    void setDefaultResources
        ( const Widget , const String *);
    void getResources
        ( const XtResourceList, const int );
public:
    virtual ~UIComponent();
    virtual void manage();
    virtual const char *const className();
};

class Application : public UIComponent {
    friend void main ( int, char ** );
    friend class MainWindow;
private:
    void registerWindow ( MainWindow * );
    void unregisterWindow ( MainWindow * );
protected:
    Display *_display;
    XtAppContext _appContext;
    virtual void initialize ( int *, char ** );
    virtual void handleEvents();
    char *_applicationClass;
    MainWindow **_windows;
    int _numWindows;
public:
    Application ( char * );
    virtual ~Application();
    void manage();
    void unmanage();
    void iconify();
    Display *display();
    XtAppContext appContext();
    const char *applicationClass();
    virtual const char *const className();
};

class Clock : public UIComponent {
private:
    int _delta;
    XtIntervalId _id;
    virtual void timeout();
    virtual void speedChanged ( int );
    static void timeoutCallback
        ( XtPointer, XtIntervalId * );
    static void speedChangedCallback
        ( Widget, XtPointer, XtPointer );
protected:
    virtual void tick();
public:
    Clock ( Widget, char *, int, int );
    ~Clock ();
    void stop();
    void pulse();
    void start();
    virtual const char *const className();
};
```

Άσκηση 6

Η παρακάτω εικόνα αναπαριστά δύο μονάδες λογισμικού M1 και M2. Οι μαύροι κύκλοι αντιστοιχούν σε μεθόδους (methods) και οι λευκοί σε χαρακτηριστικά (attributes). Τα βέλη σημαίνουν ότι η μέθοδος από την οποία ξεκινά το βέλος χρησιμοποιεί το χαρακτηριστικό στο οποίο καταλήγει το βέλος. Αν η μέθοδος *m* μεταφερθεί στο M2, τι θα ισχύει για τη μέση σύζευξη και συνοχή στις 2 μονάδες;



- Δε θα υπάρξει αλλαγή
- Η μέση συνοχή θα αυξηθεί αλλά η μέση σύζευξη θα μειωθεί
- Και η μέση συνοχή και η μέση σύζευξη θα μειωθούν
- Και η μέση συνοχή και η μέση σύζευξη θα αυξηθούν

Έλεγχος

Άσκηση 1

Ο Tom είναι ο επιβλέπων ομάδας που αναπτύσσει ένα σύστημα λογισμικού. Προκειμένου να εκτιμήσει την ποιότητα του συστήματος που αναπτύσσεται, εμφυτεύει σφάλματα στο λογισμικό ώστε να προσεγγίσει τον αριθμό των εγγενών σφαλμάτων που παραμένουν στο σύστημα. Ο Tom έχει στη διάθεσή του δύο ομάδες ελέγχου. Της 1ης ομάδας ηγείται ο David και της 2ης ο Daniel. Υποθέστε ότι έχουν εμφυτευθεί 50 σφάλματα. Κατά τη διάρκεια του ελέγχου από την ομάδα του David ανακαλύπτονται 70 σφάλματα, 40 από τα οποία είναι εμφυτευμένα.

- Ποια είναι η εκτίμηση του ποσοστού των εγγενών σφαλμάτων που απομένουν στον κώδικα;
- Ποια είναι η εκτίμηση του αριθμού των εγγενών σφαλμάτων που απομένουν στον κώδικα;

Ο ίδιος κώδικας ελέγχεται από την ομάδα του Daniel. Η ομάδα του ανακαλύπτει ένα σύνολο 50 σφαλμάτων. 35 από τα λάθη που βρέθηκαν από την ομάδα του David βρίσκονται και από την ομάδα του Daniel

- Χρησιμοποιώντας τα δεδομένα από την ομάδα του Daniel, ποια είναι η εκτίμηση του αριθμού των εγγενών λαθών που απομένουν;
- Ποια είναι η αποδοτικότητα της ομάδας του David και ποια η αποδοτικότητα της

ομάδας του Daniel;

- Χρησιμοποιώντας τα δεδομένα και από τις δύο ομάδες ελέγχου, ποια είναι η εκτίμηση του συνολικού αριθμού σφαλμάτων.

Άσκηση 2

Υποθέστε ότι 39 σφάλματα έχουν εμφυτευθεί σε ένα τμήμα λογισμικού. Ο έλεγχος αποκάλυψε 32 από τα εμφυτευμένα σφάλματα, χωρίς να βρεθεί κανένα εγγενές σφάλμα.

- Ποιος είναι ο βαθμός εμπιστοσύνης ότι το συγκεκριμένο τμήμα λογισμικού είναι άψογο (χωρίς εγγενή σφάλματα);
- Πόσα εμφυτευμένα σφάλματα θα έπρεπε να βρεθούν, χωρίς να βρεθεί κανένα εγγενές σφάλμα, ώστε ο βαθμός εμπιστοσύνης να είναι 90%;

Άσκηση 3

Σας δίνεται το ακόλουθο τμήμα κώδικα:

```
void SE_2007(int x,y) {
    int a, b;
    a = 0;
    if (x > 0) {
        a = 4;
    }
    b = 4 * a;
    if (x <= 10) {
        if (y > 10) {
            b = 3 * a;
        }
    }
    printf("b = %d\n", b);
}
```

- Σχεδιάστε το γράφο ροής ελέγχου (flowchart) του κώδικα.
- Υπολογίστε την κυκλωματική πολυπλοκότητα του κώδικα.
- Δώστε σενάρια ελέγχου για τον κώδικα ώστε να επιτυγχάνεται κάλυψη εντολών (C0). Προσπαθήστε να πετύχετε κάλυψη C0 με τα λιγότερα δυνατά σενάρια ελέγχου.
- Δώστε σενάρια ελέγχου για τον κώδικα ώστε να επιτυγχάνεται κάλυψη διακλαδώσεων (C1). Προσπαθήστε να πετύχετε κάλυψη C1 με τα λιγότερα δυνατά σενάρια ελέγχου.
- Έστω ότι μια ομάδα ελέγχου εμφυτεύει λάθη στον κώδικα ώστε μετά το πέρας του ελέγχου να μπορεί να εκτιμήσει το βαθμό εμπιστοσύνης στο λογισμικό. Πόσα λάθη θα πρέπει να εμφυτευθούν ώστε, εάν εντοπιστούν όλα χωρίς να εντοπιστεί κάποιο εγγενές λάθος, να υπάρχει βεβαιότητα 95% ότι το πρόγραμμα είναι απόλυτα σωστό;

Άσκηση 4

Σας δίνεται ο παρακάτω κώδικας:

```
void characterize(int p, int q) {  
    if (p + q > 100)  
        printf("Sum is huge\n");  
    if (p > 50)  
        printf("P is quite large\n");  
    if (q < 10)  
        printf("Q is rather small\n");  
}
```

α) Υπολογίστε την κυκλωματική πολυπλοκότητα του κώδικα με τουλάχιστον 2 διαφορετικούς τρόπους (δείξτε τους 2 διαφορετικούς τρόπους στην απάντησή σας).

β) Εξηγήστε πώς η κυκλωματική πολυπλοκότητα που υπολογίσατε στο βήμα (α) σχετίζεται με τη διαδικασία ελέγχου.

γ) Δώστε ένα σύνολο σεναρίων ελέγχου βασιζόμενοι στην απάντησή σας στο (β).

Άσκηση 5

Σας δίνεται η παρακάτω συνάρτηση quicksort.

```
void quicksort(list_t list[], int m, int n) {  
    if (m < n) {  
        i=m, j=n+1, k=list[m].key;  
        do {  
            do  
                i=i+1;  
            while (list[i].key < k);  
            do  
                j=j-1;  
            while (list[j].key > k);  
            if (i < j)  
                swap(list[i], list[j]);  
        } while (i < j);  
        swap (list[m], list[j]);  
        quicksort(list, m, j-1);  
        quicksort(list, j+1, n);  
    }  
}
```

α) Υπολογίστε την κυκλωματική πολυπλοκότητα με 2 τουλάχιστον διαφορετικούς τρόπους (δείξτε τους τρόπους στην απάντησή σας).

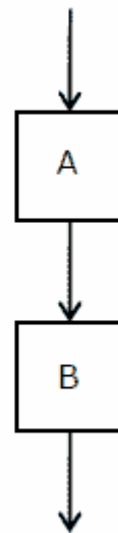
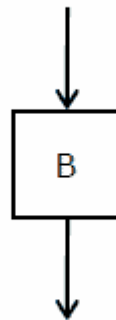
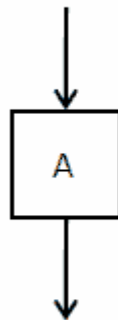
β) Εξηγήστε πώς ακριβώς η κυκλωματική πολυπλοκότητα που υπολογίσατε στο (α) σχετίζεται με τη διαδικασία ελέγχου.

γ) Δώστε ένα σύνολο σεναρίων ελέγχου βασιζόμενοι στην απάντησή σας στο (β).

Άσκηση 6

Η κυκλωματική πολυπλοκότητα καθενός από τα modules λογισμικού A και B στην εικόνα είναι 10. Ποια είναι η κυκλωματική πολυπλοκότητα της σύνθεσης σε ακολουθία των A και B που φαίνεται στο δεξί μέρος της εικόνας;

- α) 19
- β) 21
- γ) 20
- δ) 10

**Άσκηση 7**

Το παρακάτω πρόγραμμα πρέπει να ελεγχθεί για κάλυψη εντολών

```

if (a== b) {
    S1;
    exit;
}
else
    if (c== d)
        S2;
        else {
            S3;
            exit;
        }
    S4;

```

Τα σενάρια ελέγχου T1, T2, T3 και T4 που δίνονται παρακάτω εκφράζονται ως ιδιότητες που ικανοποιούνται από τις τιμές των μεταβλητών a, b, c και d (χωρίς να δίνονται οι ακριβείς τιμές των μεταβλητών).

T1: Τα a, b, c και d έχουν την ίδια τιμή

T2: Τα a, b, c και d έχουν διαφορετικές μεταξύ τους τιμές

T3: a = b και c != d

T4: a !=b και c = d.

Ποια από τις παρακάτω επιλογές σεναρίων ελέγχου είναι η ελάχιστη που διασφαλίζει κάλυψη εντολών κατά των έλεγχου;

- α) T1, T2, T3
- β) T2, T4
- γ) T3, T4
- δ) T1, T2, T4