

**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ (Τ.Ε.Ι.) ΚΑΒΑΛΑΣ**  
**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ**  
**ΤΜΗΜΑ ΒΙΟΜΗΧΑΝΙΚΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ**  
**ΜΑΘΗΜΑ**  
**«ΔΙΑΧΕΙΡΙΣΗ ΕΡΓΩΝ ΛΟΓΙΣΜΙΚΟΥ-ΠΟΙΟΤΗΤΑ ΛΟΓΙΣΜΙΚΟΥ»**  
**Χειμερινό Εξάμηνο 2009-2010**  
**Τελική Εξέταση Α΄ Περιόδου – 4 Φεβρουαρίου 2010**  
**Εισηγητής : Δρ. Παχίδης Θεόδωρος**

Όνοματεπώνυμο: \_\_\_\_\_ Α.Μ. \_\_\_\_\_ Εξάμ \_\_\_\_\_

Διάρκεια Εξέτασης: 2 ώρες

**B**

**Θέμα 1ο:**

Σε μια ομάδα ανάπτυξης λογισμικού ανατίθεται ένα έργο λογισμικού. Το έργο έχει ως στόχο τη δημιουργία ενός ολοκληρωμένου συστήματος λογισμικού που να μπορεί να διαχειριστεί από ένα κέντρο ελέγχου ένα μεγάλο αριθμό από φορτηγά αυτοκίνητα που ανήκουν στην εταιρεία-πελάτη. Ο διαχειριστής του έργου σχεδιάζει ένα αρχικό διάγραμμα Gantt και ένα διάγραμμα PERT. Για τη σχεδίαση τους στηρίχθηκε στον εξής αρχικό πίνακα εργασιών στον οποίο συμπλήρωσε την εκτιμώμενη διάρκεια κάθε εργασίας καθώς και την εξάρτηση της κάθε εργασίας από άλλες προηγούμενες.

ΕΡΓΑΣΙΑ	ΔΙΑΡΚΕΙΑ (ΗΜΕΡΕΣ)	ΕΞΑΡΤΗΣΗ
T1. Έναρξη – προετοιμασία	30	
T2. Συλλογή απαιτήσεων	45	T1
T3. Δημιουργία προδιαγραφών	25	T2
T4. Ανάλυση – Σχεδίαση αρχιτεκτονικής	30	T3
T5. Λεπτομερής σχεδίαση	40	T4
T6. Κωδικοποίηση (δημιουργία μονάδων)	50	T5
T7. Έλεγχος μονάδων	60	T6
T8. Τροποποίηση - έλεγχος υπάρχοντος λογισμικού	90	T4
T9. Διόρθωση σφαλμάτων	60	T8
T10. Συνένωση μονάδων – Έλεγχος συνένωσης	35	T6
T11. Έλεγχος συστήματος	40	T10
T12. Διόρθωση σφαλμάτων	35	T11
T13. Αξιολόγηση ολοκληρ. έκδοσης - Αποδοχή	40	T12
T14. Παράδοση προϊόντος – εγκατάσταση - εκπαίδευση	10	T13

Με τη βοήθεια του παραπάνω πίνακα:

α) να σχεδιαστεί το διάγραμμα PERT. (10 μονάδες)

β) Να βρεθεί το κρίσιμο μονοπάτι (ΥΠΟΔΕΙΞΗ: να δημιουργηθεί πίνακας στον οποίο να φαίνονται αναλυτικά οι μετρούμενοι χρόνοι για κάθε εργασία, να εξηγηθεί πως προκύπτει το κρίσιμο μονοπάτι και να δειχθεί επάνω στο διάγραμμα PERT). (12 μονάδες)

γ) Να σχεδιαστεί το διάγραμμα Gantt. (8 μονάδες)

## Θέμα 2ο:

Σε μια χημική βιομηχανία τα υγρά χημικά απόβλητα πριν πέσουν στην θάλασσα ουδετεροποιούνται από όξινα ή αλκαλικά διαλύματα. Η διεργασία αφορά τη συνεχή ροή ενός όξινου ή αλκαλικού διαλύματος (ανάλογα με το pH των αποβλήτων) μέσα στο δοχείο των αποβλήτων μέχρις ότου η μετρούμενη τιμή με τη βοήθεια ενός πεχαμέτρου να γίνει 7 (ουδέτερο διάλυμα). Το σύστημα αποτελείται από δύο υπερυψωμένες δεξαμενές 10000 lt (όξινο διάλυμα, αλκαλικό διάλυμα), μία δεξαμενή 5000 lt (απόβλητα), η ελεύθερη ροή σε κάθε περίπτωση ελέγχεται από ηλεκτροβάνες ενώ η κατάσταση του συστήματος ελέγχεται από ένα μόνο υπάλληλο που παρακολουθεί την ένδειξη του πεχαμέτρου και τις στάθμες των δεξαμενών. Επίσης ένας αναδευτήρας έχει ως σκοπό να εξασφαλίζει την ομοιογένεια του υλικού στη δεξαμενή των αποβλήτων. Για τη διεργασία αυτή:

- 1) Καταγράψετε όλους τους κινδύνους που μπορούν να εμφανιστούν για το περιβάλλον, το προσωπικό και τις εγκαταστάσεις της βιομηχανίας. (5 μονάδες)
- 2) Επιλέξτε αιτιολογημένα τους δύο σημαντικότερους κατά τη γνώμη σας κινδύνους και προσδιορίστε για αυτούς (9 μονάδες):
  - Την πιθανότητα να συμβούν.
  - Τη δριμύτητα της ζημιάς που προκαλούν.
  - Την επικινδυνότητά τους (εκτίμηση επικινδυνότητας) και απεικονίστε την επικινδυνότητα αυτή σε διάγραμμα.
- 3) Για τους κινδύνους που επιλέξατε να γίνουν τα δένδρα λάθους. (6 μονάδες)

## Θέμα 3ο:

Για το παρακάτω τμήμα προγράμματος:

```
33 int low = 0; // low end of the search area
34 int high = size - 1; // high end of the search area
35 int middle = ( low + high + 1 ) / 2; // middle element
36 int location = -1; // return value; -1 if not found
37
38 while ( ( low <= high+1 ) && ( location == -1 ) ) // loop to search for element
39 {
40     // print remaining elements of vector to be searched
41     displaySubElements( low, high );
42
43     // output spaces for alignment
44     for ( int i = 0; i < middle; i++ )
45         cout << " ";
46
47     cout << " * " << endl; // indicate current middle
48
49     // if the element is found at the middle
50     if ( searchElement == data[ middle ] )
51         location = middle; // location is the current middle
52     else if ( searchElement < data[ middle ] ) // middle is too high
53         high = middle - 1; // eliminate the higher half
54     else // middle element is too low
55         low = middle + 1; // eliminate the lower half
56
57     middle = ( low + high + 1 ) / 2; // recalculate the middle
58 }
```

- 1) Να σχεδιαστεί το διάγραμμα ροής. (5 μονάδες)
- 2) Να σχεδιαστεί το γράφημα ροής. (5 μονάδες)
- 3) Να βρεθεί η κυκλωματική πολυπλοκότητα με δύο τρόπους που θα επιλέξετε. (5 μονάδες)
- 4) Να βρεθούν οι περιπτώσεις τεστ ανοικτού κουτιού για την κάλυψη αποφάσεων και συνθηκών. (15 μονάδες)

**Θέμα 4ο:**

Κατά την ανάπτυξη ενός έργου λογισμικού μέχρι και τη φάση ελέγχου του συστήματος με διαφορετικές μεθόδους αξιολόγησης του έργου βρέθηκε ο εξής αριθμός λαθών σύμφωνα με τον πίνακα:

Φάση	Λάθη Μικρής Σημασίας	Λάθη μέτριας Σημασίας	Σοβαρά Λάθη
Προσδιορισμός απαιτήσεων	10	8	2
Αρχιτεκτονική Σχεδίαση	8	4	4
Λεπτομερής Σχεδίαση	17	25	8
Κωδικοποίηση και Έλεγχος	20	16	4
Έλεγχος Συστήματος	48	35	17

Αν τα βάρη για τα μικρής σημασίας, μέτριας σημασίας και σοβαρά λάθη είναι 1, 2 και 4 αντίστοιχα, ο συνολικός κώδικας είναι 5548 SLOC και το συνολικό πλήθος των λαθών που βρέθηκαν μετά την παράδοση στον πελάτη είναι 24, να υπολογιστούν:

- 1) Οι δείκτες φάσης για κάθε φάση του κύκλου ζωής του έργου. (5 μονάδες)
- 2) Ο δείκτης λάθους όπως διαμορφώνεται μέχρι και την συγκεκριμένη φάση. (5 μονάδες)
- 3) Η αποδοτικότητα αφαίρεσης σφαλμάτων. (5 μονάδες)
- 4) Το εκτιμώμενο κόστος του λογισμικού σε € σύμφωνα με το μοντέλο COCOMO II και το υπομοντέλο πρώιμης σχεδίασης αν είναι  $B=1.2$ ,  $M=1.12$  και το κόστος / ανθρωπομήνα είναι 1500 €. (5 μονάδες)

**ΚΑΛΗ ΕΠΙΤΥΧΙΑ**