

Université Badji Mokhtar Faculté des technologies
département d'Informatique EMD QLT M1ILC durée 1H30mn

Questions (10pts)

1. En Ingénierie de logiciel, que représente d'après vous la « spécification » ?
2. A quels types de besoins essayent de répondre les modèles de cycle de vie qui sont proposés en général ?
3. Décrire deux cas de problèmes que l'on peut rencontrer lors d'un développement en équipe.
4. Pourquoi le développement logiciel est devenu de plus en plus un travail d'équipes de professionnels en logiciel ?
5. Citez les caractéristiques d'un logiciel
6. Citez les facteurs de non qualité.
7. Citer 4 Méthodes d'évaluation de qualité logicielle.
8. Pourquoi normaliser et citer les types de normes
9. Qu'apporte La démarche qualité dans l'entreprise en interne et Vis-à-vis de l'extérieur
10. Citer l'objectif des Logiciels de l'évaluation de la qualité citez un Logiciel

Exercice 1 (4pts)

Soit un programme qui lira au clavier l'heure, les minutes et les secondes, et il affichera l'heure qu'il sera une seconde plus tard. Par exemple si l'utilisateur tape 21 puis 32 puis 16, le programme doit répondre : "Dans une seconde, il sera 21 heures 32 minutes et 17 secondes".

1. Réaliser le teste boîte noire sur ce programme
2. donner les erreurs possibles.

Exercice2 (6 pts)

1. Donnez un graphe de contrôle associé au code source fourni. Donner un chemin d'exécution.
2. Votre graphe de contrôle a-t-il des possibilités de réduction ? Si oui, réduisez votre graphe de contrôle.

Lire (b,c,x)

If b<c

then begin

d:=2*b

f:=3*c

if x>0

then begin

 y:=x

 e:=c

 if (y=0)

 then begin

 a:=f-e

 while d<a

 begin

 d:=d+2

 end

 end

 end

else begin

 b:=b-1

end

end

Solution

Question 1 pt par question.

1- La "spécification" en ingénierie logicielle représente la description détaillée et formelle des exigences et des fonctionnalités d'un système logiciel. Elle définit ce que le logiciel doit faire, comment il doit le faire, et quelles sont les contraintes qui s'appliquent à lui.

2- Les modèles de cycle de vie en ingénierie logicielle visent à répondre à différents types de besoins, tels que les besoins fonctionnels, les besoins de qualité, les besoins de coût, les besoins de délai, et les besoins de communication avec les parties prenantes. Ils proposent des approches pour planifier, concevoir, développer, tester, déployer et entretenir le logiciel afin de satisfaire ces besoins.

3- Deux cas de problèmes courants lors du développement en équipes sont :

a) Les conflits de gestion de versions : Lorsque plusieurs développeurs travaillent sur le même code source, il peut y avoir des conflits lors de la fusion de leurs modifications dans le référentiel de code. Cela peut entraîner des erreurs et des incompatibilités.

b) La communication inefficace : Les équipes de développement doivent constamment communiquer pour coordonner leurs efforts. Les problèmes de communication peuvent entraîner des malentendus, des retards et une mauvaise compréhension des exigences.

4- Le développement logiciel est devenu de plus en plus un travail d'équipes de professionnels en logiciel en raison de la complexité croissante des systèmes logiciels. Les logiciels modernes sont de plus en plus interconnectés, comportent de nombreuses fonctionnalités, et doivent respecter des normes de qualité élevées. Le travail en équipe permet de répartir la charge de travail, de tirer parti des compétences spécialisées de chaque membre de l'équipe, et d'améliorer la qualité et l'efficacité du développement. De plus, la collaboration entre professionnels permet une meilleure gestion des risques et une réponse plus rapide aux changements et aux évolutions des besoins des utilisateurs.

5- Produit unique : conçu et fabriqué une seule fois, reproduit

Inusable : Défauts pas dus à l'usure mais proviennent de sa conception

Complexe : Le logiciel est fabriqué pour soulager l'humain d'un problème complexe ; il est donc par nature complexe

Invisible : Fabrication du logiciel est une activité purement intellectuelle avec une difficulté de perception de la notion de qualité du logiciel

Techniques non matures : Encore artisanal (fait à la main) malgré les progrès

6- Facteur de non qualité

Mauvaise spécification : Les spécifications des clients qui ne sont pas des informaticiens

Vague, incomplète, instables...

Mauvaise estimation : Fausse, oublis, précisions insuffisantes...

Mauvaise répartition des tâches : Organisation inadaptée, contraintes omises ou écart non détecté à temps

Mauvaise réalisation technique : Codage, tests, documentation

Problèmes humains : Mauvaise distribution des travaux, conflits ou rétention d'information

Manque d'expérience du métier de chef de projet : compétence inadaptés

7-méthode d'évaluation de la qualité

Université Badji Mokhtar Faculté des technologies

département d'Informatique EMD QLT M1ILC durée 1H30mn

- **Méthodes formelles :**
- **Métriques :**
- **Revue et Inspections :**
- **Auto-correction** (desk-checking) :
- **Lecture croisée** (author-reader cycle)
- **Revue** (walkthrough)
- **Revue structurée :**
- **Inspection :**

8- L'existence de normes non harmonisées pour des technologies semblables, dans des pays ou des régions différents, peut contribuer à: ce que l'on appelle des " obstacles techniques au commerce ". Les industries tournées vers l'exportation ont depuis longtemps senti la nécessité de s'accorder sur des normes mondiales pour aider à: rationaliser le processus des échanges internationaux. C'est cet objectif, justement, qui a présidé à: la création de l'ISO.

Quatre types de normes

1. Normes fondamentales

Règles en matière de terminologie, sigles, symboles, métrologie (exemple ISO 31 : grandeurs et unités)

2. Normes de spécifications

Caractéristiques, seuils de performance d'un produit ou d'un service (exemple : EN 2076-2 : Série aérospatiale - Lingots et pièces moulées en alliages d'aluminium et de magnésium - Spécification technique)

3. Normes d'analyse et d'essais

Méthodes et moyens pour la réalisation d'un essai sur un produit (exemple : ISO 6506-1 : Matériaux métalliques - Essai de dureté Brinell)

4. Normes d'organisation

Fonctions et relations organisationnelles à l'intérieur d'une entité (exemple : ISO 9001 : Systèmes de management de la qualité – Exigences)

9- l'intérêt pour l'entreprise?

La démarche qualité dans l'entreprise (telle que définie dans la série des normes ISO 9000) apporte :

1. En interne :

L'amélioration de la compétitivité et la diminution des coûts. En effet, une démarche qualité induit une optimisation et une rationalisation de l'organisation par la mobilisation des ressources de l'entreprise au service des clients, et par la réduction des défauts et des coûts associés. La démarche qualité est, par ailleurs, l'occasion de mobiliser l'ensemble du personnel de l'entreprise autour d'un projet d'entreprise.

2. Vis-à-vis de l'extérieur:

La réponse aux exigences des donneurs d'ordres, qui imposent de plus en plus la certification ISO 9000 à: leurs fournisseurs. Mais aussi la possibilité de se démarquer par rapport à: la concurrence, grâce à: une forte communication externe.

10- Logiciels de l'évaluation de la qualité

Objectif et fonctionnement

Les logiciels de l'évaluation de la qualité ont pour objectif principal l'analyse statique du code source d'une application grâce à des règles de codage et des métriques.

Le logiciel Sonar

Université Badji Mokhtar Faculté des technologies
département d'Informatique EMD QLT M1ILC durée 1H30mn

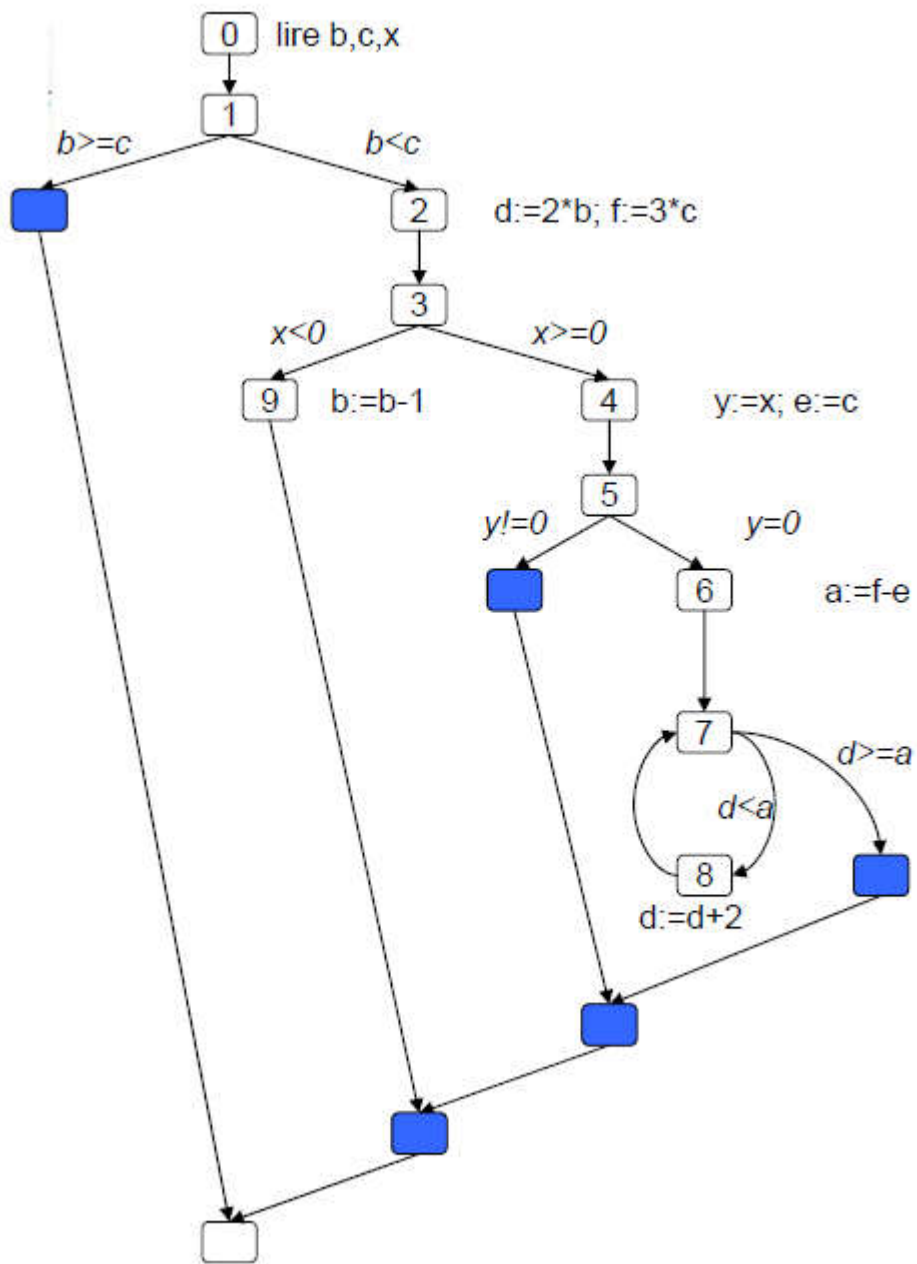
Exercice 1 (2 pts par question)

Donner des valeur et exécuter et donner des valeurs au limites pour les seconde ex 23, 59, 59 .

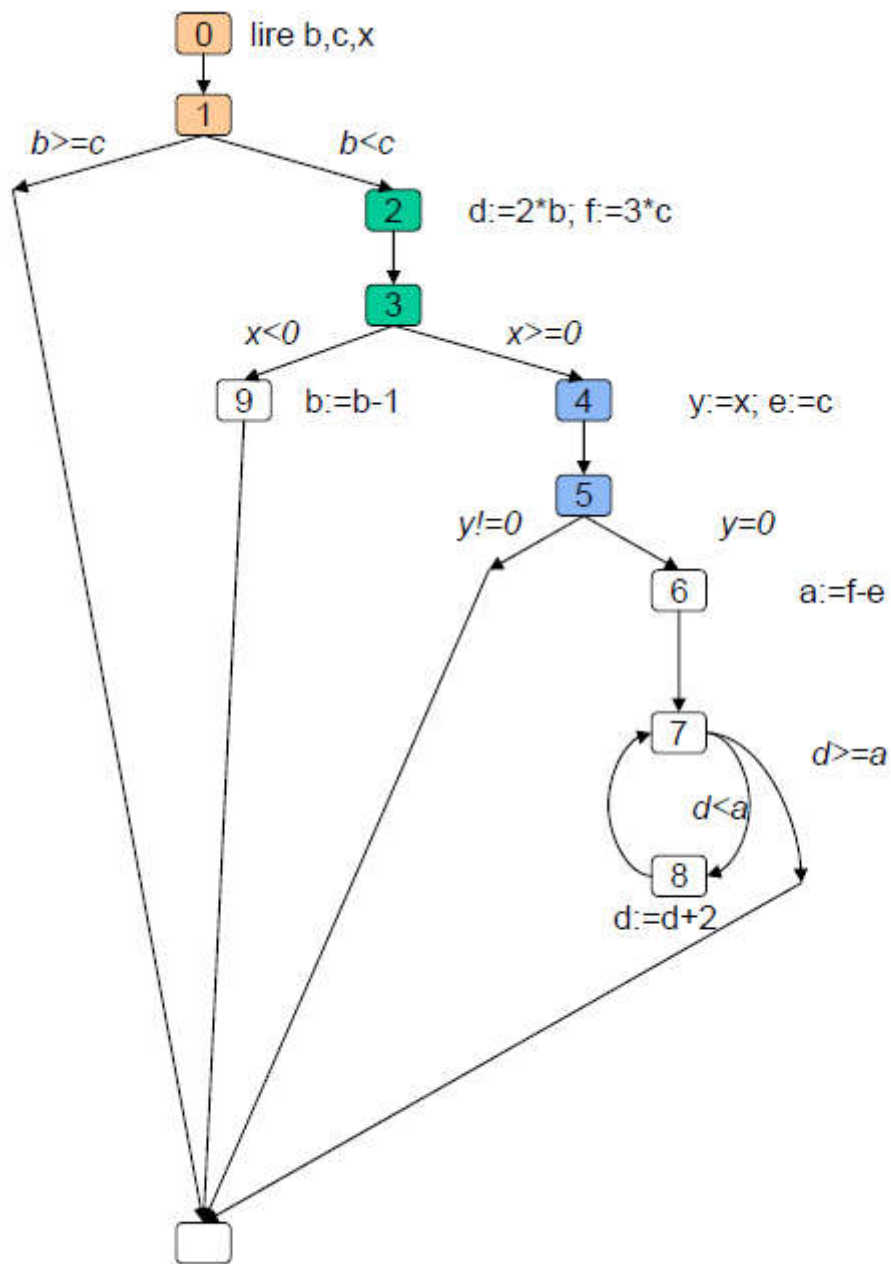
Les erreurs possible donné une valeur pus grande que 60 au niveau des minutes des secondes et plus grand que 23 pour les heures.

Exercice 2 (3pts pour le graphe de contrôle, 1 pt pour le chemin, 2 point pour le graphe réduit)

```
lire(b,c,x) /*0*/
if b<c /*1*/
then begin
  d :=2*b /*2*/
  f :=3*c
  if x>=0 /*3*/
  then begin
    y := x /*4*/
    e := c
    if (y=0) /*5*/
    then begin
      a :=f-e /*6*/
      while d<a /*7*/
      begin
        d:=d+2 /*8*/
      end
    end
  end
else begin
  b:=b-1/*9*/
end
end
```



2. .



3.

