

14 Software Maintenance 1



To: All Employees
From: c.bellman@shorsoft.net
Subject: Updated Policies for **Software Maintenance**

Good Morning Employees,
I understand that most departments are practicing **corrective maintenance**. This practice should continue, but I think it is **insufficient**. I believe we need to focus on **adaptive maintenance** as well. According to the **law of continuing change**, this will allow us to grow more rapidly.

Developing new software is important. But to stay competitive, we must **enhance** our existing software, too. New **releases** are the best way to keep customers interested in our products. This will require engineers to practice **perfective maintenance**. Always **repair** problems as soon as they are identified.

However, engineers must also remember the **law of increasing complexity**. If software becomes too complex, it becomes difficult to maintain. Engineers should know when to update and when to write a new program.

Unstructured code will no longer be tolerated. It causes confusion and makes further updates more difficult. Remember, **preventive maintenance** is the strongest software maintenance practice.

-Clinton Bellman
CEO, ShorSoft Corporation



Get ready!

1 Before you read the passage, talk about these questions.

- What are the benefits of software maintenance?
- What are some different types of software maintenance?

Reading

2 Read the memo. Then, choose the correct answers.

- What is the purpose of the memo?
 - to warn employees about maintenance risks
 - to reprimand employees who are not following maintenance procedures
 - to show techniques for software maintenance
 - to inform employees about new policies
- Which of the following is NOT something that the CEO wants engineers to do?
 - Avoid unstructured code.
 - Focus more on corrective maintenance.
 - Get existing software ready for new releases.
 - Increase preventive maintenance measures.
- According to the email, how can engineers enhance existing software?
 - creating less unstructured code
 - using templates from other software programs
 - repairing problems in the software
 - practicing corrective maintenance

Vocabulary

3 Match the phrases (1-8) with the definitions (A-H).

- ___ adaptive maintenance
 - ___ corrective maintenance
 - ___ law of continuing change
 - ___ law of increasing complexity
 - ___ perfective maintenance
 - ___ preventive maintenance
 - ___ software maintenance
 - ___ unstructured code
- the practice of accommodating new user requirements
 - the practice of repairing software faults
 - the format of a system with no clear order
 - the practice of making systems easier to maintain
 - the process of fixing faults and making improvements in software
 - states that a system should undergo modification until it is no longer cost-effective
 - the practice of updating software according to changes in environment
 - states that a structure becomes more complex with every change

5 Write a word that is similar in meaning to the underlined part.

- The attributes of an older system may be unsuitable or not strong enough to work on updated operating systems.
_ _ s _ _ t _ _ _ _ t
- A software engineer should fix any problems he or she finds in a code. _ _ p _ _ r
- Each new updated version of existing software should come with some modifications.
_ _ l _ a _ _
- Consumers of software are happy when engineers improve existing components in new versions of software products.
_ n _ _ n c e

5 Listen and read the memo again. Why does the CEO want engineers to perform adaptive maintenance?

Listening

6 Listen to a conversation between two engineers. Mark the following statements as true (T) or false (F).

- ___ The man wants to start with corrective maintenance.
- ___ The engineers are adding new functionality to old software.
- ___ The woman discovered unstructured code in the software.

7 Listen again and complete the conversation.

- Engineer 1: We have 1 _____ to do on that accounting software.
- Engineer 2: Yeah. I'm really 2 _____ all of that work.
- Engineer 1: Neither am I. But I think if we make a plan, we can save ourselves a lot of time.
- Engineer 2: That's a good idea. 3 _____ fix all of the problems with the software.
- Engineer 1: Okay. We can start with a round 4 _____, then.
- Engineer 2: Exactly. Next, we need to 5 _____ all of the code.
- Engineer 1: Why do we need to do that?
- Engineer 2: So that it can 6 _____

Speaking

8 With a partner, act out the roles below based on Task 7. Then, switch roles.

USE LANGUAGE SUCH AS:

We're going to be ...
First, we need to ...
Next ...

Student A: You are an engineer. Talk to Student B about:

- software maintenance that is needed
- reasons for particular types of maintenance

Student B: You are an engineer. Talk to Student A about software maintenance.

Writing

9 Use the conversation from Task 8 to complete the memo from a project manager to an engineer.

Hi Karen,

I'm pleased with the new programs your team is developing. However, I want you to shift your focus to software maintenance for a few weeks.

On the GrayWhite 9.7 software, we need _____.
This is because _____.

On the SP008 software, we need _____.
This is because _____.

On the FANFARE II software, we need _____.
This is because _____.

-Arnold

