Examination report – December 2013 series

2730-025 Software engineering

Section 1 – Areas of good performance
Syllabus reference: 1.6 / 1.7 – The need for software engineering – software process. The section on the waterfall model of software development was answered extremely well. Syllabus reference: 1.15 / 1.16 – Software specification – requirements analysis. Overall solutions were good although state diagrams were sometimes inaccurate and poorly labelled. Syllabus reference: 1.22 – Software design and implementation – design methodology. The answers to the meaning of a pseudo-code program and the designs of a program, using pseudo-code, were satisfactory. Syllabus reference: 1.25 / 1.33 / 1.34 – Programming practice and software tools. This section was answered satisfactorily. Syllabus reference: 1.42 / 1.43 – Software validation – testing. This section was answered satisfactorily. Syllabus reference: 1.48 / 1.49 – Programming languages. This section was answered satisfactorily. Syllabus reference: 1.58 – Programming languages. The descriptions of the function of the kernel, file storage system and shell, with reference to the Unix operating system were satisfactory.
Section 2 – Areas for development
Syllabus reference: 1.6 / 1.7 – The need for software engineering – software process. Most candidates failed to describe what the required documentation might contain with reference to any two of the stages of the waterfall model. Syllabus reference: 1.22 – software design and implantation – design methodology. Few candidates could describe two attributes of a well written pseudo-code program. Syllabus reference: 1.58 – Programming languages. Very few candidates could state the function of the standard Unix directories. Syllabus reference: 1.10 – Software specification – requirements analysis. Solutions to this section were often inaccurate or vague. Very few candidates could name a modelling tool with reference to systems modelling. Syllabus reference: 1.23 - Programming practice – and software tools. Candidates gave very little mention regarding being consistent with formatting and naming conventions.

Syllabus reference: 1.47 – Programming languages. Solutions were very poor. Candidates failed to adequately state the defining feature of each programming language type. The examples given for each of the languages listed were often incorrect.
Section 3 – Recommendations
Candidates need to improve their knowledge in the following areas: · System modelling. · Good programming practice. · The characteristics of the Functional, Logical, Visual and Scripted programming languages.