SOFTWARE PROJECT MANAGEMENT
TUTORIAL LETTER 203
FOR
INF3708

Assignment 03: Due date 02 April 2015

Unique nr: 594147
Marks weight: 40%

ASSIGNMENT 03 – SEMESTER 1

ASSIGNMENT 03	
Due date	02 April 2015
Study material	Hughes & Cotterell: Chapters 5, & 6
Total marks	60 marks = 100%
If your assignment is late, please DO NOT PHONE OR E-MAIL asking for an extension but include a note in your assignment stating the reason for the late submission and we will decide whether or not it will be marked.	

Complete this assignment and submit online via myUnisa as a .pdf document.

QUESTION 1: Questions on Chapter 5 (20 marks)

1.1 Provide the equation and identify the variables in Boehm's equation for calculating effort in the use of the COCOMO model. (4)

ANSWER:

Boehm's equation: Formula:

effort = c* (size)*^{k} (**1 mark)**

effort measured in person months – 152 working hours (**1 mark**)

size is measured in kdsi – thousands of delivered source code (**1 mark)**

c and k are constants **1 mark**

where effort is measured in pm, or the number of "person-months" consisting of units of 152 working hours, size is measured in *kdsi,* thousands of delivered source code instructions, and *c* and *k* are constants. The first step was to derive an estimate of the system size in terms of *kdsi*. The constants, *c* and *k*, depended on whether the system could be classified, in Boehm's terms, as "organic", "semi-detached" or "embedded". These related to the technical nature of the system and the development environment. *c* and *k* are constant values derived from the table to be read off.

1.2 Five systems with the following estimated lines of code were identified. Identify which can be completed in less than three years. (16)

System	Lines of code	System type
A	23557	Organic
B	18553	Organic
C	17014	Semi-detached

D	10572	Embedded
E	9568	Semi-detached

Table for Question 1: System details

ANSWER:

Use the following COCOMO constants available in the prescribed textbook:

System type	c	K
Organic	2.4	1.05
Semi-detached	3.0	1.12
Embedded	3.6	1.20

NOTE: The COCOMO constants table is to be given in case of exams, as part of exam question.

Calculations:

A	$=2.4*((23557/1000)^{1.05})$	=	66.21208	person months / 12 = years:	=	5.517673	years
B	$=2.4*((18553/1000)^{1.05})$	=	51.52833	person months / 12 = years:	=	4.294027	years
C	$=3.0*((17014/1000)^{1.12})$	=	71.71738	person months / 12 = years:	=	5.976448	years
D	$=3.6*((10572/1000)^{1.20})$	=	60.99456	person months / 12 = years:	=	5.082880	years
E	$=3.0*((9568/1000)^{1.12})$	=	37.63925	person months / 12 = years:	=	3.136604	years
	5 marks		**5 marks**			**5 marks**	

None will be completed in three years - **1 mark**

QUESTION 2: Questions on Chapter 5 (12 marks)

Allan Albrecht developed the IFPUG method that can be use to estimate a system size. Explain the steps in detail using your own example(s). (12)

ANSWER:

NOTE: This question was marked based on the student's answer. However, below is a solution sample:

"OWN EXAMPLE Sample"

The total function point (FP) count of a subsystem to be written consists of 30 unadjusted function points (UFP), calculated according to the IFPUG method developed by Allan Albrecht. The Project Manager (PM) discovered that one FP count did not include a reference to the Personnel file.

This file consists of the following 3 record types:

- general information;
- employment history; and
- performance information.

Twenty data types are to be referenced by the program.

If the file is an internal logical file (ILF), how many FPs must the PM add to the current FP count to make provision for the Personnel file? Explain all the steps in detail.

The **first** step is to determine which method to use. Since the IFPUG or Albrecht Function Point method has been used (and suggested in the question) to calculate the FPs thus far, it makes sense to continue in this vain.

The **second** step (when using IFPUG) is to identify what type of object is to be evaluated. Table 5.2 (page 115 of Hughes & Cotterell) indicates five External user types. According to the question, the additional object is of a ***logical internal file type, also referred to as an ILF or internal logical file***.

External user type	**Multiplier**		
	Low	Average	High
External input type	3	4	6
External output type	4	5	7
Logical internal file type	7	**10**	15
External interface file type	5	7	10
External enquiry type	3	4	6

Table 5.2 Albrecht complexity multipliers (p. 115)

The **third** step is to determine the complexity of the object (high, average or low). The method to do this is prescribed by IFPUG and contained in tables 5.3 (page 115). For logical internal files and external interface files, table 5.3 is used.

Number of record types	Number of data types		
	<20	**20-50**	>50
1	Low	Low	Average
2 to 5	Low	**Average**	High
>5	Average	High	High

Table 5.3 IFPUG file type complexity (p. 115)

Because the file consists of 3 record types and reference 20 data types, the complexity is average, the 2nd row and middle column in table 5.3.

In step **four** we have to identify the number of record types and data types. According to the question there are ***three record types*** and ***twenty data types*** to be referenced by the program. Reading from table 5.3, the file type complexity will thus be ***average.***

In step **five** we use table 5.2 to determine the complexity multiplier for a logical internal file type of average complexity, which is **10**.

Step **six** we multiply the multiplier with the number of files to be added (1) to determine the Fps: = 10 and add these to the existing Fps calculated (30) resulting in a new UFP count of 40.

Mark allocation:

- first step **1 mark**
- second step **2 mark**
- third step **2 mark**
- fourth step 2 marks (**1 mark** for three record types and twenty data types and **1 mark** for average read off.
- Fifth step **1 mark**
- Sixth step 4 marks (**2 mark** for 1 * 10 and **2 mark** for 30 + 10 = 40)

Question 3: Questions on Chapter 6 (5 marks)

"Planning does not only take place during the project start-up." Discuss this statement, referring to **when** and **why** planning takes place as it does. (5)

ANSWER:

NOTE: This question was marked based on the student's answer. However, below is a solution sample:

"The importance of ongoing planning throughout the project life cycle cannot be stressed enough – often neglect in this area may result in overall project failure or costly delays. Planning is an ongoing process of refinement with each iteration becoming more detailed and more accurate than the last. The emphasis and purpose of planning shifts over successive iterations.

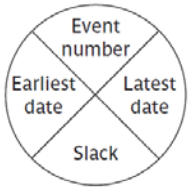
During the feasibility study and project start-up the main purpose of planning will be to estimate timescales and the risks of not achieving target completion date or keeping within budget. The emphasis will be placed upon the production of activity plans for ensuring resource availability and cash flow control as the project proceeds beyond the feasibility study.

Monitoring and replanning must continue throughout the project to correct any drift that might prevent meeting time or cost targets until the final deliverable has reached the customer" (Saunders F, 2015).

Question 4: Questions on Chapter 6 (23 marks)

4.1 There are a number of different conventions that have been adopted for entering information on network and activity planning diagrams, such as activity-on-arrow networks and the Precedence or activity-on-node network diagrams. Give the naming convention of events for activity-on-arrow networks **as well as** for the nodes in Precedence (or activity-on-node network) diagram as used in Hughes & Cotterel. (8)

ANSWER:



Activity-on-arrow network diagram (4)

Earliest start	Duration	Earliest finish
Activity label, activity description		
Latest start	Float	Latest finish

OR

Early Start	Duration	Early Finish
Task Name		
Late Start	Slack	Late Finish

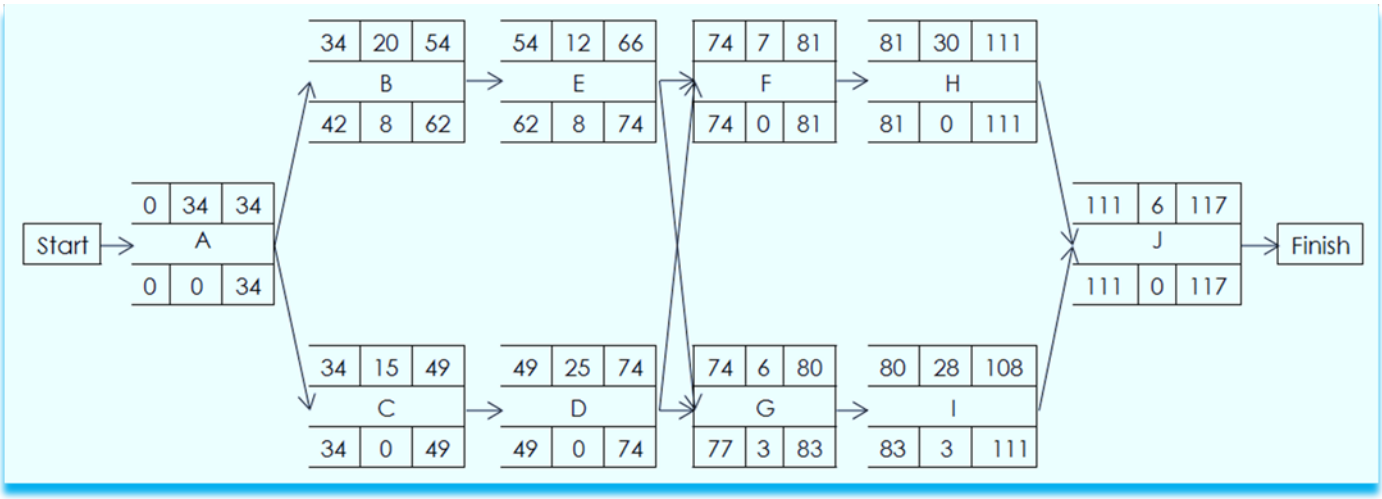
Precedence or activity-on-node network diagram (4)

4.2 Consider the following activities with their precedents and durations.

Activity	Precedents	Estimated duration (days)
A	None	34
B	A	20
C	A	15
D	C	25
E	B	12
F	D, E	7
G	D, E	6
H	F	30
I	G	28
J	I, H	6

Draw a complete Precedence network (Activity-on-node) diagram. Use the naming convention for nodes as used in Hughes & Cotterel, which is based on the British Standard BS 4335. (see textbook page 144) Complete both a forward and backward pass to determine the total duration and critical path. (15)

ANSWER: Precedence network (Activity-on-node) diagram (14)



The critical path is: ACDFHJ = 117 (1)