BUE
The British University In Egypt
الجامعة البريطانية في مصر

15CSCI08I

Software Project Management

Lab (4)

Topics to be covered:

- Size estimations
  - Function points
  - Albrecht II function points
- Exercises

Lab (4)

Estimations

After specifying the detailed aspects of the project whether as tasks or deliverables, we now focus on estimating the expected effort needed for project completion as well as the project size in terms of lines of code.

In order to calculate each, we use specific algorithms that provide a clear and structured method of calculation. The Function point and Albrecht II (modified version of Function points) algorithms are used to estimate software size.

A. Function Points:

This method of calculation considers all system components to belong to one of the following 5 categories:

1. External input types (EI)
2. External output types (EO)
3. External inquiry types (Queries) (EQ)
4. Logical internal file types (ILF)
5. External interface files (UI)

Each of these components may be of low, average or high complexity. The complexity level can be determined based on the tables below:

Table 1: Complexities for External input (EI)

	Number of data types		
Number of record types	1 – 4	5 – 15	16 – *
0 – 1	Low	Low	Average
2	Low	Average	High
3 – *	Average	High	High

Table 2: Complexities for External Output (EO) and External Queries (EQ)

	Number of data types		
Number of record types	1 – 5	6 – 19	20 – *
0 – 1	Low	Low	Average
2 – 3	Low	Average	High
3 – *	Average	High	High

Table 3: Complexities for files (ILF) and User Interfaces (UI)

	Number of data types		
Number of record types	< 20	20 – 50	> 50
0 – 1	Low	Low	Average
2 – 5	Low	Average	High
5 – * (More than 5)	Average	High	High

Steps of the Function Point algorithm:

1. Count the number of components per category
2. Identify the complexity of each component (using tables 1, 2, and 3)
3. For each component, determine the multiplier (weight) that is associated with each category at each complexity level.
4. Add the newly calculated numbers to get the total number of function points

After determining the complexity of each component, use the table below to determine the corresponding multipliers:

Table 4: Multipliers for each of the 5 categories based on the complexity level

Component category	**Complexity of component**		
	Low	Average	High
External Input	3	4	6
External Output	4	5	7
External Queries	3	4	6
Internal Logical Files	7	10	15
User Interface	5	7	10

Note:

✓ The unit for the result is FP (function points)

✓ FP can be converted into lines of code (LOC) if needed

✓ Conversion of FP to LOC depends on the programming language that is to be used

Example:

Brigette at Brightmouth University is responsible for creating a **payroll system**. One of the main components of this system is to analyze the staffing costs for each course. She needs a program that will extract yearly salaries from the payroll file, and hours taught in each course by each member of staff and the details of courses from two files maintained by the time table system. The program will produce a report showing for each course the hours taught by each staff member and the cost of those hours.

Solution:

1. Count # of components per each category “EI, EO, EQ, ILF, EIF”

   External input types ------ none

   External output types ------ the report, that is 1

   External inquiry types ------ none

   Logical internal file types ------ 1 (Payroll file)

   External interface file types ------ 2

   a. Course file (timetable subsystem)
   b. Staff file (timetable subsystem)

2. Identify the complexity of each of the components

   External output types ------ the report, which is 1 data type and is retrieved from 3 record types (Low complexity)

   Logical internal file types ------ 1 data type (salary) to be extracted from 1 record entity (Employee table in payroll file) (Low complexity)

   External interface file types ------ 2 record types and 5 data types (hours taught, course name, course code, course description, cost) (Low complexity)

   a. Course file (timetable subsystem)
   b. Staff file (timetable subsystem)

3. Multiply number of components by the corresponding multiplier

Component category	**Complexity of component**		
	Low	Average	High
External Input	3	4	6
External Output	**4**	5	7
External Queries	3	4	6
Internal Logical Files	**7**	10	15
User Interface	**5**	7	10

So, we get: **7x1 = 7, 2x5 = 10 and 4x1 = 4**

4. Add

By adding both calculated numbers we get:

Size = 10 + 4 + 7
= **21 FP**

After determining the size in function point, we need to calculate the size in terms of lines of code. This depends on which programming language that will be used to develop the system. As such, you would have to search for the multiplier suitable on your own. For example, assuming that the above example will be developed using Java, the multiplier is 21. As a result, we have 60*21 = 1,260 LOC = 1.26 KLOC.

A. Albrecht II Function points:

- This is an improved version of the above algorithm that handles data a bit differently

- This algorithm is concerned with 3 main type categories:
    1. Inputs
    2. Outputs
    3. Entities (Storage)

- The number of elements that belongs to each calculated is calculated

- We then substitute the calculated numbers into the following equation:

$$Fp\ count = N_i * 0.58 + N_e * 1.66 + N_o * 0.26$$

Where

$N_i \rightarrow Number\ of\ input\ elements$

$N_e \rightarrow Number\ of\ entity\ elements$

$N_o \rightarrow Number\ of\ output\ elements$

Example:

A cash receipt transaction in the IOE maintenance accounts subsystem accesses two entity types- Invoice and Cash-Receipt.

The data inputs are:

- ✓ Invoice number
- ✓ Date received
- ✓ Cash received

If an Invoice record is not found for the invoice number then an error message is issued. If the invoice number is found, then a cash receipt record is created. The error message is the only output of the transaction.

Calculate the Albrecht II function points for this system

Solution:

In order to calculate the Albrecht II function points, we will use the following equation:

$$Fp\ count = N_i * 0.58 + N_e * 1.66 + N_o * 0.26$$

We can identify from the above description that there are 3 inputs to consider (listed in bullet points), 2 entity types (Invoice and Cash-Receipt) and one output file which is the error message. Thus, by plugging these values into the equation we get:

$$Fp\ count = 3 * 0.58 + 2 * 1.66 + 1 * 0.26 = \mathbf{5.32\ FP}$$

Exercises:

1.

A logical internal file might contain data about purchase orders. These purchase orders might be organized into two separate record types: the main PURCHASE-ORDER details, namely purchase order number, supplier reference and purchase order date, and then details for each PURCHASE-ORDER-ITEM specified in the order, namely the product code, the unit price and the number ordered.

Using the above information, calculate the size of the software using the function point I method

2.

Detailed course information describes the course code, course name, number of credits, the list of periods for available course implementations (i.e. same course can have several implementations during the year; course implementations are identified by a number and year), course events (course event has a name, place and time), course event enrolment statuses (either: closed, open for enrolment, already enrolled), longer description of the course, course prerequisites and contact information. Student is able to enroll to a course event from this view. Calculate the size using the Albrecht II method.