

## Cost Estimation Activity

1. Working in with a group of 3 or 4 students estimate the time and personnel costs required to build the ATM system based on the use cases provided.
2. Compute a person month estimate for the time required to build the ATM system using the proxy technique using data from the table below.

Type	LOC per item				
	VS	S	M	L	VL
Calculation	2.34	5.13	11.25	24.66	54.04
Data	2.60	4.79	8.84	16.31	30.09
I/O	9.01	12.06	16.15	21.62	28.93
Logic	7.55	10.98	15.98	23.25	33.83
Set-up	3.88	5.04	6.56	8.53	11.09
Text	3.75	8.00	17.07	36.41	77.66

You will need to reach a group consensus as classify the modules needed to implement each use case.

Do you believe that language choice is important to completing the estimate?

Do these numbers seem consistent with your experiences?

## Cost Estimation Activity

3. Compute the Function Points needed to build the ATM system from the use cases provided. Be sure to separate each module based on their proposed implementation language.

Parameter	Simple	+	Average	+	Complex	=	F <sub>i</sub>
Distinct input items	3 ( )	+	4 ( )	+	6 ( )	=	?
Output screens/reports	4 ( )	+	5 ( )	+	7 ( )	=	?
Types of user queries	3 ( )	+	4 ( )	+	6 ( )	=	?
Number of files	7 ( )	+	10 ( )	+	15 ( )	=	?
External interface	5 ( )	+	7 ( )	+	10 ( )	=	?
					<b>Total</b>	=	<b>?</b>

$$\text{F.P.'s} = T * (0.65 + 0.01 * Q)$$

T = unadjusted table total

Q = score from questionnaire  
(14 items with values 0 to 5)

4. Try to reconcile your function point estimates with another group. If when the two estimates are within 5-10% of each other you may consider them reconciled.

# Cost Estimation Activity

## Function Point Complexity Weighting Factors

(0 = No influence, 1 = Incidental, 2 = Moderate, 3 = Average, 4 = Significant, 5 = Essential):

Question	0	1	2	3	4	5
1. Does the system require reliable backup and recovery?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2. Are data communications required?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3. Are there distributed processing functions?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4. Is performance critical?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. Will the system run in an existing, heavily utilized operational environment?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6. Does the system require on-line data entry?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7. Does the on-line data entry require the input transaction to be built over multiple screens or operations?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8. Are the master file updated on-line?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
9. Are the inputs, outputs, files, or inquiries complex?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10. Is the internal processing complex?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11. In the code designed to be reusable?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
12. Are conversion and installation included in the design?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
13. Is the system designed for multiple installations in different organizations?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
14. Is the application designed to facilitate change and ease of use by the user?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Total						

## Cost Estimation Activity

5. Convert your function point estimate to a line of code (LOC) for the ATM system using the table below.

Language	QSM SLOC/FP Data			
	Avg	Median	Low	High
ABAP (SAP) *	28	18	16	60
ASP*	51	54	15	69
Assembler *	119	98	25	320
Brio +	14	14	13	16
C *	97	99	39	333
C++ *	50	53	25	80
C# *	54	59	29	70
COBOL *	61	55	23	297
Cool:Gen/IEF *	32	24	10	82
Datastage	71	65	31	157
Excel *	209	191	131	315
Focus *	43	45	45	45
FoxPro	36	35	34	38
HTML *	34	40	14	48
J2EE *	46	49	15	67
Java *	53	53	14	134
JavaScript *	47	53	31	63
JCL *	62	48	25	221
LINC II	29	30	22	38
Lotus Notes *	23	21	19	40
Natural *	40	34	34	53
.NET *	57	60	53	60
Oracle *	37	40	17	60
PACBASE *	35	32	22	60
Perl *	24	15	15	60
PL/1 *	64	80	16	80
PL/SQL *	37	35	13	60
Powerbuilder *	26	28	7	40
REXX *	77	80	50	80
Sabretalk *	70	66	45	109
SAS *	38	37	22	55
Siebel *	59	60	51	60
SLOGAN *	75	75	74	75
SQL *	21	21	13	37
VB.NET *	52	60	26	60
Visual Basic *	42	44	20	60

## Cost Estimation Activity

6. Use Basic COCOMO to compute person month estimate for the ATM system.

- Effort in person months =  $a S^b * m(x)$
- S measured in KLOC = LOC/1000

	BASIC		INTERMEDIATE	
MODE	a	b	a	b
Organic	2.4	1.05	3.2	1.05
Semidetached	3.0	1.12	3.0	1.12
Embedded	3.6	1.20	2.8	1.20

How will you convert the person month estimates to dollars so the client will believe the numbers?

7. Use Intermediate COCOMO to compute a person month estimate for the ATM system. Recall  $m(x) = \pi m(x_i)$

Cost adjustments for computing the EAF (Effort Adjustment Factor)

	v. low	low	nominal	high	v. high	ex. high
<b>product attributes</b>						
required software reliability	0.75	0.88	1.00	1.15	1.40	
database size		0.94	1.00	1.08	1.16	
product complexity	0.70	0.85	1.00	1.15	1.30	1.65
<b>computer attributes</b>						
execution time constraints			1.00	1.11	1.30	1.66
main storage constraints			1.00	1.06	1.21	1.56
virtual machine volatility	0.87	1.00	1.15	1.30		
computer turnaround time		0.87	1.00	1.07	1.15	
<b>personnel attributes</b>						
analyst capability	1.46	1.19	1.00	0.86	0.71	
applications experience	1.29	1.13	1.00	0.91	0.82	
programmer capability	1.42	1.17	1.00	0.86	0.70	
virtual machine experience	1.21	1.10	1.00	0.90		
programming language experience	1.14	1.07	1.00	0.95		
<b>project attributes</b>						
use of modern programming practices	1.24	1.10	1.00	0.91	0.82	
use of software tools	1.24	1.10	1.00	0.91	0.83	
required development schedule	1.23	1.08	1.00	1.04	1.10	