Full Paper

Introduction

Assessment is an important component in teaching where a student is assessed for his knowledge and skills in the concepts that are taught to him. Improved assessment techniques improve learning (Kesuma, 2013). Assessment strategies include written examinations, viva voce, seminars, assignments, projects, etc. Marking techniques in written examinations primarily focus on the correctness of results often using numerical values. A score of 0 is offered for wrong answers, and 1 (full mark) for correct answers. (We refer to this as 0/1 marking.) Partial marking is more exhaustive and fairer than 0/1 marking since an incorrect answer can be partially correct. However, partial marking strategies are not based on formal techniques, and may not be uniformly consistent. Existing automated tools at best provide passive support to assessment and they do not explicitly model the concepts students learn in their curriculum. There is thus a need for providing a methodology for a more accurate method of assessing a student's answer in written examinations.

Goal

Our goal in this research is to provide a systematic method for assessing student answers using marking techniques. We focus on the answers presented in written examinations, though our methodology can be applied to other types of assessments as well. Our approach involves using a task-data flow graph to guide marking.

Background

Formative assessment techniques involve quizzes, examinations, short essays, direct observations, team work, etc (Formative, 2014). Markulis et al. (2008) present several guidelines for implementing oral examination methodologies. However, typically a viva voce is "reactive" in nature (unless it is well planned in advance) in the sense that questions are determined by the response of the student to earlier questions, and thus they are not based on concept map navigation. An elaborate strategy for conducting viva voce by systematically organizing questions based on domain ontologies and dialog techniques is proposed in (Parameswaran et al. 2014). It is well recognized that assessment techniques can directly motivate students for learning concepts (Ramsden, 2007). Further, assessment process not only measures student understanding a concept, but also it is a great motivator (Ooi & Buskes, 2011). Multiple choice question structure is known to favour improved performance in an assessment process (Klimovskia & Cricentia, 2013). Sanz-Lobera, et al. (2011) propose a methodology for automatic generation of questions which can be used as self-assessment questions by the student and it is based on the utilization of parametric questions, formulated as multiple choice questions and generated and supported by the utilization of common programs of data sheets and word processors.

Hartia (2011) describes an initiative for skill development for students. In this, a blended approach is adopted that combines a training based practicum module along with an on campus professional development program. In this effort, assessment consists of a combination of reflective learning and VET based competencies. Bhave, et al. (2011) investigate the effectiveness of summative and formative assessment of the group presentations from multiple perspectives by different assessors, viz., technical perspective by a technical expert, presentation perspective by a generic skills expert, and students perspective using tools of peer assessment.

None of the assessment techniques mentioned above, however, is based on any well-defined methodology and they suffer from lack of objectivity and consistency. Marking can be either top down or bottom up. A tutor can use either of the techniques but in practice, they do not yield the same results. Top down marking verifies the method first and then the answer (data) at every step. Bottom up verifies the result and then the method. In bottom up marking, if the answer is correct, the method is often assumed to be correct; else, the method is assessed. In our methodology, we use the top down technique since we believe that for partial marking top down methodology is likely to produce more accurate results.

Methodology

In order to assess a student's answer S, we compare it with a sample solution E provided by the instructor. Comparing S and E will not be easy in general since both S and E can in the form of unstructured data. Thus, in our methodology, to make the task easy, we begin by representing the sample answer E using a task-data flow graph and then use it to guide marking the student answer systematically.

Task-data flow graph

Definition

A task-data flow (TDF) graph is a graph where each node denotes a (sub) task and each edge denotes the flow of data from one node to another.

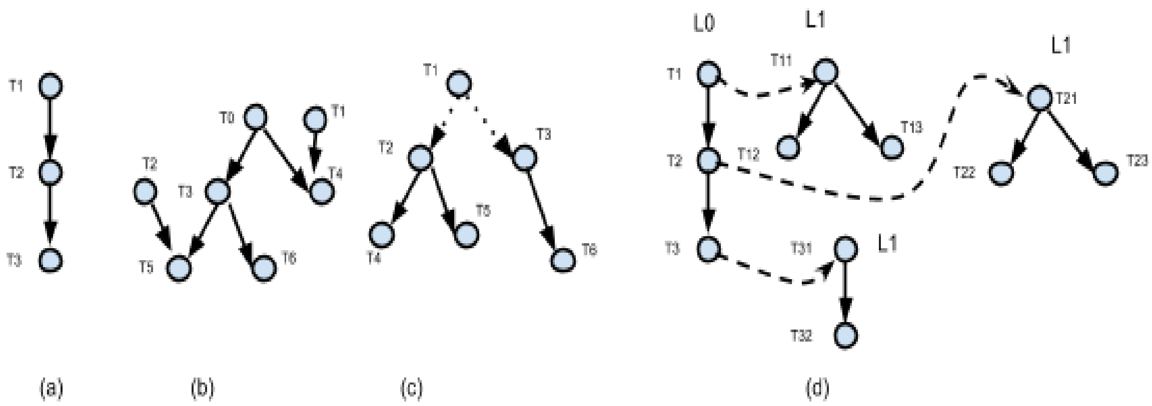

Figure 1. Task-data flow graphs: (a) simple sequence of subtasks; (b) graph with parallel subtasks; (c) graph with parallel subtasks and options shown by dotted edges; (d) graph showing sub task expansion into levels L0 and L1.

Figure 1 shows four simple TDF's. Figure 1(a), shows a TDF that has three subtasks T1, T2, T3 occurring in a sequence. Figure 1(b) shows subtasks organized in sequence and in parallel.

Thus, T0 and T1 occur in parallel after which T3 and T4 occur in parallel. In the meantime, T2 occurs independently. After T2 and T3 are finished, T5 occurs. Figure 1(c) shows a TDF where options occur in task structure. Thus, after performing the subtask T1, either T2 can be performed or T3 can be performed. Figure 1(d) shows a TDF at two levels. In the first level L0, the sub tasks shown are T1, T2, and T3. In the second level L1, each sub task is expanded with more detailed sub tasks. Thus for example, the sub task T1 is achieved by executing lower level subtasks T11, T12, and T13 organized as a tree as shown. There may be more than two levels in general. When a node that has no expansion is called a primitive node and it represents a primitive task. A primitive task may be a simple task that can be executed without requiring further elaboration or a set of rules each of which can be executed without further expansion. Thus, nodes T31 and T21 are primitive nodes. A rule specifies a conditional sub task which is performed when a specified condition is satisfied. Typically, when the details of a subtask are not required for marking, the sub task can be represented by primitive nodes. However, when details of the sub task are essential for an assessment, the node is expanded to provide the details of the sub task. Further, it must be noted that in a TDF-graph, while the nodes denote the sub task to be performed and the arcs denote the data flow between the nodes, the structure of the graph itself depicts the step-by-step method of the overall task that the graph denotes.

Building task-data flow graphs

We consider three types of answers.

Type I The answer is in the form of a sequence of subtasks to a given problem applying a predefined technique, such as finding the current in a particular branch in a given electronic circuit.

Type II The answer is in the form of a description of a sequence of events, such as the ones that occur in a process.

Type III The answer is in the form of a description of logical flow, such as in the description of a design of a machine.

Assessment Process

An assessment process must be consistent. Consistency refers to following uniform policies across all answers being marked. We perform the assessment by first obtaining the task-data flow graph G from the sample solution provided by the instructor. Assessment proceeds in a top down fashion starting from the highest level. It involves two phases:

Phase 1 In this, we assess the method of solution by matching the structure of the graph with the structure of the solution. The match proceeds from the first node of G. The match may be complete or partial when only the initial part of the graph matches successfully. We offer marks for the part that matched successfully.

Phase 2 In this, we assess the task performed at each node. If the node is primitive, then it is straightforward to assess it and offer marks for it. If it is not, then assessment is carried over to the part of the graph that is shown as an expansion of the node. We assess this part of the graph once again in two phases. The process continues until all nodes are assessed and offered marks.

Finally, using the aggregate at all levels, we compute the overall score using the weights chosen at each node. To account for the relevance of concepts at each node, the weights of the nodes are chosen carefully: (a) the concepts at the node that are believed to be central to learning are assigned higher weights. Note that such concepts may occur at any node in the graph; (b) concepts that are considered prerequisite to the course or are part of assumed knowledge are given lower weights. We illustrate our assessment process using an example below.

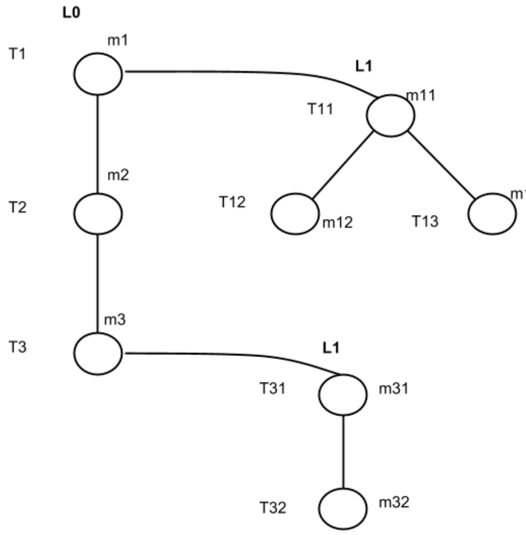

Figure 2. TDF graph: T1: calculate power P1 = I3*I3*R3; T2: calculate total power P2 = P1 + I1*I1*R1 + I2*I2*R2; T11: calculate I3=I1+I2; T12: calculate I1=V1/R1 T13: calculate I2=V2/R2. L0 and L1 denote levels L0 and L1. mi denotes partial mark for node Ti.

We assume, for the sake of simplicity, the TDF-graph shown above in Figure 2 which has its subtasks at two levels. Marking begins by first checking the method of the solution and this is achieved by checking if the method in the student solution matches correctly with the structure of the graph at level L0. We thus check if the student has performed two subtasks T1 followed by T2 (without worrying about the details shown in level L1). If he has, then the method of his solution is correct, and we offer marks for the correct method, and we proceed then to mark each node. If the structure does not match correctly, we identify the largest initial segment that matches correctly and we offer partial marks for the initial correct segment and proceed to mark each node in this segment. Thus, for example, if T1 is correct but not T2, then the initial segment that is correct is T1. We offer marks for the partial method consisting of T1, and proceed to mark the node T1. Note that we do not mark the node T2.

Marking an individual node involves marking its expansion (if any) and then marking the node itself. If a node Ti is primitive, then it is marked using 0/1 logic. That is, if the task at a primitive node Ti is done correctly, then a full mark assigned for that node is offered. If the task is done erroneously, then the mark offered is 0. However, if the node Ti is not primitive, marking Ti involves marking its expansion and then marking Ti. Thus, in Figure 2 above, in order to mark the node T1, we mark its expansion shown as a tree at level L1. We mark this tree following the method we used for level L0 (as above), and then we mark T1.

Wrong data and Invalid data

We observe that in a TDF-graph the data generated at a node flows to its neighbouring nodes. Sometimes minor mistakes in calculations leads to *wrong results*. Wrong results that satisfy certain constraints will be called as wrong data whereas wrong results that are unconstrained are categorized as Invalid data. Thus, for example, the voltage across a resistor in an amplifier (with a supply voltage as 12V) computed wrongly as 10V (while the correct value was 2V) may be viewed as wrong data, but when computed as 100V will be taken as invalid data. (The constraint here is that the voltage computed should be less than 12V.)

Wrong data flow We propagate wrong data to the remaining part of the structure to continue the assessment subject to certain policies. Wrong data is permitted to flow to the remaining part of the structure only when the remaining part of the structure has already been assessed to be correct. In this case, the wrong data that propagates may generate a "wrong answer" but still will be valid with respect to the structure. However, when the remaining part of the structure has already been assessed to be incorrect, the assessment is aborted. We assume that when wrong data flows through an incorrect, invalid data will be generated. Thus, for example, if the method at L0 and at L1 are both correct, but due to a minor error in the calculations at level L1 the answer computed is wrong, then we mark node T2 for the wrong data and check if T2 has been performed "correctly" for the wrong data. If it has been performed "correctly", we offer full mark m2 for the node T2.

Applications to Electronic Circuit Analysis

We applied our methodology for marking student answers in a course on fundamentals of electronic circuits. A typical question presented to the students had two components: a) testing the knowledge of the concept taught from the textbook (also called the domain concepts); and b) solving problems using the concepts taught. Certain questions emphasised the first and the others the second.

Task-data flow graph for Type I answers

In Type I answers, the sample answers are in the form of sequences of steps that were assessed for their correctness. An example is given below. Figure 3 shows the TDF-graph for this example.

$$R_{eq} = \frac{1}{1/R_2 + 1/R_3 + 1/R_4} = 9.231\,\Omega \qquad i_1 = \frac{20\text{ V}}{R_1 + R_{eq}} = \frac{20}{10 + 9.231} = 1.04\ A$$

$$v_{eq} = R_{eq} i_1 = 9.600\text{ V} \qquad i_2 = v_{eq}/R_2 = 0.480\ A \qquad i_3 = v_{eq}/R_3 = 0.320\ A$$

$$i_4 = v_{eq}/R_4 = 0.240\ A$$

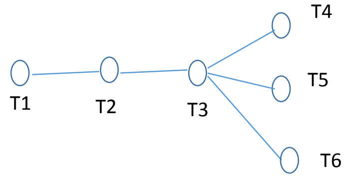

Figure 3. TDF-graph. T1: calculate Req; T2: calculate i1; T3: calculate Veq; T4: calculate i2; T5: calculate i3; T6: calculate i4.

From Figure 3, we see that the sub tasks T1, T2 and T3 are done one after another while T4, T5 and T6 are done parallel (that is, in any order).

Task-data flow graph for Type II answers

In this type, the answer is in the form of a description of events occurring in a process. An example is shown below.

The current ***passes*** *from the emitter to the collector through the base. Changes in the voltage connected to the base* ***modify*** *the flow of the current. This* ***changes*** *the number of electrons in the base. This* ***affects*** *the current reaching the collector. The current in the collector* ***affects*** *the voltage measured at the collector.*

In the above description there were five events caused by five actions (shown in bold letters). The TDF-graph in Figure 4 shows this sequence of events.

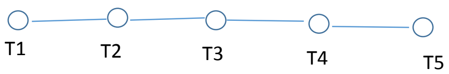

Figure 4. TDF-graph. T1: The current **passes** from the emitter; T2: **modify** flow of current; T3: **changes** the number of electrons ; T4: **affects** current; T5: **affects** the voltage measured.

Task-data flow graph for Type III answers

In this type, the answer is in the form of a description of assertions in a logical sequence as shown in the example below.

Under open-circuit conditions, 5 A circulates clockwise through the current source and the 10-Ω resistance. The voltage across the 10-Ω resistance is 50 V. No current flows through the 40-Ω resistance so the open circuit voltage is V_t*=50V.*

A TDF-graph is drawn by representing each assertion by a node and the logical dependency between assertions by the edge of the graph as shown in Figure 5.

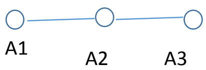

Figure 5. TDF-graph for the above answer. A1: Under open-circuit conditions, 5 A circulates clockwise through the current source and the 10-Ω resistance. A2: . The voltage across the 10-Ω resistance is 50 V. A3: No current flows through the 40-Ω resistance so the open circuit voltage is V_t=50V.

TDF-graph application to Assessing Electronic Circuit

We applied our technique to assess answers provided by students in a written examination. TDF-graphs were derived for each case from the sample answers provided by the instructor and the graphs were then used to mark the student answers.

Challenges in extracting TDF-graphs

Since the TDF-graphs were extracted from the sample answers provided by the instructor, the task of extraction was straightforward particularly when assistance was available from the instructor. The total marks for the question was distributed across the nodes in the graph. It was decided that the fraction of the mark assigned to a node would depend on the degree of relevance of the concept associated with the node. Though at times this task was challenging, it is important to carry out this step carefully to ensure fairness particularly in marking incorrect answers.

Challenges in marking student answers

While correct answers were easy to mark most of the time, marking partially correct answers posed several challenges:

- Locating the steps in the answer corresponding to a task node in the graph.
- Absence of intermediate steps corresponding to a task node. (This happens when the student did not clearly show the steps, or did not know the step.)
- Incorrect methods and thus unsuccessful match.
- Wrong data propagation. This was at times time consuming. Most of the marking time was spent on wrong data propagation. However, this time was spent more fruitfully as it improved fairness in marking.

When intermediate steps are not shown, we have two cases to consider: (1) The final answer is correct. In this case, we assume that all missing intermediate steps are also correct; (2) The final answer is wrong. In this case, we assume that missing intermediate steps are all wrong; and (c) The final answer is partially correct. In this case, except the nodes that are responsible for the correct answer, all other nodes were assumed to be incorrect.

Applying our methodology to a sample of about 100 papers, we observed that about 90% of them had to be partially marked. Marking involved both checking the task structure and the data flow. An error node is a node where a mistake in the student answer was noticed. Error nodes may occur at any depth d of the graph. The occurrence of an error node triggers partial marking since error nodes generate wrong data. According to our wrong data policy described above, flow of wrong data was permitted in marking only when the task structure was valid, since permitting data flow across incorrect task structure produced semantically incorrect results. While marking the 100 answer books using our TDF-graph method and comparing the marks with the marks obtained from a (human) tutor marking, we were able to make the following observations.

Table 1. TDF-graph based marking Versus Tutor marking

TDF-graph based marking	Tutor marking
Systematic and consistent.	Ad hoc and often inconsistent.
Worked well both for completely correct answers and partially correct answers.	Worked well only for completely correct answers, and not so well for partially correct answers.
Fair always.	Mostly fair only for correct answers.

Marking was exhaustive by going through all steps.	Often non exhaustive. Once an error was noticed, remaining part of the answer was either ignored or judged with personal bias.
Fair *wrong data* policy.	No evidence for any *wrong data* policy was noticed. Marking was not strategy based, but rather on personal bias.
Marking was always top down irrespective of whether the answer was completely correct or partially correct.	Marking was often bottom up. Partial marking was attempted only when incorrect answers were encountered. Marking was terminated when too many incorrect answers were encountered.
The severity of the penalty imposed for incorrect answers depended on the severity of conceptual violations noticed.	Severity of penalty appeared to have been prompted by personal bias and judgment.
Consistency was independent of at what depth an error occurred and how many nodes were affected by the error node.	No calculations were done about how many nodes were affected. But, larger depth of an error node usually meant low penalty.

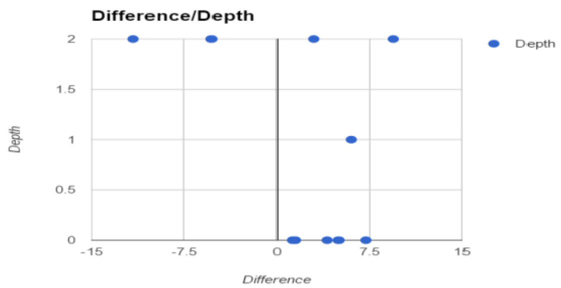

Figure 6. Depth of the node as a function of difference.

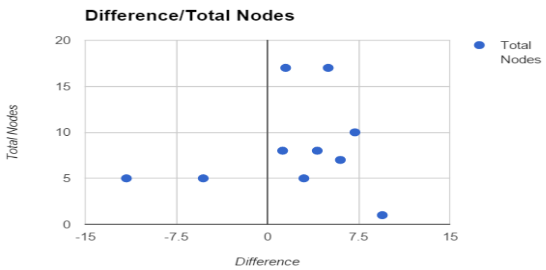

Figure 7. Total number of nodes affected as a function of difference

In partial marking, the tutor sometimes ignores certain details of the answer whereas in our approach it was possible for us to identify the corresponding nodes in the TDF-graph. We thus see ignoring a set of nodes and substructures can contribute to the difference between TDF-graph based marking and Tutor marking. If mi and ni are the marks offered by our TDF-based marking and the tutor marking, respectively, then difference is defined as Di = mi – ni. Figure 6 shows depth di for an error node Ti as a function of Di. We notice that most of the depths have clustered around a difference of 0 to 7.5.

The occurrence of an error at a node propagates to other nodes resulting in wrong answers at those nodes. Figures 7 shows that most of the affected nodes occur in the difference range 0 to 7.5.

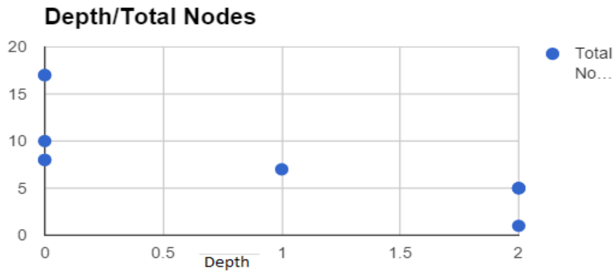

Figure 8. Total number of nodes affected as a function of depth of error node.

Figure 8 shows that the total number of nodes decreases as the depth increases as one would expect. Thus, if errors occur at a greater depths, marking strategy is likely to be more consistent. Errors in the student answers were noticed more at problem solving steps rather than at the concept level. Validating our results with experts showed that our evaluation methodology was fairer and exhaustive. It was also possible for us to identify and target chosen concepts and task structure.

Discussion and Conclusion

Problem solving skills play a central role in education and assessing them often means assessing the student skills to perform individual tasks. In an examination based assessment, most of the students solve at least a few problems incorrectly. In fact, most of the time, most of the answers are only partially correct. Marking solutions that are only partially correct is more challenging than marking solutions that are completely correct. Incorrect solutions can be incorrect for different reasons, and an assessor has no way of predicting them. Thus, partial marking is often not well formalized and consequently is not transparent to students. Our TDF-graph based method is not only more objective in nature but also makes partial marking more transparent to students and tutors.

We argue the need for concrete model for solutions before marking actually begins. The difficulty in obtaining such a model only points to the fact that marking will be difficult otherwise. This also means, the instructor must focus on those materials that are objectively assessable. At some point, from an instructor's point of view, what a student has learned is only what the instructor's assessment shows.

Our methodology is mechanizable since it is possible to obtain a TDF-graph from a given solution in a fairly straightforward manner. Using the TDF-graph, an implemented system can ask questions, and the tutor examining the student answers can provide replies. Thus, the tutors with lower skills will often be adequate for marking complex answers. Without the system, marking a student answer ideally requires the skill of the instructor. When the TDF-graph is more detailed, the tutor’s skill required correspondingly reduces. The problem of marking can be partially eased by building an interface where a student is guided to present the solution in a systematic manner where the system evaluates the partial structure interactively as the student presents the solution. This will considerably solve the problem of identifying the parts of the written answers for a given task node in the TDF-graph. The TDF-graph can also be used to prepare the slides for teaching. Well before teaching a course begins, a teaching strategy should clearly identify the domain concepts and the degree of their relative importance within the scope of that course in the curriculum, and the set of problem solving strategies that will be discussed and practised in the course. Each question in the assessment scheme must explicitly target a chosen set of concepts and the associated problem solving skills. This will then help us obtain the task-data flow graph systematically with relative ease and will thus help in an easy implementation and effective partial evaluation to the problem of student assessment.

References

Bhave, P., Topkar, V., Bhonsle, M. (2014). *A Multi-Perspective Approach to Technical Presentation Assessment*, Paper presented at the AAEE2014 Conference, Wellington, New Zealand.

Formative (Informal) Assessment Strategies (2014). Retrieved October 19, 2015 from http://www.bostonpublicschools.org/cms/lib07/MA01906464/Centricity/Domain/44/Formative_Informal.pdf (viewed 2014)

Dave Hartea; Mark Symesa. (2011) *Alternative Approaches to Skill and Assessment Development in a Co-operative Education Program*, Paper presented at the AAEE2014 Conference, Wellington, New Zealand.

Klimovskia, D., and A. Cricentia. (2013) *Does the multiple choice question structure in examinations have an effect on student performance*, Paper presented at the AAEE2013 Conference, Gold Coast, Queensland, Australia.

Markulis,P.M., and Strang,D.R.(2008) Viva Voce: Oral Exams as a Teaching and Learning Experience, *Developments in Business Simulation and Experiential Learning*, 35,118-127.

Ooi, A , & Buskes, G. . (2011*) A survey of strategies for feedback and assessment in engineering subject: Discussions and examples*, Paper presented at the AAEE2011 Conference, Western Australia, Australia.

Parameswaran, N., Gorthi, R.P., and Tiwari,A. (2104) *Viva voce for Student Assessment and Learning*, Paper presented at the AAEE2014 Conference, Wellington, New Zealand.

Ramsden, P. (2007) Learning to Teach in Higher Education, London; New York.
Sanz-Lobera, A., Roig,A., and Requena,I. (2011) *Methodology for automated generation of multiple choice questions in self-assessment*, Paper presented at Research in Engineering Education Symposium 2011 Madrid, 4th - 7th October 2011.

Kesuma, G.C. (2013) Relationship Between Students' Learning And Assessment Towards Evaluation System With Students' Learning Achievement Of Tarbiyah Faculty IAIN Raden Intan Lampung, INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 2, ISSUE 6, JUNE 2013.