EAPoster: NSF/TCPP Early Adopter Experience at Beijing Normal University Zhuhai

Wei Lu[1], Wen Yu[1], Wei Sun[1], Hongyu Xiao[1] and Feng Gu[2]
[1]Department of Software Engineering, Beijing Normal University Zhuhai, Zhuhai, Guangdong, China 519087.
{luwei, wenyu, asunwei, hyxiao}@bnuz.edu.cn
[2]Department of Computer Science, College of Staten Island, Staten Island, NY, USA 10314.
Feng.Gu@csi.cuny.edu

ABSTRACT
The software engineering department at Beijing Normal University Zhuhai is integrating the NSF/IEEE-TCPP Curriculum Initiative on Parallel and Distributed Computing (PDC) into its department-wide undergraduate multi-semester core courses. The related courses include SE101 (Introduction to Computer Programming), SE102 (Data Structures), SE303 (Algorithms) and SE304 (Operating Systems). As the first effort, the inclusion of the PDC modules was gradual since PDC ideas were not systemically introduced before. Preliminary training of instructors was performed and PDC topics to be covered were identified. Part of our works were evaluated. Further efforts, results and evaluation will be shared on CDER website.

1. BACKGROUND
The Department of Software Engineering in Beijing Normal University Zhuhai (BNUZ) currently has an enrollment of 508 students and 21 faculty members in the department. The program offers Bachelor's Degree of Engineering in three specialization areas, including Commercial Software Development, Mobile Software Development, and Software Testing.

In order to get the current level of the PDC knowledge of students, we performed 2 surveys. The target students were freshmen and juniors. They both participated the survey at the beginning of the fall semester in 2015. PDC topics were not systematically introduced to the students then.

Table 1 shows the result of survey to the freshmen. In this survey, students were asked whether they heard about the given 8 PDC related terms. The information gives us hits for developing teaching strategies for SE101 and SE102, such as paying special attention on Speedup, OpenMP and MPI terms.

Table 2 shows the result of survey to the juniors. The related profession courses they had taken including Introduction to Computer Programming, Data Structures, Computer Network, Programing Design in Java I and II, Database and Software Engineering. Students were asked to rate their knowledge levels on 8 PDC terms, which the same as the terms were asked to the freshmen except Hadoop and Go Programming Language. The term PDC was removed because it is too general and the term Multithread was remove because they learned it in the course Programming Design in Java II. Different from the "yes or no" format used in the survey to the freshmen, we designed 3 rating levels to get more detailed information. Level 0 represents not heard. Level 1 represents simply knowing, and level 2 represents knowing well. As we expected, no students knew the terms Speedup, OpenMP, MPI, Go programming language and just a few students understand the internal structure of Hadoop. Hence, we should emphasize above 5 topics in SE303 and SE304.

Table 1. PDC terms survey to freshmen

Terms	Percentage of knowing the topic
PDC	60
Multicore	100
Shared Memory	55
Distributed Memory	33
Multithread	25
Speedup	0
OpenMP	0
MPI	0

Table 2. PDC terms survey to juniors

Terms	Percentage of Positive Answers		
	0	1	2
Hadoop	19	76	5
Multicore	0	88	12
Shared Memory	0	55	45
Distributed Memory	0	80	20
Go Programming Language	8	92	0
Speedup	95	5	0
OpenMP	0	100	0
MPI	0	100	0

2. EARLY ADOPTING COURSES

Four courses are planned to be integrated with the NSF/IEEE-TCPP Curriculum Initiative on PDC. They are:

SE101 Introduction to Computer Programming: This is the first programming course that introduces basic programming knowledge and techniques. It also presents an overview picture of a computer system. Some of topics covered in this course include Multicore, Shared Memory, Distributed Memory, Time and Space/Memory. Instructor will show a project including OpenMP and let them fell how PDC techniques can improve the efficiency of software.

SE102 Data Structures: This is the second low level core course in our course system, following SE101. Common data structures, such as linear lists, stacks, queues, trees and graphs, and associated operations are introduced. This course continues developing some fundamental parallelism concepts that are introduced in SE101. In addition, some topics related to efficiency are introduced, including Speedup, Divide & Conquer and Recursion.

SE303 Algorithms: This is a high level core course introducing advanced algorithms after SE102. Algorithm analysis will be covered. In terms of PDC, this course also focuses on efficiency, but the analysis is in depth. In this course, PDC topics include Shared Memory, Distributed Memory, and Synchronization, Load balancing, Speedup and Dependencies. PDC knowledge is expected to be exercised in the course project.

SE304 Operating Systems: This is a high level core course that introduces process and thread management, memory management, file system management and I/O management. The PDC topics intended to cover in this course include Multicore, Shared memory, Distributed memory, Synchronization, Deadlocks and Load balancing. This course plans to use UNIX as an example to introduce all kinds of parallelization reform. A course project that designs and implements micro kernel with some parallel features will be designed. In addition, Hadoop and Go will be introduced and exercised.

3. ACTIVITIES

Our efforts on integrating the NSF/IEEE-TCPP Curriculum Initiative on PDC started the beginning of 2015. The details of our activities are show in Table 3. The evaluation of the course SE101 was performed by including the PDC topics in the final exam. The result shows that over 90% of the students understood the basic PDC concepts. The evaluations of the course SE303 and SE304 were done by identifying the PDC related contents in their course projects. Most students mentioned PDC related techniques, such as Multicore, Shared Memory, Distributed Memory, Speedup and Multithread in their projects. What impressed us is that 30% of them chose to develop distributed systems. Selected topics are several personal blog systems (using Go), an IAAS system (using Hadoop), a web crawler system (using Hadoop), a recommender system (using Spark). Some students also participated in industrial projects, such as customer flow analysis for YY Inc. (using Hadoop). In the future, special concern will be given to revising PDC topics, teaching materials, teaching strategies and evaluating strategies. In addition, we will extend our efforts to other suitable courses.

Table 3. Plan of integrating the NSF/IEEE-TCPP Curriculum Initiative on PDC to Courses

Date	Activities	Operated By	Status
Jan. 2015	Preliminary training for instructors in BNUZ	Feng Gu	Finished
Feb. 2015	Identifying PDC topics covered in each course	Wei Lu, Wen Yu, Wei Sun, Hongyu Xiao, Feng Gu	Finished
Feb. 2015 – Aug. 2015	Revising course materials to cover desired PDC topics	Wei Lu, Wen Yu, Wei Sun, Hongyu Xiao	Finished
Aug. 2015	Starting to develop teaching and evaluating strategies	Wei Lu, Wen Yu, Wei Sun, Hongyu Xiao	Finished
Sept. 2015	Survey on Freshmen and Juniors	Hongyu Xiao and Wei sun, respectively	Finished
Sept. 2015 – Jan. 2016	Implementing PDC modules in SE101, SE303 and SE304	Hongyu Xiao, Wen Yu and Wei Sun, respectively	Finished
Jan. 2016	Evaluating efforts put on SE101, SE303 and SE304	Wei Lu, Wen Yu, Wei Sun, Hongyu Xiao	Finished
Mar. 2016 – June 2016	Implementing PDC modules in SE102	Wei Lu	On Going
July 2016	Evaluating efforts on SE102	Wei Lu, Wen Yu, Wei Sun, Hongyu Xiao	Planned
July 2016 – Aug. 2016	Summarizing experience; adjusting PDC topics; revising teaching and evaluating strategies; upload course materials to CDER resource site	Wei Lu, Wen Yu, Wei Sun, Hongyu Xiao	Planned