



计算机编程协同学习环境中最优团队规模的实证研究

姜赛达

(商丘职业技术学院 计算机系,河南 商丘 476000)

摘要:协同编程一直被广泛认为是计算机编程的最佳方式。当今,协作已被证明可以在编程过程中产生更好的效果。但是,协作所需的最佳团队规模尚需得到更深入的研究。旨在向参与研究的学生灌输和熟悉软件项目的团队合作精神,并根据研究确定在编程/学习的现实环境中需要的有效(最佳)团队规模。组织并进行了两个不同的实验,制定了确定最佳团队规模的参数。不同性别的自愿参与者被随机分为5个不同大小的平行团队,在第一个实验中从1到5不等。每个团队规模被复制六次。第二个实验涉及不同大小的同性别成分(男性或女性)团队。通过问题分析和编码的时间以及每个团队规模记录的编译时的错误。最终分析了团队的有效性。该研究表明,协作对新的计算机编程学习者非常有益。当学习在他人的陪伴下完成时,他们很容易掌握编程概念。该研究还表明,计算机编程协作学习中可采用的最佳团队规模为四。

关键词:最佳团队规模;协作学习;协同编程;计算机编程

中图分类号: TP312.2

文献标识码: A

文章编号: 1671-931X (2019) 03-0115-05

一、背景分析

众所周知,计算机编程是一项很难完成的复杂任务。计算机相关专业学生遇到的第一个也是最具挑战性的任务是如何理解计算机编程。学生面临的问题归因于高度失败以及课程中出现的困难,这些困难直接依赖于编程能力,逻辑推理能力和解决问题的能力。由于每个班级的学生人数众多,教师在编程实验室教学中不能完全对每个学生进行有效的指导。现有研究表明通过协作编程可有效改善这些问题,例如更高的程序质量,更快的编程速度,更少的错误和更大的乐趣^[1]。协同编程的历史可以追溯到穿孔卡,它成为一种可行的软件编程方法在20世纪

90年代早期被应用。同时它已被广泛接受为传统个人编程的替代方案。

通过观察表明,在协同学习环境中,学生能够在主动性、批判性思维和合作行为方面取得进步。协同学习的基础是假设两个或两个以上的人一起工作可以达到平衡的状态,在这个状态下,可以在小组成员之间交换和分享想法,从而产生新的想法和知识^[2]。但是,许多因素可能会影响协同学习的效率。将群体成员之间的群体组合、群体规模和个体差异列为可能影响协作的条件。实际情况表明,拥有一定核心的团队可以成就或破坏一个项目,因此在选择团队成员时应该采取非常谨慎的态度。同时考虑团队规模,整体团队组成,制定团队成员选择和协作标准是非

收稿日期:2019-04-10

基金项目:2017年河南省高等教育教学改革研究与实践项目“计算机网络技术专业课程体系与教学内容整体优化的研究与实践”(项目编号:2017SJGLX580);2019年度河南省高等学校重点科研项目“高职网络学习空间中智能推荐系统的设计与应用研究”(项目编号:19B520025)。

作者简介:姜赛达(1984-),男,河南商丘人,商丘职业技术学院计算机系讲师,研究方向:计算机网络和云计算。

常有必要的工作。鼓励小组互动,建议团体人数应该在五到七人之间,超过七人的团体可能会导致沟通问题。因此,计算机科学工作学习中的团队合作或编程任务协作中最困难的决定之一是如何确定团队中应该有多少人。目前该领域的研究并未精确确定适当的团队(团队)规模。本研究旨在通过实验经验确定协同计算机编程/学习环境中最佳团队规模。

二、相关工作

目前的趋势表明,软件工程领域的合作开发将会发生变化。针对这些变化研究的重点应该有所转移,以实现和适应这种合作变化。具体而言,小型软件将变得更受欢迎,甚至是大型软件由于更便捷的开发工具,将由更少的人完成。对于大型项目,除了开发人员和项目成员的协作需求之外,还包括质量保证工程师,构建工程师,架构师和运营经理。普遍认为,协同编程提高了软件设计的质量,减少了代码的不足,提高了技术能力,改善了团队沟通,并且被认为对参与者来说更加愉快。在协同编程中,两个程序员在一台计算机上以相同的设计,算法或代码协同工作。之前对这种编程风格的研究主要集中在学术环境中的讨论^[3]。在过去的几十年中,作为开发软件的改进方法已被多次提到。在协同编程中,两个程序员就像一个统一的、智能的有机体一起工作,负责这个工件的每个方面。一个合作伙伴,进行设计和编写代码,另一个合作伙伴不断积极地观察他的工作,观察缺陷,考虑替代方案,寻找资源以及考虑实施意义。合作伙伴按照约定定期切换角色,无论是阶段性工作还是整个项目,两者都是过程中始终如一的积极参与者并完全分享完成产品的所有权。

最近,随着越来越多的商业公司考虑使用它,学术界的注意力正在转向合作实践。观察表明,在软件开发团队中工作,特别是随着现代软件项目的复杂性和规模的增加,大多数专业程序员并不是单独工作。随着协调工作的需要越来越大;编程工作已经越来越多地成为社交组件。程序员通常会向团队成员寻求技术知识支持、建议或编程帮助。

三、研究方法

(一)对象

本研究的对象是计算机系 200 名自愿学生。这些学生被选为该项目的对象,是因为使用刚刚学习如何编程的学生,可以很容易地注意到协作的效果而不浪费太多资源。这些实验是在 2018/2019 学年第一学期学习 Java 计算机编程时进行的,持续了十三周。

(二)实验设计与设置

在该研究中组织了两个实验。第一个涉及不同

性别的团队组成。学生们被随机分成五个不同大小的平行小组,范围从 1 到 5。每个团队规模被复制六次。也就是说,在一个规模中,我们有 6 个平行团队独立协同工作一样的问题。这些团队由男女学生的混合体组成,但不一定相同。第二个实验涉及相同性别成分的团队(男性或女性)。第二个实验中只涉及一组并行团队。每个团队规模的协同时间和表现的平均值被确定并记录。为这个实验设计的题目是两个涉及对象使用的实际问题,涵盖编程概念,数据库和数据结构以及控制结构。每个并行团队提供每个阶段的报告。在实验期间,每个团队记录了给定问题的分析,编码,编译和调试的开始时间和结束时间。代码编译期间发生的错误数量也由不同的团队记录。

(三)实验实施

第一次实验于 2018 年 9 月 8 日上午 10 点左右进行,持续 5 个小时。该实验涉及不同性别成员的随机规模团队。在第一次实验后两周进行了大小为 1 至 5 的包含性别特征的团队(男性和女性)。该实验由研究人员监测和进行。研究人员担任实验期间的协调员。报告中填写了团队识别号,以便识别每个团队。获取实验数据,包含时间、编码、编译和调试效用。分析了实验期间每个团队的工作(分析,编码和调试所花费的时间)。从分析阶段到编译和调试阶段再到最终执行,团队被给予相同的编程问题以避免实验中获得的结果偏差。此外,研究人员确保在不同的团队之间没有相互联系。确定小组的一名成员担任编写员,一名担任打字员,一名担任协调员。所有错误和时间均由报告上的记录人员记录。

(四)变量

该实验设置的自变量是团队规模(1 至 5)和性别团队(男性和女性)。测量的因变量是:

第一,分析工作量:分析时间是团队理解问题所花费的时间。团队进行的分析基于:

解释对问题的理解;

识别和设计输入和输出变量,其数据类型及其来源(键盘,数据库或文件)或(显示器,文件或数据库);

使用适当的编程控制结构(选择和循环控制);

输出接口设计;

使用面向对象编程的一些概念、方法、类和对象。

第二,编码工作量:编码时间是团队将分析转换为选定的计算机编程语言所花费的时间。

第三,编译工作量:编译时间是将代码翻译成机器语言所花费的时间。

第四,调试工作量:调试时间是团队调试程序所花费的时间。

第五,错误数量:错误数量是编译期间记录的错误数量。

第六,团队的计划准确性:遵守问题规范,即完全解决所有问题的规范。

第一,团队的计划效率:效率取决于生成的代码总量,使用适当的编程逻辑和控制结构以及使用面向对象的概念。

第二,团队的有效性:增加在准确性和效率方面获得的分数。

第三,最有效的团队:团队分析和编程工作量最少(编码时间),最佳效果和最少的错误。

(五)对有效性的影响

实验的有效性问题决定了人们对一项措施在多大程度上可用性的衡量。因此,在本实验中,我们考虑了可能影响本研究结果有效性的两个重要方面。

1.对内部有效性的影响

内部有效性的影响是影响因变量的因素,我们考虑了三种这样的影响:选择效果;经验效果;仪器效果。

选择效果是由个体差异决定的。例如,如果实验对象仅由经验丰富的人进行,则他们的平均技能可能会被误认为实验效果的差异。我们通过为每个编程任务随机分配学生来限制这种效果,这样就可以在所有实验中分散个体差异。经验效果源于参与者通过经验改善的能力。随机分配学生并在同一时间段内进行实验限制了这种效果。仪器效果是由学生需要解决的问题,数据收集形式的差异或其他实验材料引起的。在这项研究中,这是非常微不足道。同样,整个团队使用了一组数据收集表格,团队也解决同样的问题。

2.对外部有效性的影响

对外部有效性的影响是限制我们将实验结果推广到实际工作中的效果。我们考虑了这些影响的三种来源:实验规模;问题普遍性;问题代表性。该实验是在学术环境中进行的,因此这限制了所获得的结果在实际环境中的检验。当实验主体不是来自从业人员时,可能存在对主体普遍性的影响,同时出现关于问题普遍性和代表性的影响。

四、结果和讨论

(一)实验 1:性别混合随机分配团队

表 1 和图 1 给出了具有五个不同团队规模的六个平行组的总体平均表现。

表 1 平均团队的表现

团队规模	平均分析时间(分钟)	平均编码时间(分钟)	平均有效性	平均编译错误
1	36.3	81.2	19.5	21
2	44.3	80	16.5	11
3	37.5	72.5	16.7	13
4	26.8	57.6	16.8	8
5	27.8	58.8	19.1	21

表 1 和图 1 显示,团队规模 1、2 和 3 的团队平

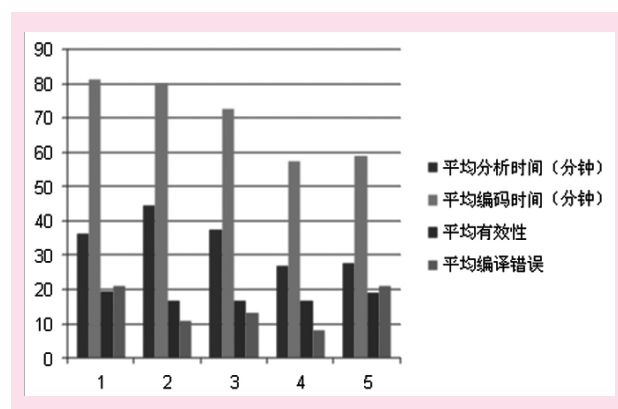


图 1 团队在分析、编码时间、错误和有效性方面的表现

均分析时间很长,团队规模 2 的时间最长。团队规模 4 和 5 似乎具有相同的平均分析时间。对于团队规模 1、2 和 3,平均编码时间也很高。团队规模 4 和 5 也具有或多或少相同的平均编码时间。至于衡量程序员准确性和效率的平均有效性,团队规模 1 和 5 的价值最高;随后是团队规模 4。团队规模 2、3 和 4 的差距很小。团队规模 4 的编译错误最小,而团队规模 1 和 5 则导致实验中出现最多的错误。最佳团队规模是根据一下标准获得的:

- 一是花在问题分析上的时间最少;
- 二是编码的最短时间;
- 三是最佳效果得分;
- 四是报告的 bug 数量最少。

从图 1 中可以推断,团队规模 4 最低限度地满足了上述四个标准。

(二)实验 2:随机分配具有相同性别成分的团队

使用不同大小的相同性别团队组合重复该实验。表 2 和图 2 给出了不同规模的单一男性性别团队获得的结果。对于男性团队,团队规模 3 的问题分析时间最长。团队规模 1、4 和 5 的值较低,边际值较小。团队规模为 2 时,编码时间最长。团队规模 4 的效率最高,其次是团队规模 3。团队规模 1 编制时间错误非常高,后面是团队规模 5。团队规模 4 的团队错误报告数量最少。使用我们设定的标准来确定最佳团队规模,还可以推断出男性团队获得的最佳团队规模是 4。表 3 和图 3 给出了不同规模的单一女性性别团队获得的结果。

表 2 男队的表现

团队规模	平均分析时间(小时)	平均编码时间(小时)	平均有效性	平均编译错误
1	0.42	3.44	18	59
2	0.67	4.67	18	10
3	0.87	3.98	24	6
4	0.42	3.14	26	3
5	0.40	4.05	14	28

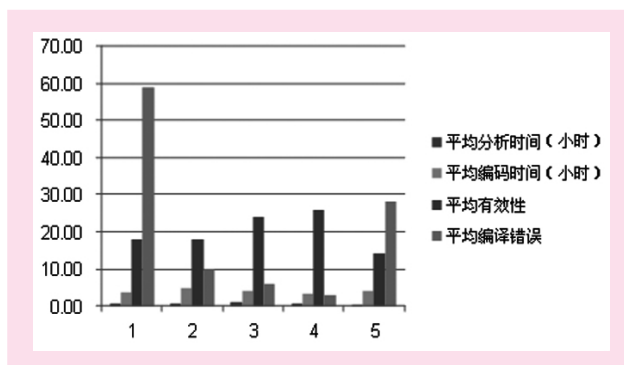


图2 男队的表现

表3 女队的表现

团队规模	平均分析时间(分钟)	平均编码时间(分钟)	平均有效性	平均编译错误
1	0.68	4.68	21	21
2	0.58	4.76	19	11
3	0.78	5.20	15	18
4	0.78	4.03	25	5
5	0.77	4.19	12	24

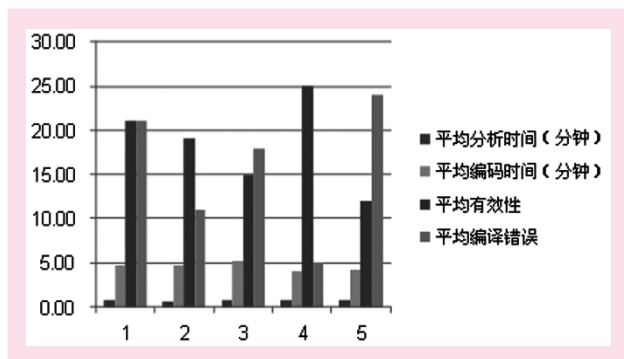


图3 女队的表现

对于女性团队,团队规模3、4和5在问题分析上花费的时间很长。团队规模4在编码上花费的时间最少,团队规模也报告了最少的错误。结果显示,团队规模4在实验中也达到了最佳水平。

(三)讨论结果

协同学习越来越被认为是一种改善学习成果的方法。在协同学习中,学生通过沟通概念来积极参与解决有关任务。培养了学生的团队合作能力。根据市场反馈的结果,虽然大多数毕业生的技术技能令人满意,但他们的团队合作能力往往不足。开发高质量软件的任务需要大量的分析,设计和编程,这也需要非常仔细的设计。在软件开发过程中要完成的各种任务要求参与人员被分为分析师,设计师和程序员。协同学习涉及将学生分组或配对以协同工作。在很长一段时间里,这种做法已被多次应用,作为开发软件的改进方法。

在计算机科学中编程团队合作或协作任务的最困难的决定之一是如何确定团队中应该有多少人。目前在该领域的研究还没有对适当的团队规模进行过精确的研究。诸如参与人数的各种因素影响协作的有效性。关于协同情境中影响效率的问题,列出了团队组成、组成人员、组的大小和个体差异等影响因素。

本研究旨在实现两个目的。首先,以软件项目团队合作的精神向参与研究的学生灌输和熟悉软件项目团队合作的精神。其次,是凭实验经验确定在学习和现实生活中的编程环境中可能需要有效(最佳)团队规模。为了实现这些目标,本研究中建立了两个不同的实验。列出了用于确定最佳团队的参数。参数是最佳团队应该在问题分析和编码上花费更少的时间,团队应该报告更少的错误,并且团队的有效性(效率和准确性)必须是很好的。两次实验的结果表明,四人团队合作开展的软件项目能够在最佳水平上有效地发挥作用。这一结果可归因于这样一个事实,即与四个人的合作可以提高参与者的分析和编程技能,从而减少花在任务上的时间。其中一人担任团队的协调员,另一人负责输入代码;其他人可以充当错误和错误的编码风格的检查员推动任务进程。在本研究中证明了四位优秀的程序员将成为计算机编程任务的最佳团队。

五、结论

这项研究中进行的两项独立实验表明,不熟悉计算机的学生在协同环境中完成编程/学习时,编程可以获益良多。研究表明学习/编程环境中可采用的最佳团队规模应为四。这将提高成员的有效性,减少在问题分析和编码上花费的时间,以及编程中可能产生的错误数量。

参考文献:

- [1] de Faria, E.S.J. and Adan-Coello, et al. Forming Groups for Collaborative Learning in Introductory Computer Programming Courses Based on Students' Programming Styles: An Empirical Study [J]. 36th ASEE/IEEE Frontiers in Education Conference, 2006, (10): 6-11.
- [2] Bryant S., Romero P. and du Boulay B. The Collaborative Nature of Pair Programming [J]. Lecture Notes in Computer Science, 2006, (4044): 53-64.
- [3] Müller, M.M. Two Controlled Experiments Concerning the Comparison of Pair Programming to Peer Review [J]. Journal of Systems and Software, 2005, (780), 166-179.

[责任编辑:胡大威]

An Empirical Study of Optimal Team Size in Computer Programming Collaborative Learning Environment

Jiang Sai-da

(Department of Computer, Shangqiu Vocational College, Shangqiu 476000, China)

Abstract: Collaborative programming has long been widely recognized as the best way of computer programming. Today, collaboration has proven to produce better results in the programming process. However, the optimal group size required for collaboration needs to be studied more deeply. The purpose of this paper is to instill and familiarize with the teamwork spirit of software projects and to determine the effective (best) team size required in the real-world programming/learning context. Two different experiments were organized and conducted to determine the parameters that determine the optimal team size. Voluntary participants of different genders were randomly divided into 5 parallel teams of different sizes, ranging from 1 to 5 in the first experiment. Each team size is replicated six times. The second experiment involved teams of the same sex (male or female) of different sizes. Compile-time errors recorded by problem analysis and coding time and for each team size record. Finally, the effectiveness of the team was analyzed. The study shows that collaboration is very beneficial for new computer programming learners. When learning is done with the company of others, they are easy to master the programming concept. The study also showed that the optimal team size that can be used in computer programming collaborative learning is four.

Key words: optimal team size; collaborative learning; collaborative programming; computer programming



(上接第 115 页)

120

Determination of Amino Acids and Analysis of Antioxidant Activity in Vitro

QIN Yu

(School of Bioengineering, Wuhan Polytechnic, Wuhan 430074China)

Abstract: The molecular weight distribution and amino acid composition of the ferrous chelates of fish scale peptides were detected by gel permeation chromatography and automatic amino acid analyzer. The scavenging effects of the scalar peptide ferrous chelate on DPPH free radicals and hydrogen peroxide (H_2O_2) were studied, and the scavenging ability of ascorbic acid, fish scale peptide and scaly peptide ferrous chelate salt was compared. Molecular weight determination and amino acid composition analysis showed that the relative molecular mass distribution range of fish scale polypeptide was 500~5000Da, the relative molecular mass distribution range of polypeptide ferrous chelate was 5000~10000Da. The content of glycine, glutamic acid, aspartic acid and proline in the ferrous chelates of the fish scale polypeptide is relatively high, and the content of tyrosine and histidine is relatively low. The results of in vitro antioxidant experiments showed that the sulphate peptide ferrous salt can scavenge the DPPH free radicals and hydrogen peroxide, ascorbic acid, fish scale peptide ferrous salt and fish scale peptides. It is a vitamin C <fish scale polypeptide ferrous salt> fish scale polypeptide, so the fish scale polypeptide ferrous chelate salt has a strong ability to scavenge DPPH free radicals and hydrogen peroxide.

Key words: fish scale polypeptide ferrous chelate; amino acid; antioxidant; DPPH free radical; hydrogen peroxide