World Transactions on Engineering and Technology Education
Vol.22, No.2, 2024

High-performance parallel computing on the Param Bilim supercomputer in higher education

Nursaule Karelkhan, Diana Meiirbekkyzy & Perizat Gapbarova

L.N. Gumilyov Eurasian National University
Astana, Kazakhstan

ABSTRACT: The escalating significance of computational power across diverse scientific, engineering and data-driven disciplines has elevated the necessity for robust high-performance computing (HPC) skills in educational settings. This article offers an extensive review and analysis of the current status of HPC education on a worldwide scale. Aims of this study are to delve into an extensive exploration of HPC education in various countries, evaluate the breadth and depth of curricular offerings and the accessibility of resources; determination of pedagogical features in teaching high-performance parallel computing to students; observation of availability and use of supercomputers at universities. In the final analysis, the article presents the results of parallel computations on the Param Bilim supercomputer, located at *L.N. Gumilyov* Eurasian National University, Astana, Kazakhstan. By consolidating all this information, the authors aim to identify opportunities for enhancement, and offer insights that can aid in fostering a more cohesive and effective landscape for HPC education.

INTRODUCTION

In today's era, revolutionary findings and innovations rely heavily on technology, data and advanced computing that leads to rapidly growing demands for computation requirements. In this connection, high performance computing (HPC) has been widely adopted in a wide range of research, as well as practical applications. HPC methods have the potential to aid scientists, engineers and other experts in tackling exceptionally intricate issues in the fields of linear algebra, physics, chemistry, biology, aerospace, etc. Utilising HPC involves combining modelling, algorithms, software creation and computational simulation, which has evolved into an essential foundation for advanced research in the field of cutting-edge fundamental science [1]. This matter is directly impacted by the education in HPC. This is one of the domains that critically needs a good course in the programme curriculum at universities due to its complexity.

LITERATURE REVIEW

Currently, numerous research papers are being published, since the relevance of HPC is rapidly increasing. For example, in the article by Spataro et al *High performance computing in modelling and simulation*, the authors discuss the impact of HPC on modelling and simulation, highlighting its crucial role in advancing science [2].

In the article: *A comparative analysis of resource allocation schemes for real-time services in high-performance computing systems*, Qureshi et al explore resource allocation schemes for real-time services, considering various architectures and quality of service criteria, and this analysis categorises and compares these schemes across distributed and non-distributed high-performance computing domains, offering researchers a consolidated platform to study and improve resource allocation for real-time services [3].

In their article: *Recent developments in high performance computing for remote sensing: a review*, Lee at al examine the latest progress in HPC as it relates to remote sensing challenges [4].

The paper: *A high performance computing platform for big biological data analysis* by Yeh at al, reveals that by using a reconfigurable HPC platform, the computational bottleneck can be removed, and data analysis can be accelerated [5].

In the paper: *Active learning pipeline for brain mapping in a high performance computing environment*, Michaleas outlines a prototype of a scalable active learning system that uses an advanced computing capabilities [6]. The implementation of this pipeline holds promise in significantly alleviating the manual annotation workload [6].

The findings of the article: *Supercomputers to improve the performance in higher education: a review of the literature*, by Fernández et al suggest that including supercomputers in learning processes has provided a wide range of new opportunities for learners, mainly due to the improvement in the quality of training obtained, which provides better results in the practical cases through real-life situations [7].

The challenges of crafting HPC courses are revealed by Holmes and Kureshi in the article: *Developing high performance computing resources for teaching cluster and grid computing courses* [8].

In the article by López and Baydal: *Teaching high-performance service in a cluster computing course*, the authors offer various instructional methods for a cluster computing course that could address some deficiencies [9].

In summary, the examined publications collectively affirm the pivotal role of HPC in advancing scientific endeavours. They underscore its application across diverse fields, such as real-time services, remote sensing, big biological data analysis and machine learning-driven brain mapping, showcasing its capacity to overcome computational bottlenecks and enhance research outcomes. Furthermore, the literature suggests that integrating supercomputers into higher education opens up new opportunities for learners, leading to improved training quality and better practical outcomes in real-world scenarios.

METHODOLOGY

In order to find out the current state of HPC education, the curricula within the scope of HPC of high-rated universities in the QS 2023 rankings [10], as well as Kazakhstan leading universities, have been reviewed by the authors of this article.

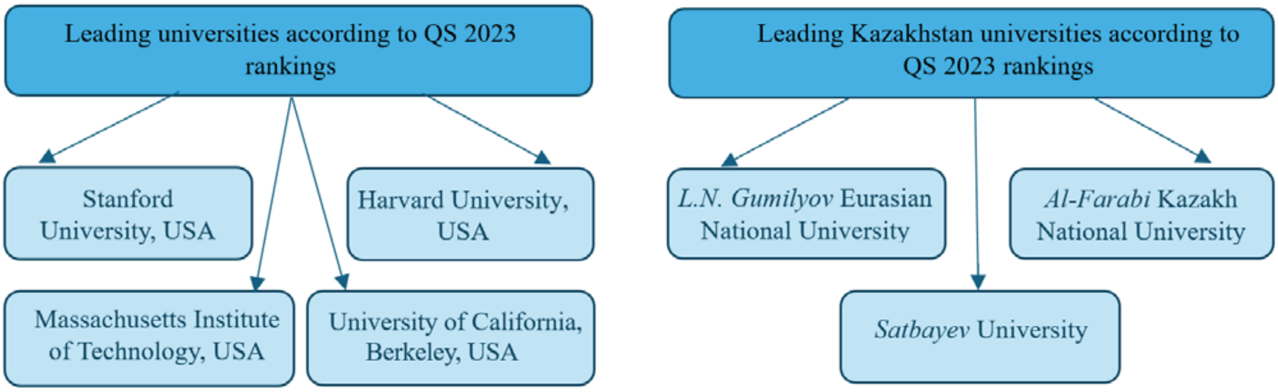

Figure 1: Leading universities according to the QS 2023 rankings.

Ranked no.1 in the QS 2023 rankings of world universities, Massachusetts Institute of Technology (MIT) in Cambridge, USA, covers a wide range of courses in the field of HPC, including parallel programming, algorithm optimisation, supercomputer architecture and parallel processing technologies [10]. MIT students can choose different programmes and courses on HPC, such as High Performance Computing Architecture, Parallel Programming for Multicore Machines with OpenMP and MPI, Parallel Computing, and others. These courses include theoretical lectures, practical exercises and project work that allow students to acquire the necessary skills in the field of HPC [11].

The Parallel Computing course at MIT is an introduction to applied parallel computing on modern supercomputers. It has a practical focus on understanding computational problems on the world's fastest machines. Topics covered include dense and sparse linear algebra, N-body problems, multigrid, fast-multipole, wavelets and Fourier transforms. Geometrical topics include partitioning and mesh generation. Other topics include applications-oriented architecture, understanding parallel programming paradigms, MPI, data parallel systems, Star-P for parallel Python and parallel MATLAB®, graphics processors, virtualisation, caches and vector processors [12].

A special feature of this course is the inclusion of very high-level languages (VHLLs) for parallel computing. This includes the Julia programming language. Julia is a high-level, high-performance dynamic language for technical computing, with a syntax familiar to users of other computing environments. It provides a sophisticated compiler, distributed parallel execution, numerical precision and an extensive library of mathematical functions [12].

At Stanford University, students use OpenHPC, Intel Parallel Studio, Environment Modules, and cloud-based architectures as part of the Introduction to High Performance Computing course. In addition, HPC clustering with remote physical hardware installation, high-speed network configuration and optimisation, and familiarisation with parallel programming and high-performance Python programs are included. To get the most out of the course, students need to be familiar with Bash and Python. Group work and projects are allowed and encouraged. All laboratory work and student projects are practiced in the specially designed High Performance Computing Center (HPCC) [13]. In addition, Stanford University has supercomputing resources that can be used to teach HPC. These resources provide students and researchers with the opportunity to gain hands-on experience using supercomputers in real-world projects [14].

Harvard University's high-performance computing for science and engineering curriculum consists of 21 lectures, six laboratories and individual or group projects, starting with an introduction to parallel computing and ending with CUDA technology. Covered topics: OpenMP, UMA/NUMA memory architectures, MPI, CUDA, etc. Programming languages - C/C++ or Fortran. The goal of the course project is to create a real program using the techniques learned in class to gain hands-on experience writing parallel code. The final report is written in the form of a proposal to request computing hours to the HPCC in order to test the project on a supercomputer. Faculty members evaluate student proposals in the role of the HPC admissions committee. Programs that meet the criteria will be tested on supercomputers [15].

The University of California, Berkeley, offers courses, such as GPU Programming for High Performance Computing, High Performance Computing for Science and Engineering and Applications of Parallel Computers. The applications of parallel computers curriculum includes the basics of parallel computer architecture and programming languages, and it also emphasises models commonly used in all applications requiring HPC. The goal of this course is writing programs that run fast while minimising programming effort. Topics covered: shared memory (OpenMP on a multicore laptop); distributed memory (MPI and UPC on a supercomputer); GPUs (CUDA and OpenCL); cloud computing (MapReduce, Hadoop and Spark) [16]. Also, in recognition of the increasing importance of research computing across many disciplines, UC Berkeley has developed the BRC high-performance computing service, as a way to grow and sustain HPC for the University.

As for Kazakhstan's universities, *Al-Farabi* Kazakh National University, which ranked 1st in the country and 150th in the QS rankings, offers the subjects: Parallel Computing System Architecture, Design and Analysis of Parallel Algorithms, High-performance Programming with Multi-Core and GPUs, and High-Performance Computing Models, in order to provide students with theoretical and practical training in the field of HPC. The courses cover bases of parallel computing system classification: Flynn's taxonomy, single-threaded instruction, single instruction/multiple data (SIMD), multithreaded command, multiple instruction/multiple data (MIMD); high performance processors, VLIW processors; NUMA architecture; working with OpenMP and creating parallel programs using the MPI library [17].

L.N. Gumilyov Eurasian National University provides courses, such as Methods of Teaching High-Performance Computing for the Bachelor degree programmes and Parallel Computing Cluster for the Master degree. These courses cover the fundamentals of parallel computing, introducing students to techniques for writing efficient code that can harness the power of parallel processing architectures. The HPC curriculum focused on HPC delve into the principles, architectures and tools used in HPC systems. This includes understanding hardware/software interactions, optimising algorithms for performance, and utilising parallel computing paradigms effectively. OpenMP, MPI, parallel algorithms, GPU architecture, CUDA threads, introduction to quantum computing and many other topics are covered. Also, the University has a supercomputer Param Bilim, which allows students to apply theoretical knowledge to real-world scenarios to gain practical skills [18].

The course called Parallel Programming of *Satbayev* University, the purpose of which is to study the basics of parallel programming and the development of thinking associated with high performance computing, focuses on fundamental knowledge of the subject area - methods of parallel programming on systems with distributed memory and systems with shared memory and methods for constructing parallel programs for solving system of linear algebraic equations (SLAE) problems, as well as problems solved by grid methods. The course provides an opportunity to get acquainted with the methods of parallel programming with shared variables, synchronisation of processes through access to shared resources, MPI and OpenMP parallel programming systems, create parallel programs for solving systems of linear equations using the Gauss method, etc [19].

Table 1: Courses/programmes and technologies used at the selected universities.

No.	University	Course/programme	Used technologies	Programming language
1	Massachusetts Institute of Technology	High Performance Computing Architecture, Parallel Programming for Multicore Machines with OpenMP and MPI, Parallel Computing,	– OpenMP – MPI – Matlab – MapReduce	– Python – Julia
2	Stanford University	Introduction to High Performance Computing	– OpenHPC – Intel Parallel Studio – Environment Modules – OpenMP – MPI	– Bash – Python
3	Harvard University	High-performance Computing for Science and Engineering	– CUDA – OpenMP	– C/C++ – Fortran

			– UMA/NUMA memory architectures – MPI	
4	University of California, Berkeley	GPU Programming for High-performance Computing, High Performance Computing for Science and Engineering, Applications of Parallel Computers	– OpenMP – MPI – UPC and UPC++ – CUDA/OpenCL	– C++
5	*Al-Farabi* Kazakh National University	Parallel Computing System Architecture, High-performance Programming with Multi-core and GPUs, High Performance Computing Models	– OpenMP – MPI	– C++
6	*L.N.Gumilyov* Eurasian National Univeristy	Parallel Programming, Methods of Teaching High Performance Computing	– OpenMP – MPI – Matlab	– C++
7	*Satbayev* University	Parallel Programming	– OpenMP – MPI	

In order to determine the effectiveness of HPC, and practically implement the scientific and theoretical bases of its introduction into the educational process, the authors have organised the motivational, content, and technical and technological aspects in the educational process at *L.N. Gumilyov* Eurasian National University as a pedagogical system (Table 2).

Table 2: Pedagogical system.

Motivational	Content	Technical and technological
Acquaintance with high-performance computing and the knowledge and skills associated with their mastery, understanding that it is necessary for an information technology professional	Having a theoretical understanding of high-performance computing	Proficiency in high-performance computing

According to the content structure, the course Methods of Teaching High-performance Computing was introduced to the Bachelor’s specialty 5B011100-Informatics. As the organisational structure, the Param Bilim supercomputer was used in the educational process during the teaching of special courses.

The Param Bilim supercomputer includes 16 core nodes equipped with 5110p Xeon Phi, nVidia K20X Card, NAS storage with 20 TB of usable storage space and RCS cards. The primary network is a low latency high-speed InfiniBand network along with essential software. A backup network based on Gigabit Ethernet has also been included [20].

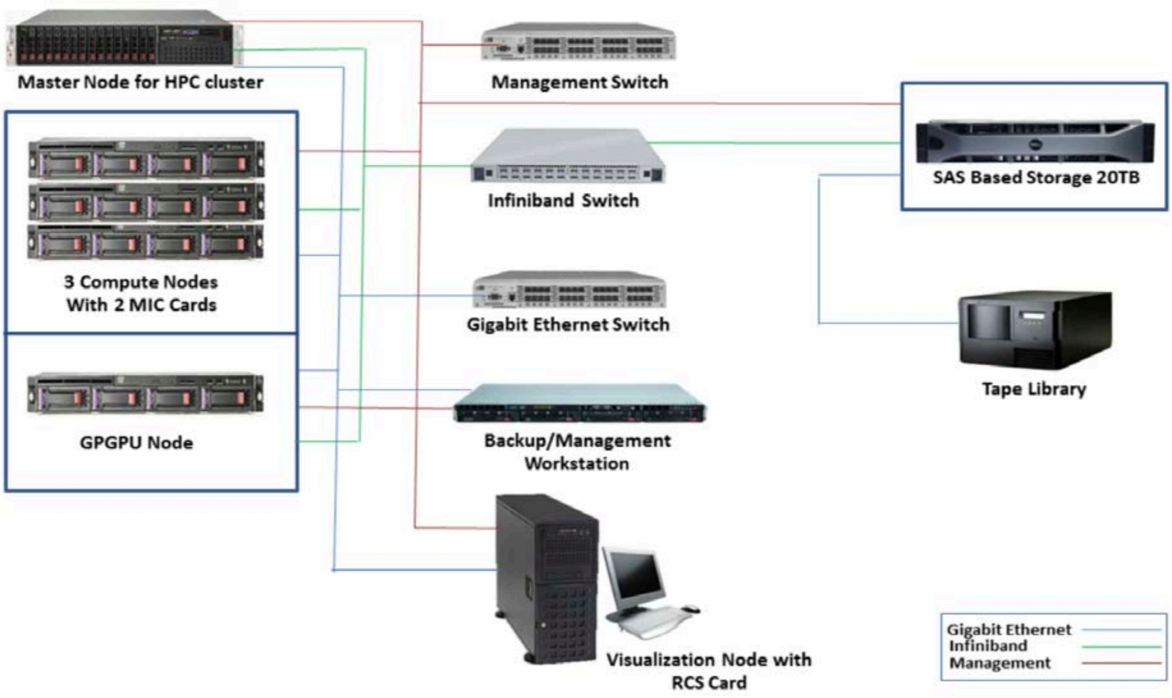

Figure 2: Param Bilim architecture diagram (image from the Param Bilim Project handbook by the HPC Solutions Group CDAC) [20].

RESEARCH RESULTS

On the Param Bilim supercomputer, the authors of this article performed sequential and parallel computations of the system of linear equations ax = b using the inverse matrix method (x = a-1*b) and determined the efficiency of parallel computation.

As an experiment, the authors calculated the multiplication of a 30,000-size vector by a 30,000*30,000-size matrix on a computer with a 4-core Intel(R)Core™ i5-7500T CPU 2.7 GHz processor sequentially, and performed parallel computations on the Param Bilim supercomputer. Codes of sequential and parallel programs in C++ are shown in Figure 3.

```
#include <iostream>
#include <cstdlib>
#include <ctime>

#define N 30000

int A[N][N];
int B[N];
int C[N];

void initMatandVec(){
    for(int i=0; i<N; ++i){
        for (int j=0; j<N; ++j){
            A[i][j]=rand()%10;
        }
        B[i]=rand()%10;
    }
}
void multiply(){
    for (int i=0; i<N; ++i){
        C[i]=0;
        for(int j=0; j<N; ++j){
            C[i]+=A[i][j]*B[j];
        }
    }
}
int main() {
    srand(time(0)); // Seed random number generator
    initMatandVec();
    clock_t start_time = clock();
    multiply();
    clock_t end_time = clock();
    double elapsed_time = double(end_time - start_time) / CLOCKS_PER_SEC;

    std::cout <<"Elapsed time:"<<elapsed_time<<"seconds."<<std::endl;

    return 0;
}
```

a)

```
#include <iostream>
#include <cstdlib>
#include <ctime>
#include <omp.h>

#define N 30000

int A[N][N];
int B[N];
int C[N];

void initMatandVec(){
    #pragma omp parallel for
    for(int i=0; i<N; ++i){
        for (int j=0; j<0; ++j){
            A[i][j]=rand()%10;
        }
        B[i]=rand()%10;
    }
}
void multiply(){
    #pragma omp parallel for
    for (int i=0; i<N; ++i){
        C[i]=0;
        for(int j=0; j<N; ++j){
            C[i]+=A[i][j]*B[j];
        }
    }
}
int main() {
    initMatandVec();
    double start_time = omp_get_wtime();
    multiply();
    double end_time = omp_get_wtime();
    double ellapsed_time = end_time - start_time;

    std::cout <<"Elapsed time:"<<elapsed_time<<"seconds."<<std::endl;

    return 0;
}
```

b)

Figure 3: a) Sequential program and b): Parallel program.

Table 3: Results of experimental computations.

Size of vector and matrices	Sequential computing time T_1	Param Bilim (16 core)	
		Computation time, T_{16}	Acceleration factor, S
100	0.010198	0.0146499	0.696114
500	0.030593	0.016345	1.871704
1000	0.0727294	0.0125579	5.791526
5,000	1.515987	0.0309549	48.974056
10,000	13.740121	0.075875	181.0889
15,000	38.170261	0.16527	230.956986
20,000	68.57965	0.226519	302.754515
25,000	107.78473	0.44294	243.339346
30,000	4851.5714	0.575069	8439.5031

It can be seen from Table 3 that the use of parallel camputations has been carried out at a high speed starting from 5,000. In addition, the computation on the ParamBilim supercomputer showed significantly higher efficiency.

After completing the Methods of Teaching High-performance Computing course, all participating students were asked to complete a survey to evaluate the course's contribution to education and provide their feedback. To evaluate the results of the study, a survey of 48 students was conducted on Google Forms. Twenty-four students from the experimental group and 24 from the control group, who did not take the Methods of Teaching High-performance Computing course comprised the tested population. The survey contained questions regarding the level of motivation, technical and technological level, and the level of content as shown in Table 4 and Figure 4.

Table 4: Results of implementing the course Methods of Teaching High-performance Computing.

No.	Course	Number of students
Students in experimental group		
1	Methods of Teaching High-performance Computing	24
Students in control group		
1	Solving Olympiad problems	24
Overall		48

Students of the experimental group verified that the course enhanced their spatial thinking abilities and enriched their understanding of the subject matter.

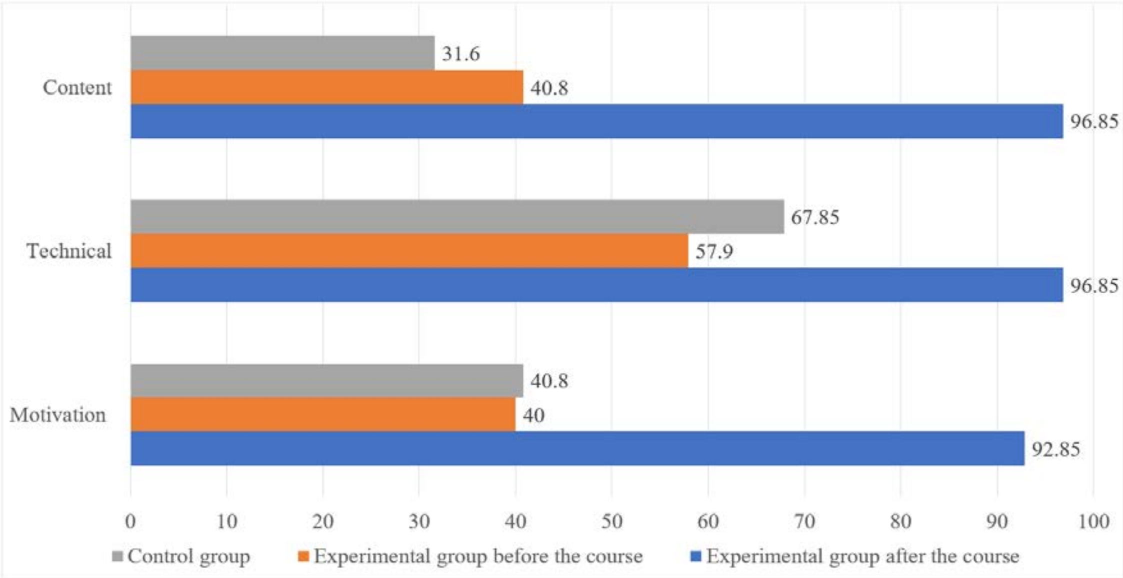

Figure 4: Results of a survey conducted to evaluate the course efficacy.

CONCLUSIONS

Teaching high-performance computing in universities is an important aspect of modern education. Due to the rapid development of information technologies, high-performance computing is becoming an integral part of scientific and engineering research, as well as industrial processes.

Teaching high-performance computing enables students to learn modern methods and technologies for working with large amounts of data, efficient algorithms and clustered computing. As graduates, they will work in science, industry, finance, medicine and other sectors, and their educational institution must ensure that they are competitive, qualified specialists ready for the labour market to solve complex problems in various fields.

As a result of the analysis of the situation of teaching high-performance computing, it was determined that universities use various software equipment and technologies in education. Among them, OpenMP and MPI technologies for algorithm computations, C++ programming language are often used. All analysed foreign and Kazakhstani universities are equipped with their own supercomputers used for research and teaching of practicing high-performance computing.

However, it can be seen, that in the educational programmes of Kazakhstani higher education institutions as compared to foreign higher education institutions, less attention is paid to the experience part and projects that would allow students to acquire knowledge and skills more effectively and in-depth.. It is hoped that within the coming years, this issue can be addressed, thus enhancing the educational programmes and equipping learners with the skills needed to solve the complex challenges of the digital era.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude to the India - Kazakhstan Centre of Excellence in Information and Communication Technology headed by Anatoliy Kremenchutskiy based in *L.N. Gumilyov* Eurasian National University, Astana, Kazakhstan, for its support in conducting this research and providing the necessary resources.

REFERENCES

1. Nielsen, E.J. and Diskin, B., High-performance aerodynamic computations for aerospace applications. *Parallel Computing*, 64, 20-32 (2017).

2. Spataro, W., Trunfio, G.A. and Sirakoulis, G.C., High performance computing in modelling and simulation. *The Inter. J. of High Performance Computing Applications*, 31, **2**, 117-118 (2017).
3. Qureshi, M.S., Qureshi, M.B., Fayaz, M., Mashwani, W.K., Belhaouari, S.B., Hassan, S. and Shah, A., A comparative analysis of resource allocation schemes for real-time services in high-performance computing systems. *Inter. J. of Distributed Sensor Networks*, 16, **8**, 1550147720932750 (2020).
4. Lee, C.A., Gasster, S.D., Plaza, A., Chang, C.I. and Huang, B., Recent developments in high performance computing for remote sensing: a review. *IEEE J. of Selected Topics in Applied Earth Observations and Remote Sensing*, 4, **3**, 508-527 (2011).
5. Yeh, C.W., Huang, C.W., Yang, C.L. and Wang, Y.T., A high performance computing platform for big biological data analysis. *Proc. 9th Inter. Conf. on Applied System Innov. (ICASI)*, Chiba, Japan, 68-70 (2023).
6. Michaleas, A., Active learning pipeline for brain mapping in a high performance computing environment. *Proc. IEEE High Performance Extreme Computing Conf. (HPEC)*, Waltham, USA, 1-6 (2020).
7. Fernández, A., Fernández, C., Miguel-Dávila, J.A., Conde, M.A. and Matellán, V., Supercomputers to improve the performance in higher education: a review of the literature. *Computers & Educ.*, 128, 353-364 (2019).
8. Holmes, V. and Kureshi, I., Developing high performance computing resources for teaching cluster and grid computing courses. *Procedia Computer Science*, 51, 1714-1723 (2015).
9. López, P. and Baydal, E., Teaching high-performance service in a cluster computing course. *J. of Parallel and Distributed Computing*, 117, 138-147 (2018).
10. QS World University Rankings 2023: Top Global Universities (2023), 04 February 2024, https://www.topuniversities.com/world-university-rankings/2023?tab=indicators
11. Parallel Programming for Multicore Machines using OpenMP and MPI (2010), 04 February 2024, https://ocw.mit.edu/courses/12-950-parallel-programming-for-multicore-machines-using-openmp-and-mpi-january-iap-2010/pages/syllabus/
12. Parallel Computing (2011), 04 February 2024, https://ocw.mit.edu/courses/18-337j-parallel-computing-fall-2011/pages/syllabus/
13. Introduction to High Performance Computing. Stanford High Performance Computing Center, 04 February 2024, https://hpcc.stanford.edu/teaching
14. Student Immersion Project (2019), 04 February 2024, https://hpcc.stanford.edu/
15. Wermelinger, F., High Performance Computing for Science and Engineering (2021), 06 February 2024, https://harvard-iacs.github.io/2022-CS205/pages/cs205_syllabus.pdf
16. Applications of Parallel Computers, 06 February 2024, https://www2.eecs.berkeley.edu/Courses/CSC267/
17. Informatika Mamandygy Boiynsha Tiptyk Oqu Bagdarlamalary (2017), 06 February 2024, https://www.kaznu.kz/content/files/pages/folder17397/5%D0%92060200-%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0.pdf (in Kazakh).
18. Param Bilim. India Kazakhstan Centre of Excellence in Information and Communication Technology, 06 February 2024, http://ikceict.enu.kz/index.php?view=about0
19. Curriculum Program *Computer Science* (2020), 06 February 2024, https://official.satbayev.university/download/document/18523/Computer%20Sience%20B%202020%20%D1%80%D1%83.pdf
20. Param Bilim Project Handbook by HPC Solutions Group CDAC, 06 February 2024, http://ikceict.enu.kz/views/pages/PARAM_BILIM_handbook_final.pdf