

ზუსტ და საბუნებისმეტყველო მეცნიერებათა ფაკულტეტი
კომპიუტერული მეცნიერების დეპარტამენტი

სამაგისტრო პროგრამა

კომპიუტერული მეცნიერება

Computer Science

კურსდამთავრებულს მიენიჭება აკადემიური ხარისხი:

კომპიუტერული მეცნიერების მაგისტრი

Master of Computer Science

1. **სამაგისტრო პროგრამის დასახელება:** კომპიუტერული მეცნიერება, Computer Science
2. **მისანიჭებელი აკადემიური ხარისხი:** კომპიუტერული მეცნიერების მაგისტრი, Master of Computer Science
3. **სამაგისტრო პროგრამის ხელმძღვანელი:**

სახელი, გვარი
თანამდებობა

აკადემიური ხარისხი

სამუშაო ადგილი

საკონტაქტო ინფორმაცია

ალექსანდრე გამყრელიძე
პროფესორი,

მეცნიერებათა დოქტორი;

თბილისის სახელმწიფო უნივერსიტეტი,
ზუსტ და საბუნებისმეტყველო მეცნიერებათა
ფაკულტეტი, კომპიუტერული მეცნიერების
დეპარტამენტი

877-44-65-25, sandro@hinkali.com

სამაგისტრო პროგრამის ხელმძღვანელი:
სახელი, გვარი

თანამდებობა

აკადემიური ხარისხი

სამუშაო ადგილი

საკონტაქტო ინფორმაცია

კობა გელაშვილი

პროფესორი,

მეცნიერებათა დოქტორი;

თბილისის სახელმწიფო უნივერსიტეტი,
ზუსტ და საბუნებისმეტყველო მეცნიერებათა
ფაკულტეტი, კომპიუტერული მეცნიერების
დეპარტამენტი

599-11-48-50, koba.gelashvili@tsu.ge

შენიშვნა: სამაგისტრო პროგრამის ხელმძღვანელების CV-ები შეგიძლიათ იხილოთ დანართში, რომელშიც აღწერილია პროგრამის განხორციელებისთვის საჭირო ადამიანური რესურსები.

4. **პროგრამის მოცულობა კრედიტებით:** 120 კრედიტი (პროგრამისთვის სავალდებულო კურსები – 70 კრედიტი, არჩევითი კურსები – 20 კრედიტი, სამაგისტრო ნაშრომი – 30 კრედიტი)
5. **სწავლების ენა:** ქართული. დარგობრივი სემინარები ჩატარდება ქართულ ან ინგლისურ ენებზე.

6. სამაგისტრო პროგრამის საკვალიფიკაციო დახასიათება:

საგანმანათლებლო პროგრამის მიზანი, რომელიც მიმართულია სტუდენტის კომპეტენციების გამომუშავებასა და დასაქმების სფეროს განსაზღვრაზე

სამაგისტრო პროგრამის მიზანია მაგისტრს მისცეს ცოდნა თანამედროვე კომპიუტერული მეცნიერების ძირეულ კომპონენტებში (ალგორითმების აგება და ანალიზი, ალგორითმების სირთულე, მონაცემთა სტრუქტურების თეორიული და პრაქტიკული ასპექტები, დაპროგრამების თანამედროვე ნიშნები, კომპიუტერული მეცნიერების განვითარების ტენდენციები და აუცილებელი მათემატიკური ფუნდამენტი) და გამოუმუშაოს მიღებული ცოდნის შემოქმედებითი და ეფექტური გამოყენებისთვის აუცილებელი უნარები.

კომპიუტერული მეცნიერების მიმართულება შრომის ბაზარზე დიდი აქტუალობით და მაღალი მოთხოვნით გამოირჩევა. მსოფლიოს დაახლოებით 2500 წამყვან უნივერსიტეტში აქტუალურია ამ მიმართულების სწავლებისა და კვლევის საკითხები. ჩვენს მიერ წარმოდგენილი სამაგისტრო პროგრამა ეფუძნება საუნივერსიტეტო გარემოში კომპიუტერული მეცნიერების სწავლების მსოფლიოში აღიარებულ მეთოდოლოგიას (Computing Curricula 2005, <http://www.computer.org/education/cc2005>). ეს მეთოდოლოგია აპრობირებულია (იხ. http://ed.sjtu.edu.cn/rank/2005/ARWU2005_Top100.htm) წამყვან უნივერსიტეტებში და შემუშავებულია ისეთი ცნობილი საზოგადოებების მიერ, როგორებიცაა: The association for Computer Machinery (ACM, <http://www.acm.org>), The Computer Society (IEEE-CS, <http://computer.org>).

სწავლის შედეგად, მაგისტრი იძენს ცოდნასა და უნარ-ჩვევებს, რაც აუცილებელია მისი შრომითი საქმიანობის წარმატებისთვის. ამ ტიპის სპეციალისტისთვის განსაკუთრებით მნიშვნელოვანია მრავალმხრივობა, საქმიანობის პრაქტიკული და თეორიული ასპექტების შეთანხმებულობა, კომუნიკაბელურობა, სუბიექტური და ობიექტური ფაქტორების ადეკვატური აღქმა.

როგორც წამყვანი უნივერსიტეტების გამოცდილება გვიჩვენებს, კომპიუტერული მეცნიერების მიმართულების მაგისტრატურაში მიღებული ცოდნა წარმოადგენს წარმატებული კარიერის გარანტიას. დასაქმების სფერო მოიცავს ანალიტიკურ, პრაქტიკულ, დამოუკიდებელ, სამეცნიერო-კვლევით (თუ მაგისტრი დოქტორანტურაში მოისურვებს სწავლის გაგრძელებას), სასწავლო, საკონსულტაციო საქმიანობას.

სწავლის შედეგი

კომპიუტერული მეცნიერების მაგისტრს შეეძლება რეალური ინტერესის მქონე პრობლემების კლასიფიცირება, მათთვის თეორიული მეთოდების და ტექნოლოგიური საშუალებების შერჩევა, ამოცანის დამუშავება და რეალიზება. კომპიუტერული მეცნიერების მაგისტრს უნდა შეეძლოს თავისი ცოდნის ეფექტური გამოყენება როგორც ინდივიდუალური, ასევე გუნდური მუშაობის შემთხვევაში.

სწავლის შედეგად მაგისტრი:

- შეიძენს როგორც ფუნდამენტური, ასევე შედარებით სპეციალური სახის ალგორითმების შესახებ მრავალმხრივ ცოდნას;
- შეისწავლის დაპროგრამების თანამედროვე ტექნოლოგიებს;
- შეიძენს კომუნიკაციური, ადამიანთმორისი და ჯგუფური მუშაობის უნარ-ჩვევებს;
- შეიძენს ანალიზური და კრიტიკული აზროვნების უნარს, რაც ასევე მოიცავს შემოქმედებით მიდგომას და ეთიკური ღირებულებების დაცვას;
- სპეციფიკურ უნარ-ჩვევებს წარმატებული კარიერის შესაქმნელად.

დარგობრივი კომპეტენციები, ცოდნა და გაცნობიერება

- კომპიუტერული მეცნიერების საფუძვლების ცოდნა;
- კომპიუტერული მეცნიერების ინფრასტრუქტურის ცოდნა;
- ანალიზის, აგების და იმპლემენტაციის მეთოდების ცოდნა;
- კომპიუტერული მეცნიერების განვითარების ტენდენციების ცოდნა;

დარგობრივი კომპეტენციები, ცოდნის პრაქტიკაში გამოყენება

- კომპიუტერული მეცნიერების თეორიული მეთოდების საშუალებით, კონკრეტული პრობლემების ანალიზის და შესაბამისი ალგორითმების აგების უნარს;
- პროგრამირების თანამედროვე ტექნოლოგიების საფუძველზე კონკრეტული პრობლემების შესაბამისი ალგორითმების ეფექტური იმპლემენტაციის უნარს;

ზოგადი / ტრანსფერული კომპეტენციები

დასკვნის უნარი

- აბსტრაქტული აზროვნების, ანალიზისა და სინთეზის უნარი;
- პრობლემის იდენტიფიცირების, დასმისა და გადაწყვეტის უნარი;
- გააზრებული გადაწყვეტილების მიღების უნარი;

კომუნიკაციის უნარი

- ინფორმაციული და საკომუნიკაციო ტექნოლოგიების გამოყენების უნარი სხვადასხვა წყაროდან ინფორმაციის მოძიების, დამუშავების და სათანადო დონეზე პრეზენტაციის მიზნით;
- მსჯელობისა და მისგან გამომდინარე დასკვნების ნათლად, ზუსტად და ადრესატისათვის მისაღები ფორმით მიწოდების უნარი, როგორც ზეპირად ისე წერილობით ქართულ და უცხოურ ენებზე;

სწავლის უნარი

- ვერბალური და წერილობითი ინფორმაციის აღქმის უნარი;
- დამოუკიდებლად მუშაობის უნარი;
- გუნდში მუშაობის უნარი;

ღირებულებები

- პროფესიული ეთიკის სტანდარტების დაცვა;
- კომპიუტერული მეცნიერების მაგისტრი შეითვისებს და შეაფასებს ღირებულებათა და ფასეულობათა იმ სისტემას, რაც მიღებულია საუნივერსიტეტო გარემოში.

7. პროგრამაზე დაშვების წინაპირობა:

მინიმუმ ბაკალავრის ხარისხი;

საერთო სამაგისტრო გამოცდა;

სულ მცირე 10 ECTS კრედიტი უმაღლესი მათემატიკის დისციპლინებში;

სულ მცირე 10 ECTS კრედიტი დაპროგრამების ენებში;

გამოცდა ინფორმატიკაში.

8. სწავლის შედეგების მიღწევის მეთოდები:

პროგრამაში გამოყენებულია სწავლების ისეთი სტანდარტული მეთოდები, როგორცაა:

- ვერბალური,
- წერითი,
- წიგნზე მუშაობის მეთოდები.

ამავე დროს განსაკუთრებული ყურადღება ეთმობა პრაქტიკულებს და პროექტებზე ჯგუფურ მუშაობას. ამ მიდგომების მიზანს წარმოადგენს, მისცეს სტუდენტებს რეალური სამუშაო გამოცდილება. როგორც წესი, პროექტებზე მუშაობა ხდება ჯგუფებში და საჭიროებს მიღებული თეორიული ცოდნის პრაქტიკულ გამოყენებას.

9. სტუდენტის ცოდნის შეფასების სისტემა

სტუდენტების ცოდნის შეფასების სისტემა და კრიტერიუმები განსაზღვრულია სასწავლო კურსების სილაბუსებში. ეს მოიცავს შუალედურ და დასკვნით გამოცდებს. ასევე იმ სილაბუსებში, სადაც ხდება პროექტზე მუშაობა, განსაზღვრულია პროექტების შეფასების კრიტერიუმები. პროგრამა ითვალისწინებს შეფასებას სასწავლო-სამეცნიერო სემინარს სწავლების მესამე სემესტრში, რაც მიზნად ისახავს ერთი მხრივ სამეცნიერო კომპონენტის სრულად რეალიზებას და მეორე მხრივ სამაგისტრო ნაშრომის მაღალ დონეზე შესრულებისთვის მის მოტივირებას.

სამაგისტრო ნაშრომის შესასრულებლად დაგეგმილია 30 კრედიტი IV სემესტრში. ნაშრომის მოცულობა უნდა იყოს არა უმეტეს 50 გვერდისა. ნაშრომში მკაფიოდ უნდა ჩანდეს დარგის თანამედროვე მდგომარეობის კარგი ცოდნა, ჩამოყალიბებული უნდა იყოს გამოსაკვლევი პრობლემის არსი, გამოკვეთილი უნდა იყოს მიღებული შედეგები და ამ შედეგების გამოყენების მიმართულებები. სასურველია ახლდეს პროგრამული რეალიზაციების ამსახავი დანართი.

10. სასწავლო გეგმა, სწავლების ორგანიზების თავისებურებების მითითებით.

პროგრამა შედგება სავალდებულო და არჩევითი კურსებისაგან. სავალდებულო კურსების მოცულობაა 70 კრედიტი, სამაგისტრო ნაშრომს ეთმობა 30 კრედიტი, ხოლო 20 კრედიტი - არჩევით საგნებს. რამდენიმე არჩევითი კურსი საზიაროა მომიჯნავე სამაგისტრო პროგრამებთან. სასწავლო გეგმა იხილეთ დანართში.

11. სწავლის გაგრძელების საშუალება: სწავლის დამთავრების შემდეგ მაგისტრი შეძლებს სწავლის გაგრძელებას დოქტორანტურაში.

12. კურსდამთავრებულთა დასაქმების სფეროები: სამეცნიერო-საგანმანათლებლო დაწესებულებები, საბანკო-საფინანსო დაწესებულებები, სხვადასხვა კავშირგაბმულობის დაწესებულებები, ჯანდაცვის ორგანიზაციები, სამეწარმეო ორგანიზაციები, სატრანსპორტო სამართავი სისტემები და სხვ.

13. პროგრამის განხორციელებისათვის საჭირო მატერიალურ-ტექნიკური ბაზა: თსუ ზუსტ და საბუნებისმეტყველო მეცნიერებათა ფაკულტეტი; თსუ-ს სამეცნიერო ბიბლიოთეკა, კომპიუტერული ბაზები, რესურსცენტრები და სხვა. თსუ ი. ვეკუას სახელობის გამოყენებითი მათემატიკის ინსტიტუტის ბიბლიოთეკა. მაგისტრანტებისთვის გათვალისწინებულია უახლესი ტექნიკითა და ინტერნეტთან წვდომით აღჭურვილი კომპიუტერული კლასის გამოყენება. მათთვის ხელმისაწვდომია საკმაოდ მდიდარი ელექტრონული ბიბლიოთეკა, რომელიც ბოლო წლებში შეგროვდა ინსტიტუტის თანამშრომლების მიერ web - რესურსების ბაზაზე.

14. ფინანსური უზრუნველყოფა: პროგრამას ფინანსურად უზრუნველყოფს თსუ.

15. მისაღები კონტინგენტი: მატერიალური და ადამიანური რესურსებიდან გამომდინარე პროგრამაზე შესაძლებელია 12 მაგისტრანტის მიღება.

სამაგისტრო პროგრამის სტრუქტურა - კომპიუტერული მეცნიერება

№	სასწავლო კურსის დასახელება	სასწავლო კურსის სტატუსი: სავალდებულო, არჩევითი	საკონტაქტო/ დამოუკიდებელი მუშაობის საათების რაოდენობა ლექ/პრ/ლაბ/სემ	ლექტორი/ ლექტორები	კურსზე დაშვების წინაპირობა	კრედიტების საერთო რაოდენობა	კრედიტების განაწილება			
							სემესტრები			
							I	II	III	IV
1.	კონკრეტული მათემატიკა	სავალდებულო	30/95 1/1/0/0	გიორგი ჭელიძე	-	5	5			
2.	მონაცემთა სტრუქტურები	სავალდებულო	60/190 2/1/0/1	კობა გელაშვილი	-	10	10			
3.	ფუნდამენტური ალგორითმები	სავალდებულო	45/205 2/1/0/0	ალექსანდრე გამყრელიძე	-	10	10			
4.	დაპროგრამების ენა ასემბლერი და კომპიუტერის არქიტექტურა	სავალდებულო	45/80 1/2/0/0	მიხეილ თუთბერიძე	-	5	5			
5.	დაპროგრამების ენების შედარებითი ანალიზი	სავალდებულო	30/95 1/0/0/1	ირინა ხუციშვილი	-	5		5		
6.	პროგრამული უზრუნველყოფის ინჟინერია	სავალდებულო	60/190 0/0/2/2	ბიძინა მიდოდაშვილი	-	10		10		
7.	სირთულის თეორია	სავალდებულო	60/190	ალექსანდრე		10		10		

			2/2/0/0	გამყრელიძე	3					
8.	სემინარი კომპიუტერულ მეცნიერებაში	სავალდებულო	45/80 0/0/0/3	ალექსანდრე გამყრელიძე	3	5			5	
9.	ფუნდამენტური ამოცანების სწრაფი ალგორითმები	სავალდებულო	30/220 1/1/0/0	ალექსანდრე გამყრელიძე	-	10			10	
10.	მონაცემთა ბაზები	არჩევითი	30/95 1/0/1/0	მანანა ხაჩიძე	-	5		5		
11.	კომპიუტერული ტოპოლოგია	არჩევითი	45/80 1/2/0/0	გოდერძი ფრუიძე	3	5		5		
12.	ბიბლიოთეკა Windows API	არჩევითი	45/80 1/2/0/0	მიხეილ თუთბერიძე	-	5		5		
13.	კომპიუტერული ალგებრის ალგორითმები	არჩევითი	45/80 1/2/0/0	რევაზ ქურდიანი	3	5			5	
14.	.NET პლატფორმის თანამედროვე ტექნოლოგიები	არჩევითი	45/80 1/1/1/0	ბიძინა მიდოდაშვილი	-	5			5	
15.	კომპიუტერული გეომეტრიის ალგორითმები	არჩევითი	45/80 1/2/0/0	ალექსანდრე გამყრელიძე	3	5			5	

16.	ინფორმაციის მოძიების თანამედროვე მეთოდები	არჩევითი	30/95 1/1/0/0	ალექსანდრე გამყრელიძე, ლევან კასრაძე	-	5			5	
17.	პარალელური ალგორითმები	არჩევითი	45/80 1/2/0/0	ალექსანდრე გამყრელიძე	-	5			5	
18.	დარგობრივი სემინარი (ინგლისურენოვანი) *	არჩევითი	15/35		-	2				
			25/50			3				
			30/95			5				
19.	უცხოური ენა 1	არჩევითი	60/65		-	5		5		
20.	უცხოური ენა 2	არჩევითი	60/65		-	5			5	
	სამაგისტრო ნაშრომი	სავალდებულო			-	30				30
	სულ					120	30	30	30	30

* - შესაძლებელია გააქტიურდეს ნებისმიერ სემესტრში.

<p style="text-align: center; color: blue;">დარგობრივი კომპეტენციები</p>		ცოდნა და გაცნობიერება				
		ცოდნის პრაქტიკაში გამოყენება				
კონკრეტული მათემატიკა		კომპიუტერული მეცნიერების საფუძვლების ცოდნა				
მონაცემთა სტრუქტურები		კომპიუტერული მეცნიერების ინფრასტრუქტურის ცოდნა				
ფუნდამენტური ალგორითმები		ანალიზის, აგების და იმპლემენტაციის მეთოდების ცოდნა				
დაპროგრამების ენა ასემბლერი და კომპიუტერის არქიტექტურა		კომპიუტერული მეცნიერების განვითარების ტენდენციების ცოდნა				
დაპროგრამების ენების შედარებითი ანალიზი		კომპიუტერული მეცნიერების თეორიული მეთოდების საშუალებით, კონკრეტული პრობლემების ანალიზის და შესაბამისი ალგორითმების აგების უნარს				
პროგრამული უზრუნველყოფის ინჟინერია		პროგრამირების თანამედროვე ტექნოლოგიების საფუძველზე კონკრეტული პრობლემების შესაბამისი ალგორითმების ეფექტური იმპლემენტაციის				
სირთულის თეორია						
სემინარი კომპიუტერულ მეცნიერებაში						
ფუნდამენტური ამოცანების სწრაფი ალგორითმები						

მონაცემთა ბაზები						
კომპიუტერული ტოპოლოგია						
ბიბლიოთეკა Windows API						
კომპიუტერული ალგებრის ალგორითმები						
.NET პლატფორმის თანამედროვე ტექნოლოგიები						
კომპიუტერული გეომეტრიის ალგორითმები						
პარალელური ალგორითმები						
ინფორმაციის მოძიების თანამედროვე მეთოდები						
სამაგისტრო ნაშრომი						

ზოგადი კომპეტენციები

	აბსტრაქტული აზროვნების, ანალიზისა და სინთეზის უნარი	პრობლემის იდენტიფიცირების, დასმისა და გადაწყვეტის უნარი	გააზრებული გადაწყვეტილების მიღების უნარი	ინფორმაციული და საკომუნიკაციო ტექნოლოგიების გამოყენების უნარი სხვადასხვა წყაროდან ინფორმაციის მოძიების, დამუშავების და სათანადო დონეზე პრეზენტაციის მიზნით	მსჯელობისა და მისგან გამომდინარე დასკვნების ნათლად, ზუსტად და ადრესატისათვის მისაღები ფორმით მიწოდების უნარი, როგორც ზეპირად ისე წერილობით	დამოუკიდებლად მუშაობის უნარი	გუნდში მუშაობის უნარი
კონკრეტული მათემატიკა							
მონაცემთა სტრუქტურები							
ფუნდამენტური ალგორითმები							
დაპროგრამების ენა ასემბლერი და კომპიუტერის არქიტექტურა							
დაპროგრამების ენების შედარებითი ანალიზი							
პროგრამული უზრუნველყოფის ინჟინერია							
სირთულის თეორია							
სემინარი კომპიუტერულ მეცნიერებაში							
ფუნდამენტური ამოცანების სწრაფი ალგორითმები							
მონაცემთა ბაზები							
კომპიუტერული ტოპოლოგია							
ბიბლიოთეკა Windows API							
კომპიუტერული ალგებრის ალგორითმები							
.NET პლატფორმის თანამედროვე ტექნოლოგიები							
კომპიუტერული გეომეტრიის ალგორითმები							
პარალელური ალგორითმები							
სამაგისტრო ნაშრომი							

სამაგისტრო პროგრამა კომპიუტერული მეცნიერება

საგამოცდო საკითხები

ნაწილი1: დაპროგრამების ენა C++

1. ფუნქციები: პარამეტრები და არგუმენტები; არგუმენტების გადაცემის ხერხები (pass by value, pass by pointer, pass by reference); არგუმენტების ნაგულისხმევი (default) მნიშვნელობები; ჩადგმადი (**inline**) ფუნქციები; ფუნქციების გადატვირთვა; ფუნქციის ტემპლიტი (**template**); ფუნქტორები. შესაბამისი მაგალითები.
2. ინკაპსულაცია (Encapsulation): კლასის წევრებზე წვდომა (**private**, **public**, **protected**); ობიექტების კონტეინერები; კლასის კონსტრუქტორი, დესტრუქტორი, მეთოდები; კონსტრუქტორის გადატვირთვა; კონსტრუქტორის და დესტრუქტორის გამოძახების დრო და რიგი; კლასის სტატიკური და კონსტანტური ელემენტები: მონაცემები და ფუნქციები.
3. მეგობარი (**friend**) ფუნქცია, მისი სინტაქსი და სემანტიკა; კლასის წევრი ფუნქცია, კლასის მეგობარი ფუნქცია და გლობალური ფუნქცია - განსვავება მათ შორის; მეგობარი კლასები; მიმთითებელი **this**, მისი დანიშნულება და გამოყენება; კომპოზიცია: კლასის ობიექტები როგორც სხვა კლასის მონაცემები.
4. მეხსიერების დინამიკური მართვა: ოპერატორები **new** და **delete**; კლასის ობიექტისთვის დინამიკური მეხსიერების (heap) განაწილება; კლასის ობიექტების მასივის განთავსება დინამიკურ მეხსიერებაში; დინამიკური მეხსიერების გათავისუფლება. ობიექტების მისამართების კონტეინერები ობიექტების კონტეინერების ნაცვლად.
5. ოპერატორების გადატვირთვა: ოპერატორების გადატვირთვის ძირითადი პრინციპი. ფუნქცია-ოპერატორი, როგორც კლასის ფუნქცია და როგორც მეგობარი ფუნქცია. I/O ოპერატორების გადატვირთვა; ერთადგილიანი და ორადგილიანი ოპერატორების გადატვირთვა: ლოგიკური ოპერატორების გადატვირთვა, ინდექსის ოპერატორების გადატვირთვა და სხვა. კოპირების კონსტრუქტორი.
6. მემკვიდრეობითობა (Inheritance): საბაზო და წარმოებული კლასები; კლასის დაცული (**protected**) ელემენტები; მემკვიდრეობითობის სახეები: მარტივი პირდაპირი მემკვიდრეობითობა; ერთი საბაზო კლასის ორი ან მეტი მემკვიდრე; ორი ან მეტი საბაზო კლასის მემკვიდრე კლასი – საბაზო და მემკვიდრე კლასების კონსტრუქტორები; ვირტუალური საბაზო კლასი.
7. ვირტუალური ფუნქციები და პოლიმორფიზმი (Polymorphism): საბაზო კლასის ფუნქციების გადაფარვა (overriding) და დამალვა (hiding). საბაზო კლასის მეთოდების გამოძახება, სტატიკური და დინამიკური ბმა; ვირტუალური ფუნქციის ცნება, ვირტუალობის მექანიზმი; წმინდა (pure) ვირტუალური ფუნქციები; აბსტრაქტული საბაზო კლასები და კონკრეტული კლასები; ვირტუალური დესტრუქტორები.

ლიტერატურა:

1. Bjame Stroustrup, Programming Principles and Practice Using C++, 2nd Edition, Addison-Wesley, 2014.
2. Jesse Liberty, Siddhartha Rao, Bradley L. Jones. C++ in one hour a day. SAMS, 2008.
3. კობა გელაშვილი, ირინა ხუციშვილი. დაპროგრამების საფუძვლები.

<https://kobage.info/Articles/Reader/20>

4. ირინა ხუციშვილი. ობიექტზე ორიენტირებული დაპროგრამება C++-ზე. ლექციათა კურსი. პრაქტიკული და ლაბორატორიული მეცადინეობების მასალა. ელექტრონული სწავლების პორტალი <http://e-learning.tsu.ge/course/view.php?id=1518>
5. C++ How to Program, By P.J. Deitel & H.M. Deitel, Pearson, 10th Edition, 2016.
6. Scott Meyer, Effective Modern C++, O'Reilly, 2014.

ნაწილი 2: ალგორითმები

1. მონაცემთა დახარისხება (სორტირება): სწრაფი დახარისხება (quicksort), შერწყმით დახარისხება (mergesort), ჩასმით დახარისხება (insertionsort), გადათვლით დახარისხება (counting sort).
2. დინამიკური პროგრამირების ალგორითმები: უდიდესი საერთო ქვემიმდევრობის ძებნის ამოცანა, სამკუთხედის ამოცანა.
3. ხარბი ალგორითმები: ამოცანა განაცხადების შესახებ. ზურგჩანთის ამოცანა.
4. ალგორითმები გრაფებზე: სიგანეში ძებნა (BFS), სიღრმეში ძებნა (DFS), დიესტრას ალგორითმი, მინიმალური დამფარავი ხის აგება, პრიმის და კრასკალის ალგორითმები.
5. არითმეტიკული ალგორითმები: მიმატება, გამოკლება, გამრავლება. მათი პარალელური რეალიზაცია
6. ფუნქციათა ზრდის რიგი, მისი შეფასება O , Ω და Θ აღნიშვნით
7. ალგორითმის ზრდის რიგის შეფასება
8. პოლინომიური და ექსპონენციური ზრდის რიგის მქონე ალგორითმები და მათი შედარება

ლიტერატურა:

1. T.Cormen, C. Leiserson, R. Rivest, C. Stein. Introduction to Algorithms, Third Edition. The MIT Press, 2009.
2. Steven S. Skiena. The Algorithm Design Manual, Springer Verlag, 2010
3. Jon Kleinberg, Eva Tardos. Algorithm Design, Pearson, 2013
4. თსუ ელექტრონული ბიბლიოთეკა
<http://e-learning.tsu.ge/course/index.php?categoryid=75>
5. www.hinkali.com

ნაწილი 3: მონაცემთა სტრუქტურები

1. STL-ის კომპონენტები: კონტეინერები, კონტეინერთა ადაპტებები, ფუნქციური ობიექტები, იტერატორები.
2. ბინარული (ორობითი) გროვა: განმარტებები, ალგორითმები `std::make_heap`, `std::pop_heap`, `std::push_heap`, `std::sort_heap` და `HEAPIFY`, მათი აღწერა და ფსევდოკოდი.
3. ჰეპ-ფუნქციები. ღია მისამართებით ჰეპირება: ელემენტის ჩამატების, ძებნის და წაშლის ალგორითმები. ჯაჭვებით ჰეპირება: ძირითადი იდეა, STL ბიბლიოთეკის დაუხარისხებელი კონტეინერების სახეები. მათზე განაცხადის გაკეთება, ელემენტების ჩამატება, ძებნა, წაშლა.
4. ძებნის ორობითი ხე (BST) და ძებნის დალაგებული ხე (OST). ძირითადი ალგორითმები (დაბალანსებასთან დაკავშირებული საკითხების გარეშე). STL ბიბლიოთეკის

დახარისხებული კონტენტების სახეები. მათზე განაცხადის გაკეთება, ელემენტების ჩამატება, ძეგნა, წაშლა.

ლიტერატურა:

1. კობა გელაშვილი. მონაცემთა სტრუქტურები. <https://kobage.info/Articles/Reader/18>
2. D. Musser, G. Derge, A. Saini. STL Tutorial and Reference Guide, Second Edition, Addison-Wesley, 2006.
3. T.Cormen, C. Leiserson, R. Rivest, C. Stein. Introduction to Algorithms, Third Edition. The MIT Press, 2009.
4. Nicolai Josuttis . The C++ Standard Library - A Tutorial and Reference, 2nd Edition, Addison-Wesley Professional, 2012