

მონაცემთა სტრუქტურები, ალგორითმები და სისტემის დიზაინი

სტეფანე გურგენიძე

ზოგადი აღწერა

კურსი განკუთვნილია დეველოპერებისთვის, რომლებსაც სურთ, რომ რთული პროგრამული ამოცანების გადასაჭრელად გამოიმუშავონ ყველა საჭირო ცოდნა და უნარი. იგი შედგება შემდეგი ნაწილებისაგან:

- შესავალი Python-ში;
- მონაცემთა სტრუქტურები;
- ალგორითმები;
- ობიექტზე ორიენტირებული პროგრამირება (OOP);
- სისტემის დიზაინი;
- იმიტირებული გასაუბრებები.

ექვსი თვის განმავლობაში სტუდენტები სრულყოფილ ცოდნას მიიღებენ ამ საკითხებში. კურსი წარმართება Python-ის გამოყენებით და მიზნად ისახავს მონაწილეთა იმისთვის მომზადებას, რომ წარმატებით გაიარონ გასაუბრებები და დასაქმდნენ დიდ საერთაშორისო ტექნოლოგიურ კომპანიებში. კურსის განმავლობაში სტუდენტები ამოხსნიან ამოცანებს ისეთი პლატფორმების გამოყენებით, როგორებიცაა **LeetCode** და **Codeforces**, რათა გაიღრმავონ ცოდნა და გაიუმჯობესონ პრობლემების გადაჭრის უნარები.

პრერეკვიზიტები

პროგრამირების ნებისმიერი ენის კარგად ცოდნა.

კურსის ხანგრძლივობა

- **სრული ხანგრძლივობა:** 6 თვე
- **ლექციები:** 48
- **ლექციის ხანგრძლივობა:** 3 საათი
- **სიხშირე:** 2-ჯერ კვირაში

სასწავლო გეგმა

ნაწილი 1 - კურსის მიმოხილვა და შესავალი Python-ში

ლექცია 01: კურსის მიმოხილვა და Python-ის საფუძვლები - ნაწილი 1

ლექცია 02: Python-ის საფუძვლები - ნაწილი 2

ნაწილი 2 - მონაცემთა სტრუქტურები

ლექცია 03: დროითი და მესხიერებითი სირთულე

ლექცია 04: სტრინგი

ლექცია 05: სია

ლექცია 06: მატრიცა და სტეკი

ლექცია 07: ჰეშ ცხრილი და სეტი

ლექცია 08: ლექსიკონი, ჰეშ ცხრილის გადიდება (Resizing)

ლექცია 09: ბმული სია

ლექცია 10: ორმხრივად ბმული სია

ლექცია 11: რიგი

ლექცია 12: რეკურსია

ლექცია 13: ხე, ორობითი ხე

ლექცია 14: ორობითი ძეხვის ხე

ლექცია 15: გროვა

ლექცია 16: Trie

ლექცია 17: გრაფი

ნაწილი 3 - ალგორითმები

ლექცია 18: სორტირების ალგორითმები

ლექცია 19: ამოცანები სორტირებაზე და ხარბი ალგორითმები

ლექცია 20: პალინდრომები და ანაგრამები, რეკურსიული ამოცანები ხეში

ლექცია 21: რთული ამოცანები ხეზე

ლექცია 22: სიღრმესა და სიგანეში ძეხვის ალგორითმები (DFS & BFS)

ლექცია 23: ტერიტორიის შევსების ამოცანა

ლექცია 24: სიღრმესა და სიგანეში ძეხვის რთული ამოცანები

ლექცია 25: ამოცანების ამოხსნა გროვის გამოყენებით

ლექცია 26: საბსეტები, უკუსვლის მეთოდი (Backtracking)

ლექცია 27: პერმუტაციები

ლექცია 28: მცოცავი ფანჯარა

ლექცია 29: ორი მიმთითებლის მეთოდი

ლექცია 30: რთული ამოცანები სიებზე

ლექცია 31: ინტერვალები

ლექცია 32: პრეფიქს/სუფიქს ჯამები

ლექცია 33: დინამიკური პროგრამირება

ლექცია 34: Disjoint Set (Union-Find)

ლექცია 35: რთული ამოცანები გრაფზე

ნაწილი 4 - ობიექტზე ორიენტირებული პროგრამირება (OOP)

ლექცია 36: შესავალი OOP-ში

ლექცია 37: OOP-ს პრინციპები

ლექცია 38: ამოცანები OOP-ზე

ნაწილი 5 - სისტემის დიზაინი

ლექცია 39: სისტემის დიზაინის საფუძვლები, ჰორიზონტალური და ვერტიკალური სქემატიკა

ლექცია 40: რელაციური და არარელაციური მონაცემთა ბაზები

ლექცია 41: ქეშირება, CDN, გამძლეობა (Resilience)

ლექცია 42: შეტყობინებების რიგები და Pub/Sub სერვისები

ლექცია 43: სისტემის დიზაინის მაგალითები - ნაწილი 1

ლექცია 44: სისტემის დიზაინის მაგალითები - ნაწილი 2

ლექცია 45: სისტემის დიზაინი რეალურ ცხოვრებაში

ნაწილი 6 - იმიტირებული გასაუბრებები

ლექცია 46-48: იმიტირებული გასაუბრებები, რომლებიც მოიცავს კურსის განმავლობაში განვლილ თემებს: ამოცანების ამოხსნა, მონაცემთა სტრუქტურები, სისტემის დიზაინი და ობიექტზე ორიენტირებული პროგრამირება (OOP)

სწავლის შედეგები

კურსის გავლით მონაწილეები:

- შეიძენენ სიღრმისეულ ცოდნას მონაცემთა სტრუქტურებსა და ალგორითმებში;
- შეძლებენ რთული ამოცანების ამოხსნას სხვადასხვა ალგორითმების გამოყენებით;
- ეცოდინებათ სისტემის დიზაინის ძირითადი პრინციპები და შეძლებენ, შექმნან მასშტაბური და ეფექტური სისტემების დიზაინები;
- სიღრმისეულად ისწავლიან Python-ს და შეძლებენ მისი ფუნქციონალის გამოყენებას;
- მზად იქნებიან იმისთვის, რომ წარმატებით გაიარონ გასაუბრებები და დასაქმდნენ დიდ საერთაშორისო ტექნოლოგიურ კომპანიებში.

შეფასების სისტემა და მეთოდები

დავალებები: ყოველკვირეული ამოცანები, ხშირად ისეთი პლატფორმებიდან, როგორებიცაა LeetCode და Codeforces, რაც სტუდენტებს ლექციებზე გავლილი მასალების გამყარებაში დაეხმარებათ.

იმიტირებული გასაუბრებები: რა დროსაც სტუდენტები ლექტორთან პრიორისპირ, ინტერვიუს ფორმატში ამოხსნიან ამოცანებს და შექმნიან სისტემის დიზაინს.