



HRVATSKI SAVEZ
INFORMATIČARA



**21. KAMP MLADIH
INFORMATIČARA HRVATSKE**

BILTEN

S A D R Ź A J

21. kamp mladih informatičara Hrvatske.....	3
Svečano otvaranje	4
Pregled radionica, voditelja i pohvaljenih polaznika	5
 Razgovori o Kampu	
Voditelji radionica i predavači	6
 S priprema na Kampu	
Olimpijci	9
Juniori.....	12
Gost iz Makedonije	14
Davor Prugovečki i Adrian Satja Kurdija	15
 Popis polaznika 21. kampa	16
3 pitanja za polaznike s "domaćeg terena"	18
Završna svečanost 21. kampa	20
Intervju: Serdo Samblić, ravnatelj OŠ F. K. Frankopan	23
Natjecanje "Zagonetke jam"	24
Popis polaznika po radionicama	29
 RADIONICE	
Logo početni.....	37
Logo napredni	39
Logo za natjecatelje.....	42
BASIC početni	48
BASIC napredni	50
C za OŠ	51
C++/STL	54
PYTHON.....	59
Visual Basic.....	64
Izrada Android aplikacija	66
HTML/PHP	70
Algoritmi za OŠ	72
Algoritmi za SŠ	74
Olimpijci	77

21. KAMP MLADIH INFORMATIČARA HRVATSKE – KRK 2011.



21. kamp mladih informatičara Hrvatske, u organizaciji Hrvatskog saveza informatičara – HSIN-a, održan je **od 18. do 28. lipnja 2011.** godine. Program Kampa odvijao se pod pokroviteljstvom Ministarstva znanosti, obrazovanja i športa RH, uz financijsku potporu Hrvatske zajednice tehničke kulture, kao i uz potporu Grada Krka i T-HT-a.

Suorganizatori ove tradicionalne ljetne škole informatike bili su Osnovna škola Frana Krste Frankopana u Krku i Klub informatičara otoka Krka – KIOK.

21. kamp organiziran je u samo jednom terminu u trajanju od 11 dana. Program Kampa uspješno je završilo **66 učenika** osnovnih i srednjih škola RH. 21. kamp pohađalo je **42 učenika osnovnih škola**, od kojih je 14 pozvao HSIN temeljem natječaja, i **24 srednjoškolca**, od kojih je 16 pozvao HSIN.

Za polaznike Kampa održano je **14 radionica**, koje su vodili stručni suradnici HSIN-a i KIOK-a. Osnovne teme ovog programa bile su:

- ❖ **Programski jezici:** Logo, Basic, C, C++, Visual Basic i Python
- ❖ **Algoritmi:** početni, napredni, rješavanje složenih zadataka na algoritamskoj osnovi
- ❖ **Sistemske softver:** Android
- ❖ **Internet:** HTML / PHP

Cjelokupan stručni program održavao se u prostorima OŠ F. K. Frankopana u Krku. Program se odvijao u četiri učionice opremljene sa šezdesetak računala. Povezanost s internetom ostvarena je CARNet-ovom vezom od 10Mb/s.

Prema vlastitim interesima učenici su pohađali jednu ili više radionica. Svi su se na radionicama istakli marljivim radom. Veliko zanimanje pokazali su za radionicu *Python*, i odlično prihvatili radionicu *Izrada Android aplikacija*, novu u programu Kampa.

Posebna radionica *Olimpijci* organizirana je za pripreme informatičkih reprezentativca Republike Hrvatske (I. Katanić, G. Matula, I. Kičić i M. Milišić), izabranih temeljem rezultata na HIO 2011. i Izbornim pripremama 2011. Uz njih se, za međunarodna natjecanja, pripremao i hrvatski juniorski tim (D. Gleich, M. Balunović, I. Lazarić i M. Liskij).

Voditelji 21. kampa bili su **Boris Bolšec**, prof. informatike u OŠ F. K. Frankopan i **Aron Bohr**, stručni suradnik u HSIN-u. Glavni i odgovorni organizator ovog najpopularnijeg programa HSIN-a bio je **Ivo Šeparović**, dipl. ing., tajnik HSIN-a.



SVEČANO OTVORENJE

21. kamp mladih informatičara Hrvatske u organizaciji Hrvatskog saveza informatičara – HSIN-a, svečano je otvoren 20. lipnja 2011. godine u auli Osnovne škole Frana Krste Frankopana u Krku.

Srdačnu dobrodošlicu učenicima i predavačima uputio je organizator Kampa i tajnik HSIN-a, **Ivo Šeparović**. Posebno je pozdravio ravnatelja škole domaćina Serđa Samblića i profesora informatike Borisa Bolšeca, tehničkog voditelja Kampa i dopredsjednika HSIN-a, te im se zahvalio na osobnom doprinosu uspješnoj organizaciji Kampa 2011. Zatim se obratio učenicima, polaznicima Kampa s napomenom da se nitko od njih još nije ni rodio kada je HSIN započeo ovaj svoj program, davne 1991. godine u Rapcu. Pozdravio je i posebnog gosta iz Makedonije – Hristijana Bogoevskog, i rekao da se Kamp ponovno može smatrati međunarodnim.

“Velike se stvari događaju ovdje na Kampu, jer su prošle godine hrvatski olimpijci koji su se ovdje pripremali, osvojili 1. timsko mjesto na Srednjoeuropskoj informatičkoj olimpijadi u Slovačkoj, i to prvi puta u cijeloj povijesti nastupanja na CEOI-u, a nakon što su prijašnjih godina dva puta pobijedili pojedinačno. 2010. godine naši su olimpijci osvojili i 5. mjesto na međunarodnoj olimpijadi – IOI u Kanadi. Ipak, odgovorne za obrazovanje mladih u našoj državi ne zanimaju previše ovi naši programi, te nam usprkos sve boljim rezultatima svake godine smanjuju financijsku potporu.”, rekao je g. Šeparović.

Na kraju je pročitao pismo predsjednika Hrvatskog informatičkog zbora mr. sc. Marijana Frkovića, u kome pozdravlja sve sudionike, i želi uspjeh učenicima na Kampu i predstojećim međunarodnim natjecanjima u ovoj godini, te zamolio ravnatelja škole domaćina da se obrati prisutnima.



Serdo Samblić, ravnatelj Osnovne škole F. K. Frankopan je kazao:

“Poštovani sudionici Kampa, ja vam želim srdačnu dobrodošlicu! Želim vam da se ugodno osjećate u našem gradu i našoj školi, te da, osim što ćete ovdje dosta naučiti, da se i lijepo zabavite. Evo, imamo sreću s prognozom vremena, pa da se malo i odmorite.

Mi smo ovdje u školi ponosni što će naši bivši učenici imati čast raditi s vama, koji ste ono najbolje što u Hrvatskoj postoji, i zato smo mi i zainteresirani da ovaj Kamp i dalje bude ovdje kod nas. Ja se zahvaljujem gosp. Šeparoviću, svim predavačima i HSIN-u koji nam već godinama ustupaju domaćinstvo ovog Kampa.

Evo, želim vam još jedanput da se ovdje lijepo osjećate. Ako bilo koji problem uočite, slobodno nam se obratite, ako bilo što budete trebali, nazovite nas, mi ćemo vam od srca pomoći.

Želim vam ugodan boravak i proglašavam 21. kamp otvorenim!”

PREGLED RADIONICA, VODITELJA I POHVALJENIH POLAZNIKA

	Radionica	Voditelji i predavači	Broj polaznika	Pohvaljeni
1.	Algoritmi za OŠ	Nikola Dmitrović, prof.	13	Branimir Filipović, Ivana Žužić, Marija Gegić, Vedran Kurdija
2.	Algoritmi za SŠ	Adrian Satja Kurdija Davor Prugovečki, dipl. ing.	17	Domagoj Čavid, Hristijan Bogoevski, Jure Ratković, Neven Miculinić
3.	BASIC početni	Zrinka Gavran	9	Lucija Franolić, Luka Braut
4.	BASIC napredni	Marijan Jurešić, dipl. ing.	10	Karlo Sintić, Fran Vasilić, Lucija Žužić, Marina Samblić
5.	LOGO početni	Edi Topić	9	Lucija Franolić, Luka Braut
6.	LOGO napredni	Edi Topić	10	Fran Vasilić, Lucija Žužić, Marina Samblić
7.	LOGO za natjecatelje	Edi Topić	7	Marija Gegić, Vedran Kurdija
8.	PYTHON	Nikola Dmitrović, prof.	15	Karlo Kovačić, Matteo Kinkela, Tonko Sabolčec, Nikola Pintarić
9.	C za OŠ	Zrinka Gavran	6	Ivan Petrinić, Josip Klepec, Vilim Lendvaj
10.	C++ / STL	Jasmin Velkić, dipl. ing.	7	Ivana Žužić, Marin Drabić
11.	Visual BASIC	Marijan Jurešić, dipl. ing.	8	Carlo Cola, Dino Ilić, Lovro Mršić, Kristijan Vulinović
12.	HTML / PHP	Gordan Nekić	9	Ivana Žužić, Marko Lazarić, Nikola Pintarić, Kristijan Vulinović
13.	Izrada Android aplikacija	Jasmin Velkić, dipl. ing.	10	Nikola Pintarić, Kristijan Vulinović
14.	Olimpijci	Davor Prugovečki, dipl. ing. Adrian Satja Kurdija	8	-



RAZGOVORI O KAMPU...

...VODITELJI RADIONICA I PREDAVAČI

Nikola Dmitrović

- *Kratki komentar ovogodišnjeg Kampa?*

Meni je ovo peta godina za redom kako sudjelujem na Kampu te mogu reći da je i ovogodišnji Kamp organiziran na tradicionalno visokoj razini. Na Krku se dobro osjećamo i imamo sve preduvjete za dobar rad. Osobno sam radio s učenicima osnovnih škola na radionicama „Algoritmi“ i „Python“. Generacije osnovaca koje dolaze odlikuje visoka motiviranost i razina znanja koja je do prije samo nekoliko godina bila svojstvena srednjoškolcima.

- *Osobni doživljaj ovakvog oblika rada s natjecateljima?*

Definitivno je riječ o uspješnom modelu koji donosi rezultate. Svake godine se u sustavu natjecanja pojavi netko tko je prve programerske korake napravio upravo na HSIN-ovom Kampu. Zajednički rad i druženje s već ostvarenim natjecateljima je dragocjeno iskustvo koje motivira na dodatni rad ne samo učenike već i predavače i voditelje.

- *Slobodno vrijeme?*

Svakodnevno smo organizirano vodili osnovnoškolce na plažu, bilo je tu vremena i za igranje nogometa i kompjuterskih igrica, natjecanje u odgonetanju pitalica, zajedničko gledanje filmova na velikom platnu.

- *Budućnost Kampa?*

Definitivno trebamo raditi na tome da zadržimo ovakav način rada. Mi koji vodimo i organiziramo Kamp smo svjesni recesije koja nas pogađa, ali i drugi trebaju biti svjesni da je ovo jedan od načina kako odgajati mlade stručnjake koji bi nas trebali i mogli izvući iz krize.



Zrinka Gavran



- *Što misliš o ovom HSIN-ovom programu?*

Program je odličan – mislim da preko intenzivnog vježbanja na kampovima (i ljetnim i zimskim) natjecatelji nauče jako puno i motiviraju se dodatno za samostalan rad.

- *Kakvi su ovogodišnji polaznici?*

Na radionici "Basic početni" polaznici shvaćaju puno brže nego prošle godine – razlog tome je vjerojatno to što je grupa manja.

- *Kako ti se sviđa mjesto boravka?*

Krk kao mjesto mi se sviđa, jedino je smještaj malo predaleko od škole.

- *Imaš li slobodnog vremena i kako ga provodiš?*

Imam slobodnog vremena između radionica i tada idem na plažu ili čitam knjige.

- *Koje su današnje prednosti i nedostaci Kampa?*

Prednost je što bivši natjecatelji pripremaju sadašnje i time se razina znanja povećava svake godine, nedostatak je to što mislim da fali dodatna radionica za srednjoškolce ujutro.

- *Imaš li kakav prijedlog za poboljšanje izvedbe Kampa u budućnosti?*

Organizirati natjecanje na kraju Kampa i pobjednicima podijeliti pohvalnice.

Jasmin Velkić

- *Što misliš o ovom HSIN-ovom programu?*

Kamp je već tradicionalno mjesto gdje mi, stariji, imamo prilike prenositi svoja znanja mlađim generacijama, te je kao takav već dugi niz godina jedan od glavnih razloga velikih uspjeha naših olimpijaca, ali i poboljšanja razine algoritamskih znanja srednjoškolaca i osnovnoškolaca općenito. Definitivno smatram da ovu tradiciju treba nastaviti još dugi niz godina.

- *Kakvi su ovogodišnji polaznici?*

Polaznici s kojima sam ja imao prilike raditi najvećim dijelom pokazuju interes za obrađene teme, a također je primjetno da razina znanja iz godine u godinu raste, pa se tako danas s osnovnoškolcima rade stvari koje su prije svega nekoliko godina bile rezervirane za učenike srednjih škola.

- *Što misliš o mjestu boravka?*

Smještaj je vrlo dobar, hrana izvrsna, a Krk kao grad domaćin je vrlo lijep, uz poneki prigovor na nešto slabiji noćni život.

- *Imaš li slobodnog vremena i kako ga provodiš?*

Održavam dvije radionice dnevno, jednu u priepodnevnom terminu i jednu u večernjem, te mi ostaje dovoljno vremena za kupanje i ostale aktivnosti. Svakako smatram da je pohvalno što je organizator ostavio period od 14 do 18 sati za slobodne aktivnosti, kako bi svi mogli iskoristiti boravak na moru i za kupanje, a ne samo za učenje informatike.

- *Koje su današnje prednosti i nedostaci Kampa?*

Glavna prednost je marljivi rad s učenicima i prenošenje znanja starijih generacija, kako bi svaka sljedeća generacija bila bolja od prethodne. Trenutno se ne mogu sjetiti nekog konkretnog nedostatka, smatram da je sve vrlo dobro organizirano.

- *Tvoj prijedlog za poboljšanje?*

Manji udio piletine u obrocima ☹️. Pokušati motivirati polaznike za pohađanje radionica C#, Java ili OOP općenito. Smatram da bi te radionice mogle biti vrlo produktivne i izazovne za sve.



Marijan Jurešić



- *Što misliš o HSIN-ovom Kampu?*

Okupljanje najboljih informatičara na jednom mjestu s ciljem daljnjeg usavršavanja je odlična zamisao, koja se kroz dugi niz godina pokazala izuzetno uspješnom. Dodatna velika prednost programa je uključivanje novih učenika da, u radu s najboljim učenicima svoje generacije te siskusnim predavačima, nauče nova znanja te se lakše uključe u sustav natjecanja i smotri.

- *Kakvi su ovogodišnji polaznici?*

Ovogodišnji polaznici su bili izuzetno motivirani i očekujem da će se većina nastaviti baviti informatikom te sudjelovati na županijskim, ali i državnim natjecanjima.

- *Kako ti je organiziran dan?*

Nemam zamjerki na organizaciju Kampa, radno i veselo baš kako i treba biti ☺️.

- *Što su prednosti, a što nedostaci? Imaš li kakav prijedlog za poboljšanje programa?*

Prednosti Kampa informatičara su daljnje usavršavanje najboljih informatičara, ali i uvođenje mladih nada hrvatske informatike u svijet natjecanja i smotri. Nedostaci Kampa su što nije bilo moguće pozvati više učenika. Jedan od nedostataka je i spajanje termina za srednjoškolske uzraste s osnovnoškolskim uzrastima što znatno otežava izvođenje radionica na kojima sudjeluju oba uzrasta. Bilo bi odlično da se tijekom Kampa organizira jedno finalno natjecanje i smotra radova za mlade natjecatelje koji još nemaju puno iskustva u natjecanju.

Edi Topić

- *Što misliš o HSIN-ovom programu?*

HSIN-ov program je odlična prilika za upoznavanje mladih informatičara s programskim jezicima i metodama rješavanja zadataka.

- *Kakvi su ovogodišnji polaznici?*

Ovogodišnji polaznici su marljivi, brzi i poprilično zaigrani. Iako ta zaigranost ponekad narušava tok predavanja, ona su uvijek odrađena do kraja i u potpunosti.

- *Kako ti je organiziran dan na Kampu?*

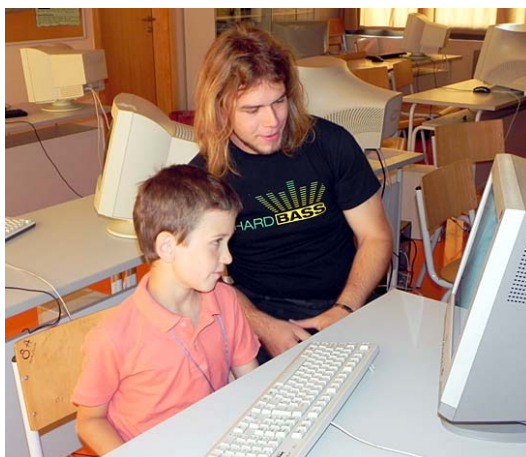
Pošto živim u Malinskoj na Krku, samo odradim tri predavanja ujutro i ručam na Kampu. Svaki dan se ustajem dosta rano jer putujem, ali raditi na Kampu je super.

- *Koje su današnje prednosti i nedostaci Kampa?*

Prednosti su mnogobrojne, a nedostatak je samo jedan i to tehničke prirode – internet.

- *Imaš li kakav prijedlog poboljšanja izvedbe Kampa za budućnost?*

Nemam.



Gordan Nekić



- *Što misliš o HSIN-ovom programu?*

HSIN-ov program izuzetno pridonosi da se novi učenici uključe u natjecanja, uz predavače koji su svi redom bivši najbolji sudionici natjecanja i smotri.

- *Kakvi su ovogodišnji polaznici?*

Polaznici su došli s osnovnim znanjem, ali su tijekom radionice dobro savladali sve zadatke.

- *Kako ti je organiziran dan na Kampu?*

Super, imamo vremena za sebe i dovoljno vremena za rad s polaznicima.

- *Koje su današnje prednosti i nedostaci Kampa?*

Prednosti kampa informatičara su usavršavanje najboljih informatičara i pripreme za natjecanja i smotre.

- *Imaš li kakav prijedlog poboljšanja izvedbe Kampa za budućnost?*

Bilo bi odlično da imamo odvojene termine za srednje i osnovne škole kao što je to bilo prošlih godina.

Razgovore vodila: Ivana Krajačić

S PRIPREMA NA KAMPU...

...OLIMPIJCI

Ivan Katanić

- *Super si startao na hrvatskim natjecanjima u 2011. godini (1. na Državnom i 1. na HONI-ju). Što nam pripremaš ovog ljeta na međunarodnom planu?*

Na CEOI-u i na IOI-u očekujem barem srebrne medalje iako svi od mene očekuju zlatne medalje, s obzirom da se već nekoliko godina natječem. Ipak, puno faktora utječe na natjecanju i nikad ne možeš biti siguran. Kao konkurente na CEOI-u vidim Poljake, koji kao domaćini imaju 2 tima, te su iznimno jaki. Konkurenti na IOI-u će nam biti Rusi, Kinezi i Amerikanci.



- *Kako su se odlični uspjesi odrazili na tvoje poimanje važnosti informatičkih natjecanja «u živo»?*

Natjecanja „u živo“ mi se jako sviđaju. Za razliku od TopCoda i sličnih natjecanja, ovdje su svi natjecatelji na jednom mjestu. Osjeti se pritisak i napeta atmosfera i to je ono što čini jedno natjecanje pravim natjecanjem.

- *Je li se (i kako) tvoj uspjeh odrazio i na tvoj privatni život?*

Nakon osvajanja medalja, svi pričaju o tome, ali nakon mjesec dana stvari se vraćaju u normalu. Osvajanje medalja mi je omogućilo direktan upis na FER, pa sam zadnju godinu srednje škole proveo bezbrižno.

- *S obzirom da većina informatičara odlazi raditi u Ameriku, misliš li i ti otići?*

Planirao sam otići jedno ljeto raditi u Facebook da vidim kako oni rade, pa ako mi se sviđa, možda ostanem raditi za stalno. Naravno, radije bih ostao u Hrvatskoj, ali sve ovisi o poslovnim prilikama.

- *Što misliš o pripremama na ovom Kampu?*

Mislim da su pripreme dobre. Svi natjecatelji su bili na nekoliko prijašnjih Kampova i prošli sve radionice, tako da nam sada preostaje samo da vježbamo zadatke za natjecanja.

- *Jesi li se stigao okupati u moru?*

Naravno. U slobodno vrijeme idemo na plažu ili se organizira gledanje filmova. Navečer se prošećemo malo do centra Krka.

Gustav Matula



- *Prvi puta si u olimpijskoj reprezentaciji RH. Reci nam ono najvažnije o svojim počecima i napredovanju u informatici?*

Sve je počelo upisom u srednju školu. Upisao sam se u gimnaziju MIOC i tada sam prvi put otišao na pripreme iz informatike jer je to bilo ono što me prvenstveno zanimalo u programu MIOC-a. Prve godine sam odustao jer su ostali bili napredniji od mene, a tada sam počeo sam raditi. Zadatke sam nalazio na web stranici HSIN-a. Svoje napredovanje vidim isključivo kao rezultat rada i vježbanja.

- *Koga (ili što) smatraš najvažnijim za svoju odluku da se posvetiš natjecanjima?*

Informatiku sam vidio kao izazov, pa me natjecateljski duh potaknuo na natjecanja. Uzore imam u bivšim natjecateljima Hrvatske. Jedan od njih je i Luka Kalinovčić koji je vrhunski u kodiranju.

- *Hoćeš li se sam dodatno pripremati za predstojeća natjecanja? Kakve rezultate očekuješ?*

Nakon završetka Kampa, planiram se još dodatno sam pripremati do CEOI-a i IOI-a. Tražit ću preko interneta zadatke sa stranih natjecanja i na taj način utvrditi sve naučeno. Zadatci, koje smo rješavali na Kampu, bit će nam od velike koristi. Što se tiče rezultata, ne mogu ništa prognozirati jer još uvijek imam oscilacija. Na CEOI-u najveći konkurenti će nam biti Poljaci, ali smatram da Ivan Katanić može osvojiti zlatnu medalju. Na IOI-u, kao glavnog favorita vidim Gennadyja Korotkevicha, bjeloruskog natjecatelja.

- *Imaš li prijedloge za poboljšanje Kampa?*

Jedan stabilan internet bi bio lijep. Moram pohvaliti hranu i samu organizaciju Kampa. Što se tiče radionice Algoritmi, koju sam i sam vodio, smatram da bi trebalo napraviti dvije radionice – početnu i naprednu, jer postoje ljudi s različitim razinama znanja u istoj grupi, pa je teže raditi.

Ivica Kičić



- *2010. godina je za tebe u informatičkom svijetu bila zaista uspješna. Kako si ti to sve doživio?*

Iskreno, nisam očekivao da ću osvojiti zlato na IOI-u u Kanadi, ali vjerovao sam da ću osvojiti barem srebrnu medalju. Osvajanje zlata bio je rezultat rada svih ovih godina.

- *Što ti je donijelo najviše radosti, a što (možda) loša iskustva?*

Dobar osjećaj je bio što sam se uopće kvalificirao na Hrvatsku informatičku olimpijadu. Naravno, osvajanjem zlata na 22. IOI-u u Kanadi bio sam iznimno sretan i zadovoljan. Ono što mi je posebno ostalo u sjećanju je Srednjoeuropska olimpijada u Slovačkoj. Tamo sam osvojio srebro. Svidjela mi se organizacija, a i sam grad Košice. Loših iskustava nisam imao.

- *Ove si godine član dvije olimpijske reprezentacije, informatičke i matematičke. Hoće li to imati pozitivan učinak na tvoj uspjeh?*

Uspjeh na matematičkoj olimpijadi vjerojatno neće biti kao na CEOI-u i IOI-u jer sam u informatici puno bolji nego u matematici. Informatika i matematika su slične tj. nadopunjuju se, pa mi je znanje iz matematike pomoglo u informatici i obrnuto. Smatram da je prednost ići na obje olimpijade, ali isto tako, manje je vremena za pripremanje.

- *Koji su ti planovi za budućnost?*

Ove godine sam završio srednju školu i upisat ću se na PMF, na Fiziku u Zagrebu. Za fiziku sam se odlučio jer mi djeluje zanimljivo i predstavlja izazov. Volio bih se baviti znanošću, što znači ili ostanak na fakultetu ili rad na Institutu Ruđer Bošković.

Matija Milišić

- *Prvi puta si u olimpijskoj reprezentaciji RH. Reci nam ono najvažnije o svojim počecima i napredovanju u informatici?*

Informatikom sam se počeo baviti u 1. razredu srednje škole. Tada sam došao na Državno natjecanje i odmah prisustvovao na HIO-u. Već slijedeće godine sam osvojio 1. mjesto na Državnom natjecanju, ali nisam prošao na Izborne pripreme. Ove godine sam prošao na Izborne pripreme i postao član olimpijskog tima.

- *Koga (ili što) smatraš najvažnijim za svoju odluku da se posvetiš računalnoj znanosti, i posebno natjecanjima?*

Moj prijatelj Antun Razum počeo se baviti informatikom u 7. razredu, te mi je on savjetovao da se počnem baviti informatikom. Počeo sam s pripremama na kampovima i u srednjoj školi. Za natjecanje me motivira želja da budem najbolji.

- *Izborio si mjesto i u olimpijskoj reprezentaciji iz matematike. Misliš li da je nastup na dvije svjetske olimpijade znanja u tako kratkom vremenu prednost ili opterećenje za natjecatelja?*

Gledam na to kao na prednost zato jer će to biti jedno veliko iskustvo za slijedeću godinu. Vesele me i putovanja u Nizozemsku, Poljsku i Tajland. Vidjet ćemo svijeta, a usput i osvojiti neke medalje.

- *Kako gledaš na odnos matematike i informatike?*

Smatram da mi matematika može pomoći, posebno kombinatorika koja je dosta povezana s informatikom. Isto tako, način razmišljanja na oba natjecanja je isti.

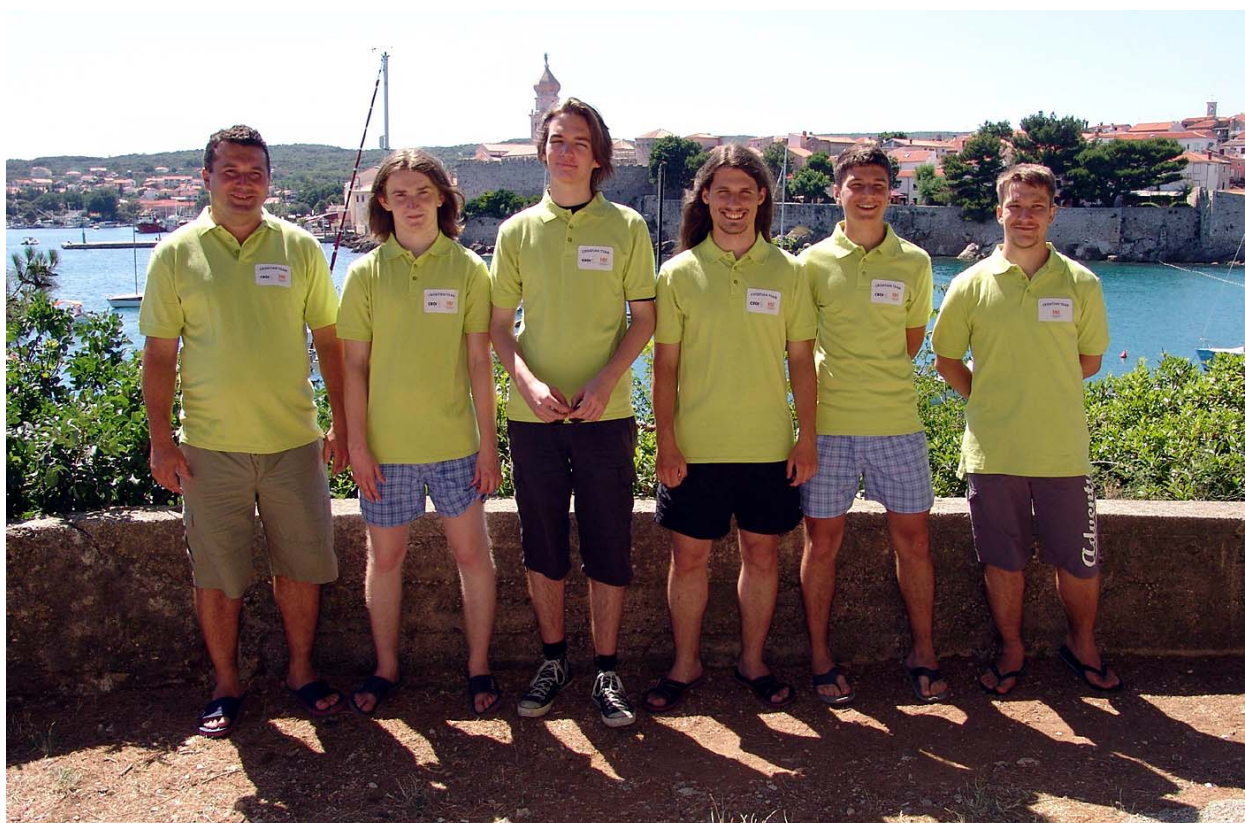
- *Kako si zadovoljan pripremama na Krku?*

Zadovoljan sam kao i svih godina do sada. Zadaci koje radimo su primjereni za moju razinu znanja i mislim da će me dovoljno dobro pripremiti za predstojeća natjecanja.

- *Tvoj prijedlog za poboljšanje Kampa?*

Volio bih kada bi internet bio stabilniji. S hranom sam zadovoljan, no što se tiče smještaja, očekivao sam da ćemo kao olimpijci biti u "predsjedničkom" apartmanu što je bio slučaj svih ovih godina.





Krešimir Malnar, Ivan Katanić, Gustav Matula, Ivica Kičić, Matija Milišić i Davor Prugovečki



Aron Bohr, Dominik Gleich, Mislav Balunović, Ivan Lazarić, Mihael Liskij i Adrian Satja Kurdija

... JUNIORI

Mislav Balunović



- *Prošle si godine opravdao procjene starijih o tvom neospornom talentu za informatiku. Kako su tvoj uspjeh primili u tvojoj školi i Slavanskom Brodu?*

U školi su se jako veselili mom uspjehu. Daju mi veliku potporu, iako trenutno nema radionica, pa većinom radim sam. Ukoliko bi trebala neka novčana potpora, uvijek su tu da pomognu. Ravnatelj, gospodin Džakula i profesori izlaze mi u susret i pomažu kada zbog informatike izostajem iz škole.

- *Je li se zbog tebe više učenika zainteresiralo za programiranje i općenito kvalitetniji rad s računalima?*

I prije je bilo ljudi zainteresiranih za informatiku, a sada ih je sve više i više. U školi postoji dodatna nastava informatike kojoj se sada mnogi priključuju.

- *Srednju školu si nastavio u Sl. Brodu. Misliš li da ćeš imati jednake (ili možda bolje) uvjete za svoj razvoj u računalnoj znanosti nego da si u Zagrebu?*

Možda bih imao bolje uvjete u Zagrebu, ali želio sam ostati u Slavanskom Brodu. Neki od razloga su obitelj i prijatelji.

- *Što očekuješ od nastupa u Rumunjskoj? Hoće li tvoj prošlogodišnji rezultat motivirati i ostale članove ekipe?*

Očekujem da će biti puno dobrih informatičara, ali isto tako očekujem da ću postići dobar plasman. Nadam se da će moj uspjeh motivirati i nove članove. Ukoliko im bude potrebna pomoć ili savjet, rado ću im pomoći.

Dominik Gleich

- *Ti već imaš iza sebe uspješan nastup na međunarodnom informatičkom natjecanju u Bugarskoj. Što očekuješ u Rumunjskoj?*

U Rumunjskoj očekujem barem medalju. Mislim da je Hrvatska vrlo jaka u odnosu na druge zemlje koje se natječu. Smatram da imamo realne šanse osvojiti dobra mjesta i medalje.

- *Učenik si 1. razreda XV. gimnazije u Zagrebu, koja je oduvijek bila rasadnik uspješnih mladih informatičara. Kako si zadovoljan nastavnim programom i profesorima iz informatike u školi?*

Zadovoljan sam nastavnim programom koji smo odradili ove školske godine. Pred kraj nastave, počeli smo raditi programiranje koje ćemo raditi do kraja srednjoškolskog obrazovanja. U školi, pod vodstvom prof. informatike Dmitrovića, imali smo pripreme na kojima smo marljivo radili. Informatički program je jako dobro organiziran jer je podijeljen na više grupa ovisno o razini znanja pojedinaca.

- *Da li (i koliko) surađuješ s bivšim olimpijcima iz tvoje škole? Tko ti je uzor?*

Informatičari su jako dobro povezani. Ako mi treba savjet oko nekog zadatka, najčešće pitam maturante koji su uvijek voljni pomoći.



Ivan Lazarić



- *Prvi puta si u mladoj reprezentaciji RH. Reci nam ono najvažnije o svojim počecima i napredovanju u informatici?*

Najviše mi je pomogao moj učitelj informatike Damir Blašković u Puli. Puno je radio sa mnom u C++-u i LOGU, pa sam upravo zbog toga imao velike uspjehe na natjecanjima. Moje prvo susretanje s informatikom bilo je u osnovnoj školi, gdje smo imali radionicu LOGA. Pošto sam bio dosta dobar u tome, učitelj je počeo raditi malo više sa mnom. S obzirom da sam išao na državno natjecanje, radili smo i C++, i to me dovelo do današnjih rezultata.

- *U ovoj si godini osvojio dva prva mjesta na Državnom natjecanju. Kako si se i s kim pripremao?*

Najviše sam se pripremao s prof. Blaškovićem, te isto tako na kampovima informatike u Krapini i na Krku. Isto tako pripremao sam se i pomoću zadataka koje bi našao na stranici HSIN-a, što mi je također, mnogo pomoglo.

- *Ima li u Puli još «zaljubljenika» u programiranje i ozbiljniji rad na kompjutorima?*

Osim mene, programiranjem se bavi i moj brat Marko koji radi u LOGU i C++-u, te još jedan srednjoškolac. Na žalost, nismo u kontaktu.

- *Veseliš li se međunarodnom natjecanju u Rumunjskoj i što od njega očekuješ?*

Jako se veselim natjecanju u Rumunjskoj i očekujem sve najbolje. Tamo se okupljaju najbolji natjecatelji te će pobjednik zapravo dokazati da je najbolji među najboljima. Od naših predstavnika, smatram da će Dominik Gleich imati najbolji rezultat.

- *Koji su ti planovi za daljnje školovanje?*

Ove godine sam završio osnovnu školu, te ću se upisati u prirodoslovno-matematičku gimnaziju u Puli, a nakon nje studirati na FER-u.

Mihael Liskij



- *Plasirao si se prvi puta u mladu reprezentaciju RH. Kada te je zainteresirao ozbiljniji rad na računalu i natjecanja u informatici?*

Kao dijete sam dosta vremena provodio igrajući igrice i to me zainteresiralo. Želio sam znati kako se rade, upoznati pozadinu toga, pa me to potaknulo na programiranje. U školi smo u 4. razredu dobili informatiku, te sam sa profesorom počeo raditi LOGO. Od 5.-8. razreda osvajao sam 1. mjesta na državnom natjecanju iz LOGA.

- *U ZRS-u si se sigurno susretao s najboljim mladim informatičarima Hrvatske, koliko su ti oni pomagali na putu do reprezentacije?*

Prije natjecanja u ZRS-u, mladi informatičari, imaju radionice koje vode bivši olimpijci. Isto tako, pripremam se i pomoću zadataka koje nalazim na stranicama HSIN-a. Tim stranicama se koriste i ostali programeri koje se može kontaktirati i upitati za savjet.

- *Kako si zadovoljan s pripremama na Kampu?*

Sviđaju mi se pripreme na Kampu. Zadaci su teški, ali nam predavač Adrian Satja Kurdija sve dobro objasni i ako trebamo pomoć, rado pomogne.

- *Imaš li dovoljno slobodnog vremena za muziku i svoje ostale aktivnosti?*

Pohađam srednju glazbenu školu i trenutno sam 2. razred. Sviram klavir i gitaru. Za sve svoje aktivnosti nađem vremena, bitno je samo dobro organizirati vrijeme.

- *Veseliš li se međunarodnom natjecanju u Rumunjskoj i što od njega očekuješ?*

Naravno da se veselim natjecanju. Ne želim si stvarati pritisak i očekivati medalje. Idem na natjecanje opušteno kao bi mi bilo lakše. Od naših predstavnika bih izdvojio Dominika Gleicha i Mislava Balunovića.

Razgovore vodila: Ivana Krajačić

...GOST IZ MAKEDONIJE

Hristijan Bogoevski

Uspjehe Hrvatske na međunarodnim informatičkim olimpijadama na svom je prvom nastupu uočio voditelj makedonskog tima i zamolio tajnika HSIN-a za pomoć. HSIN je Makedoncima nesebično ustupio svoja iskustva u svezi organiziranja saveza informatičkih udruga i programa rada s mladim, darovitim informatičarima. Suradnja Hrvatske i Makedonije nastavila se na BOI-u u Bitoli, a ovog se ljeta makedonski natjecatelj Hristijan Bogoevski pridružio pripremama naših olimpijaca na Kampu u Krku.



- Što te privuklo u svijet informatike? Opiši nam malo svoje napredovanje.

Sve je počelo igranjem igrica. Dolaskom u srednju školu, između svih ostalih predmeta, bila je i informatika. To me zainteresiralo i tada sam počeo programirati.

Dolaskom u srednju školu nisam znao ništa o programiranju. Tada nam je počeo predavati bivši olimpijac Makedonije i od njega sam puno naučio. Nakon godine dana rada, naučio sam dovoljno da idem na olimpijadu. U Makedoniji sam dva puta bio prvi na državnom natjecanju i to mi je najveći uspjeh do sada.

- Kako se razvija informatika u Makedoniji?

Makedonija nema toliko dobrih programera kao Hrvatska, pa iako idemo na natjecanja još se ne možemo pohvaliti velikim uspjesima. Osobe koje nas pripremaju za natjecanja, nemaju toliko znanja koje bi bilo potrebno za natjecanja. Bez obzira na to, Makedonija teži tome da jednog dana bude među najboljima u informatici. Država pomaže na način da oprema škole kompjuterima kako bi što više mladih zainteresirala za informatiku.

- Kako si saznao za ovaj Kamp?

U Makedoniji nema osoba koje bi me mogle naučiti ovome što sam naučio na ovom Kampu. Zahvaljujući gospodinu Jovanovu, koji je kontaktirao HSIN, došao sam ovdje da učim od najboljih.

- Kako se pripremaš za natjecanja, gdje nalaziš zadatke?

Zadatke nalazim na raznim internetskim stranicama, te se natječem na TopCoderu i sličnim natjecanjima. S obzirom da imam manje znanja od hrvatskih olimpijaca, dobivam zadatke koji su primjereniji mojoj razini znanja i iz kojih mogu nešto naučiti. Ukoliko zadatak ne mogu riješiti, predavači mi objasne.

- Kako ti se svidio Kamp i postoji li nešto slično u Makedoniji?

Jako mi se sviđa u Kampu. Svi su susretljivi i žele pomoći. U Makedoniji ne postoje ovakvi kampovi. Najbliže ovakvim kampovima su trodnevne škole u kojima se održavaju predavanja. Zahvaljujući ovom Kampu mnogo sam naučio.



Napomena urednice: Hristijan je rečeno na Kampu potvrdio na 23. IOI-u u Tajlandu osvojivši brončanu medalju za Makedoniju ☺.

Razgovor vodila: Ivana Krajačić

...VODITELJI RADIONICE

Davor Prugovečki



- *Kakva je tvoja ocjena ovogodišnjeg Kampa?*

Ovogodišnji program Kampa čini mi se dosta dobar, a uzrok tome je zasigurno i prilagodba organizatora željama polaznika. Tako postoji i posebna radionica "Android aplikacije" koja je zasigurno zanimljiva polaznicima. S druge strane, raspored održavanja radionica je vrlo dobro postavljen pa se polaznici mogu i dobro odmoriti.

- *Ima li među polaznicima novih mladih nada, odnosno budućih zlatnih olimpijaca?*

Trenutno pripremam ovogodišnju hrvatsku reprezentaciju za CEOI i IOI olimpijade, ali su zajedno s njima na pripremama i mlađi natjecatelji, oni do 16 godina. Tim koji će nas predstavljati na CEOI i IOI čini mi se jako dobar, i mislim da s pravom možemo očekivati odlične rezultate. Što se tiče mlađih, velik su potencijal i sigurno neće iznevjeriti na JBOI natjecanju.

Osim ove radionice, držao sam i radionicu naprednih algoritama za srednjoškolce i primijetio kako polaznici prvih i drugih razreda srednje škole imaju sve više znanja i uz još malo vježbe bit će odlični kandidati za olimpijski tim.

- *Misliš li da polaznici imaju dovoljno (ili previše) slobodnog vremena?*

Smatram da je slobodnog vremena taman, odnosno da je omjer radionica i slobodnog vremena optimalan s obzirom da je polaznicima tek završila školska godina. Također, mislim kako je predviđeno vrijeme za radionicu Algoritmi napredni bilo premalo, ali ne zbog količine tema koje su se namjeravale objasniti, već zbog polaznika koji su isključivo zbog te radionice došli na Kamp. Zbog toga su neki polaznici imali podosta slobodnog vremena. Način na koji smo smješteni ove godine je po meni jako dobar jer su polaznici grupno smješteni u objektima i bolje se upoznaju.

- *Imaš li kakav prijedlog poboljšanja izvedbe Kampa u budućnosti?*

Po meni je potrebna radionica poput olimpijske, ali s lakšim zadacima (recimo svakodnevno natjecanje ujutro u trajanju od 3h). To bi zasigurno privuklo polaznike kojima je cilj u budućnosti predstavljati Hrvatsku na olimpijadama. Također, ista bi možda mogla biti obavezna za sve one koji nemaju drugih radionica ujutro, a mogao bi se temeljem takve radionice proglasiti i pobjednik Kampa.

Adrian Satja Kurdija

- *Što misliš o HSIN-ovom programu na Kampu?*

Ovakve kampove HSIN održava već dvadeset godina i oni imaju golemu korist u pripremi hrvatskih informatičkih natjecatelja – od početnika do olimpijaca.

- *Kakvi su ovogodišnji polaznici?*

Vrlo sam zadovoljan! Olimpijci i mlađi natjecatelji svake su godine sve jači.

- *Što misliš o mjestu boravka?*

Krk je izvrsna lokacija, grad je vrlo lijep, a smještaj i uvjeti su odlični. To je dobra atmosfera za rad!

- *Ima li slobodnog vremena i kako ga provodiš?*

Slobodnog vremena je dovoljno. Provodim ga na plaži, uz film ili uz laptop tražeći dobre zadatke za olimpijce.

- *Koje su današnje prednosti i nedostaci Kampa?*

Prednost je svakako duga tradicija i bogato iskustvo organizatora i predavača. Nedostatak je, po mom mišljenju, što kamp za osnovne i srednje škole nisu u odvojenim terminima. Tako je bilo prije, pa je moglo biti pozvano više ljudi i organizirano više predavanja, no taj problem dolazi zbog ljudi na položajima koji ne prepoznaju vrijednost ovakvih manifestacija, pa su sredstva koja dobiva HSIN za pripreme i vodstvo hrvatskih mladih informatičara iz godine u godinu sve manja.



Razgovore vodila: Ivana Krajačić

21. KAMP – POPIS POLAZNIKA

UČENICI OSNOVNIH ŠKOLA

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Sven Ahčin	OŠ Strmec, Strmec	2.
2.	Josip Bajčić	OŠ F. K. Frankopana, Vrh	6
3.	Roberta Bajčić	OŠ F. K. Frankopana, Vrh	5.
4.	Luka Braut	OŠ F. K. Frankopana, Krk	4.
5.	Carlo Cola	OŠ F. K. Frankopana, Krk	8.
6.	Lucija Depikolozvane	OŠ F. K. Frankopana, Krk	4.
7.	Josip Đorđić	OŠ F. K. Frankopana, Baška	5.
8.	Branimir Filipović	OŠ Trnsko, Zagreb	8.
9.	Lucija Franolić	OŠ F. K. Frankopana, Vrh	5.
10.	Marija Gegić	OŠ Retkovec, Zagreb	7.
11.	Emilia Haramina	OŠ F. K. Frankopana, Krk	4.
12.	Filip Herceg	OŠ Brestje, Zagreb	7.
13.	Josip Klepec	I. OŠ Vrbovec, Vrbovec	5.
14.	Silvija Kosić	OŠ F. K. Frankopana, Vrh	6.
15.	Karlo Kovačić	Mladi informatičari Strahoninca, Strahoninec	6.
16.	Vedran Kurdija	OŠ Mladost, Zagreb	6.
17.	Ivan Lazarić	OŠ Centar, Pula	8.
18.	Marko Lazarić	OŠ Centar, Pula	6.
19.	Vilim Lendvaj	OŠ Popovača, Popovača	5.
20.	Mihael Liskij	OŠ P. Preradovića, Zagreb	8.
21.	Vedran Mihal	OŠ D. Tadijanovića, Slavonski Brod	8.
22.	Mario Mrakovčić	OŠ F. K. Frankopana, Punat	5.
23.	Lovro Mršić	OŠ F. K. Frankopana, Krk	5.
24.	Petar Orlić	OŠ J. Račića, Zagreb	7.
25.	Mario Pejinović	OŠ Cvjetno naselje, Zagreb	8.
26.	Niko Perić	OŠ F. K. Frankopana, Vrh	6.
27.	Ivan Petrinić	OŠ F. K. Frankopana, Baška	8.
28.	Nikola Pintarić	OŠ I. G. Kovačića, Sv. Juraj na Bregu	7.
29.	Luka Gabriel Rošić	OŠ F. K. Frankopana, Vrh	4.
30.	Tonko Sabolčec	OŠ I. G. Kovačića, Sv. Juraj na Bregu	8.
31.	Marina Samblić	OŠ F. K. Frankopana, Krk	5.
32.	Karlo Sintić	OŠ F. K. Frankopana, Krk	5.
33.	Leon Starešinić	OŠ J. Račića, Zagreb	7.
34.	Nikica Stijepić	OŠ F. K. Frankopana, Vrh	4.
35.	Lorenzo Šamanić	OŠ F. K. Frankopana, Krk	4.
36.	Ena Topić	OŠ F. K. Frankopana, Malinska	6.
37.	Luka Topić	OŠ F. K. Frankopana, Malinska	5.
38.	Marija Vančina	OŠ F. K. Frankopana, Krk	8.
39.	Fran Vasilić	OŠ F. K. Frankopana, Krk	5.
40.	Ivan Žmirić	OŠ F. K. Frankopana, Krk	7.
41.	Lucija Žužić	OŠ Fran Franković, Rijeka	5.
42.	Ivana Žužić	OŠ Fran Franković, Rijeka	8.

UČENICI SREDNJIH ŠKOLA

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Mislav Balunović	Gimnazija M. Mesića, Sl. Brod	1.
2.	Hristijan Bogoevski	Private Yahya Kemal College, Skopje, Makedonia	3.
3.	Domagoj Čevd	V. gimnazija, Zagreb	2.
4.	Marin Drabić	Mladi informatičari Strahoninca, Strahoninec	1.
5.	Goran Flegar	Prva gimnazija Varaždin, Varaždin	4.
6.	Dominik Gleich	XV. gimnazija, Zagreb	1.
7.	Martin Gluhak	Prva gimnazija Varaždin, Varaždin	2.
8.	Dino Ilić	SŠER, Rijeka	1.
9.	Ivan Katanić	Gimnazija Požega, Požega	4.
10.	Ivica Kičić	V. gimnazija, Zagreb	4.
11.	Matteo Kinkela	Gimnazija A. Mohorovičića, Rijeka	1.
12.	Gustav Matula	XV. gimnazija, Zagreb	4.
13.	Zvonimir Medić	XV. gimnazija, Zagreb	2.
14.	Neven Miculinić	Udruga darovitih informatičara Rijeke, Rijeka	2.
15.	Matija Milišić	XV. gimnazija, Zagreb	3.
16.	Ivan Paljak	Prva gimnazija Varaždin, Varaždin	3.
17.	Mihael Peklar	V. gimnazija, Zagreb	1.
18.	Drago Plečko	XV. gimnazija, Zagreb	1.
19.	Antun Razum	XV. gimnazija, Zagreb	3.
20.	Jure Ratković	Gimnazija L. Vranjanina, Zagreb	3.
21.	Ivan Sesar	XV. gimnazija, Zagreb	2.
22.	Marin Tomić	V. gimnazija, Zagreb	2.
23.	Tomislav Tunković	Gimnazija L. Vranjanina, Zagreb	3.
24.	Kristijan Vulinović	Gimnazija A. Mohorovičića, Rijeka	1.



3 PITANJA ZA POLAZNIKE S "DOMAĆEG TERENA"

Kao i proteklih godina na Kamp u Krku odazvalo se mnogo mladih s Krka i okolice, željnih novih informatičkih znanja. Nekolicinu smo zamolili za kratak razgovor, a to su:

Luka Braut, Emilia Haramina, Luka Gabriel Rošić, Nikica Stijepić, Lorenzo Šamanić – 4. razred OŠ F. K. Frankopana Krk i Vrh, Roberta Bajčić, Lucija Franolić, Marina Stamblić, Fran Vasilić – 5. razred OŠ F. K. Frankopana Krk i Vrh, Ena Topić – 6. razred OŠ F. K. Frankopana Malinska, Ivana Žužić – 8. razred OŠ Fran Franković Rijeka i Dino Ilić – 1. razred SŠ za elektrotehniku i računarstvo Rijeka.

• 1. Imaš li doma kompjuter i što voliš raditi na njemu?

Luka: Imam i volim ići na internet – najviše na Facebook. Volim igrati igrice "1, 2, 3" i ostale.

Emilia: Imam doma pet kompjutera i volim se na njima igrati – najviše igricu "Spore", ali volim i ići na internet na kojem idem na Facebook. Na Facebooku imam puno prijatelja.

Luka Gabriel: Imam doma kompjuter, na njemu najviše volim igrati online igrice. Volim ići na Facebook i pretraživati glazbu.

Nikica: Da, imam kompjuter i volim igrati igrice na internetu ili one koje sam instalirao.

Lorenzo: Imam, volim ići na internet, igrati igrice. Na internetu idem na Facebook, igram igrice, dopisujem se s prijateljima.

Roberta: Imam, volim igrati igrice, raditi prezentacije za školu i iz zabave i crtati u Paintu. Također volim ići na Facebook.

Lucija: Imam i volim biti na Facebooku i igrati igrice. Volim slušati glazbu, najviše Tonija Cetinskog i dopisivati se preko MSN-a.

Marina: Imam doma kompjuter. Najčešće sam na internetu, ponekad igram i igrice.

Fran: Nekad igram igrice, programiram – Logo, QBasic. Idem i na internet, na Facebook, dopisujem se s prijateljima.

Ena: Imam i najviše volim ići na internet i tamo najčešće igram igrice i volim istraživati Wikipediju.

Ivana: Imam kompjuter, volim programirati i igrati igrice. Također volim i gledati filmove na njemu.

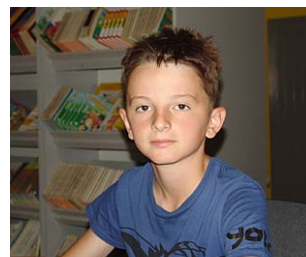
Dino: Imam kompjuter i obožavam raditi s kućnom mrežom, općenito volim raditi s mrežama.



Luka Braut



Emilia Haramina



Luka Gabriel Rošić



Lorenzo Šamanić



Roberta Bajčić



Nikica Stijepić

• **2. Kako ti se sviđa na Kampu?**

Luka: Hm... dobro je, dobro. Na Kampu mi je lijepo, igramo se i učimo, našao sam neke nove prijatelje. Rado bi došao i sljedeće godine.

Emilia: Super. Sviđa mi se jer se i igramo i učimo.

Luka Gabriel: Na Kampu mi se jako sviđa zato što sam upoznao nove prijatelje i naučio stvari koje prije nisam znao.

Nikica: Pa dobro je, sve u svemu je OK. Voditelj je super, ne kažnjava nas kao u školi pa nam je zabavnije.

Lorenzo: Fora je. Zezamo se, zabavljamo, učimo. Rado ću doći i sljedeće godine, ako me pozovu.

Roberta: Super je. Tu su mi prijatelji iz razreda i upoznala sam nove prijatelje.

Lucija: Super je zato što smo upoznali nove prijatelje i zajedno se družimo.

Marina: Sve mi se sviđa, ponekad nam na kraju daju da idemo na internet.

Fran: Super mi je na Kampu – idem na Logo, QBasic i Visual Basic. Radionice su ujutro pa sam popodne slobodan da se igram i družim s prijateljima.

Ena: Odlično je, ovdje sam već treću godinu i nadam se da ću doći sljedeće godine.

Ivana: Lijepo mi je na Kampu, puno naučim i dobra je podloga za buduća natjecanja. Najdraža radionica su mi Algoritmi za osnovne škole.

Dino: Super je. Smještaj i hrana mi se jako sviđaju. Radionice su jako, jako dobre i mislim da sam puno naučio.



Lucija Franolić



Marina Samblić

• **3. Jesu li profesori dobri?**

Luka: Hm... da. Idem na QBasic i Logo početni. QBasic mi je teži.

Emilia: Jako dobri.

Luka Gabriel: Da.

Nikica: Hm... da, dosta su dobri.

Lorenzo: Da.

Roberta: Da.

Lucija: Da.

Marina: Da, objasne nam što ne znamo, pomažu.

Fran: Da, super su.

Ena: Odlični su, kao i uvijek.

Ivana: Jesu, odlični su, sve objasne, čak ostanu i duže.

Dino: Profesori su izvrsni, pogotovo profesor Marijan iz Visual Basica jer sam kod njega najviše naučio zbog samostalnih projekata koje smo radili.



Ena Topić



Fran Vasilić

Razgovore vodila: Ivana Krajačić

ZAVRŠNA SVEČANOST

Nakon deset dana učenja i druženja, uz usvojena nova znanja i stečena nova poznanstva, 27. lipnja 2011. svečano je zatvoren 21. kamp mladih informatičara.



Na početku svečanosti domaćin nas je ugodno iznenadio kratkom priredbom učenica i učenika Osnovne škole.

Plesna grupa "Osjeti ritam" iz Pinezića pod vodstvom svoje trenerice Dajane Baković temperamentno je izvela odlično uvježbanu plesnu točku, a zatim se lijepim glasom i sugestivnom izvedbom sudionicima Kampa predstavila mlada pjevačica Stefany Žužić.

Voditeljica plesne udruge „Osjeti ritam“ Dajana Baković

Naša udruga je započela s radom sasvim slučajno. Prestankom rada u Krku, osnovali smo udrugu u Pinezićima jer je tamo bila najveća zainteresiranost. Općina nam je ustupila mjesni dom, a sve su pokrenule dvije mame, gđa. Milojka Mršić i gđa. Karolina Nazo. Udruga je aktivna već osam mjeseci.

Nastupali smo za dan škole u Vrhui, što je ujedno bio i prvi nastup udruge. Očito smo bili jako dobri kad nas zovu za daljnje nastupe.

Jako smo zadovoljni nastupom na završnoj svečanosti Kampa. Nas je inače u skupini 20, ali ovdje nas je nastupalo samo 7 i to: Josip Bajčić, Darinka Čutul, Ivona Mršić, Niko Perićić, Nina Vajdić, Stela Žužić i ja. Zadovoljni smo onim što smo pokazali.

Mlada pjevačica Stefany Žužić

Pjevanjem sam počela u ranom djetinjstvu. Sa 4 godine krenula sam na zbor „Mići boduli“ i njihova sam članica sve do današnjeg dana. Prošle godine sam krenula na satove solo pjevanja u HNK Ivana pl. Zajca u Rijeci. Što se nastupa tiče, to su ljetni nastupi. Zadnje dvije godine nastupila sam na natjecanju za mlade „Mladi otoka Krka pjevaju“ gdje sam sudjelovala kao gost. Isto tako nastupila sam i na Smotri glazbenog stvaralaštva djece i mladeži u Varaždinu.

S obzirom na tehničke probleme, ovim nastupom na Kampu jako sam zadovoljna. Pjesmu *Someone like you* od Adele odabrala sam za nastup jer naglašava moje pjevačke sposobnosti.



Usljedila je **dodjela Priznanja HSIN-a** svim sudionicima Kampa 2011., a podijelili su ih profesori **Boris Bolšec** i **Nikola Dmitrović**.



Tajnik HSIN-a, Ivo Šeparović predstavio je zatim članove reprezentacije RH i hrvatski juniorski tim, kao i voditelje, koji će nastupiti na međunarodnim informatičkim olimpijadama u 2011. godini. Pridružio im se Hristijan Bogoevski, član makedonske reprezentacije.



Pohvale HSIN-a najuspješnijim polaznicima dodijelili su voditelji radionica.





Na kraju prigodne svečanosti svima se zahvalio predsjednik HSIN-a, **Alen Spiegl**, te zajedno s Ivom Šeparovićem uručio **Zahvalnice HSIN-a** predavačima, organizatorima i pokroviteljima 21. kampa mladih informatičara.



21. kamp mladih informatičara službeno je zatvoren poticajnim govorom **Čedomira Milera**, zamjenika gradonačelnika Grada Krka.

INTERVJU:

SERĐO SAMBLIĆ, ravnatelj OŠ F. K. Frankopan

- *Kako ste zadovoljni Kampom ove godine?*

Prezadovoljan sam Kampom. Počasćen sam time da nam HSIN, već 7 godina, ukazuje čast da budemo domaćini Kampa mladih informatičara Hrvatske. Ovim putem bih se želio zahvaliti gospodinu Šeparoviću i svim članovima HSIN-a. Zadovoljni smo sa suradnjom i želja nam je da i dalje budemo domaćini Kampa. Nama je u interesu da djeca s Krka rade i uče od vrhunskih informatičara koje Hrvatska ima.



- *Planovi za budućnost?*

Što se tiče samoga Kampa, mi smo zainteresirani da se on i dalje održava ovdje u Krku.

Naravno, našim ambicijama tu nije kraj. Hrvatska će 2013. godine biti domaćin Srednjoeuropske informatičke olimpijade, te bi nam bila iznimna čast da se olimpijada održi u Krku. Imamo iskustva u organizaciji jer smo u posljednjih 10 godina šest puta bili domaćini državnim natjecanjima iz različitih predmeta. Ukoliko bi bili domaćini olimpijadi natjecanje bi bilo u hotelu Dražica ili ovdje u školi, ovisno o dogovoru.

- *Vaši prijedlozi kako poboljšati Kamp?*

Ideja nam je da Kamp poboljšamo na način da na vrijeme kandidiramo program, te dobijemo veću potporu županije i ministarstva. Tada bismo mogli organizirati nekoliko izleta po otoku Krku i izlete brodom krčkim akvatorijem za sve sudionike i profesore, te na taj način postati još bolji domaćini.

Razgovor vodila: Ivana Krajačić



NATJECANJE "ZAGONETKE JAM"

Polaznici Kampa **Kristijan Vulinović, Dino Ilić i Ivana Žužić** u slobodno vrijeme organizirali su natjecanje koje su nazvali "Zagonetke jam". Natjecanju su se odazvali skoro svi prisutni sudionici. Rješavali su se logički problemi, bez uporabe računala, što je svima dobro došlo kao mali predah od cjelodnevnog rada u radionicama.

Natjecanje "Zagonetke jam" održalo se u subotu 25. srpnja 2011. godine u 13 sati i 30 minuta u Osnovnoj školi Frana Krste Frankopana. Natjecanje je nadgledao izvršni odbor u sastavu: Kristijan Vulinović, Dino Ilić i Ivana Žužić u suradnji sa stručnim suradnikom za legitimnost rada izvršnog odbora, Aronom Bohrom.



Natjecanje je podijeljeno u četiri skupine. Svaka skupina se sastojala od određenog broja zagonetki, koje su se prikazivale na projektoru, a svaka skupina je bila različite teme. Tako smo imali kratke zagonetke, matematičke zagonetke, krimi-zagonetke i na kraju zagonetke koje su pripadale skupini "razno". Natjecatelji su bili grupirani u timove od tri sudionika. Za rješavanje zagonetki sudionici su imali određeno vrijeme te nakon što je jedan od sudionika točno riješio zagonetku, ostali bili upozoreni da imaju oko dvadesetak sekundi za dovršavanje istih. Postavljeno je 46 zagonetki te je ukupno bilo isto toliko mogućih bodova.



Prvo mjesto zauzeo je HSIN tim (u sastavu: Nikola Dmitrović, Jasmin Velkić i Aron Bohr) s ostvarenih 36 bodova. Slijede ih Leon Starešinić i Vedran Kurdija svaki s 30 bodova. Treće mjesto dijele Marija Gegić s 29 bodova kao i Mario Pejinović.

"ZAGONETKE JAM" – PITANJA I ODGOVORI

I.dio (kratke i jednostavne zagonetke, traži se kratak odgovor)

Što je to?

- | | |
|---|---|
| 1. Korice ima - nož nije, listove ima - drvo nije.
<i>Knjiga</i> | 8. Zakukuljeno, zamumuljeno, nitko ga ne može
otkukuljiti, odmumuljiti, samo onaj tko ga je zakukuljio,
zamumuljio!
<i>Brava i ključ</i> |
| 2. Jedna glava, a sto kapa.
<i>Glavica kupusa</i> | 9. Gospodar u kući stoji, a brada mu izvan kuće.
<i>Klip kukuruza</i> |
| 3. Koja se dva brata nikada ne sastanu, a samo ih brijeg
dijeli?
<i>Oči</i> | 10. Ja čuvam blago, a ljudi mene čuvaju
<i>Ključ</i> |
| 4. Ima 12 djece i 52 unučadi.
<i>Godina</i> | 11. Trči, trči trčuljak, visi, visi visuljak, Boga moli
trčuljak da otpadne visuljak?
<i>Svinja i žir</i> |
| 5. Gorom ide ne šuška, vodom ide, ne pljuska.
<i>Sjena</i> | 12. Izgovoriš mi ime i više me nema.
<i>Tišina</i> |
| 6. Cijeli dan ide, a nikud ne odmiče.
<i>Sat</i> | 13. Niti ti velim, niti ti kažem, samo ti se kaže.
<i>Niti</i> |
| 7. U gorici na jednoj nožici?
<i>Gljiva</i> | |

14. Bez kože uđe, s kožom izade.

Kruh

15. Od njega živiš, niti ga jedeš, niti ga piješ.

Zrak

16. Puna škola đaka, a niotkuda vrata?

Lubenica

17. Dva lokvanja oko panja?

Uši

18. Koji pojas ne možeš opasati?

Geografski pojas

19. Ima ih čovjek, stablo i knjiga.

List

II.dio (matematika)

1. Postoji jedno stablo i na dvije grane se nalaze dvije skupine ptica.

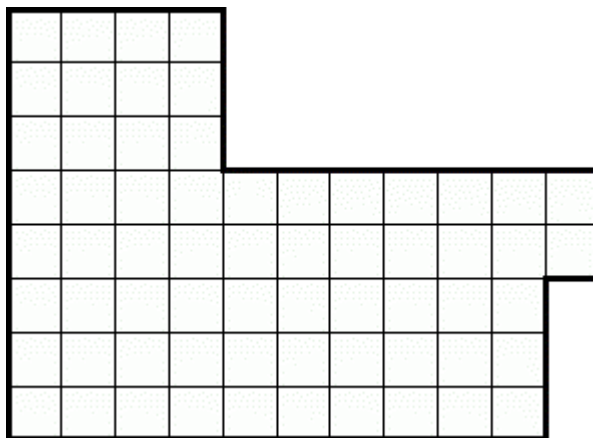
Ove s prve grane kažu ovima s druge: "Neka dođe jedna od vas kod nas na granu pa će nas biti isto."

Ove druge odgovaraju: "Neka jedna od vas dođe kod nas na granu pa će nas biti dvostruko više od vas!"

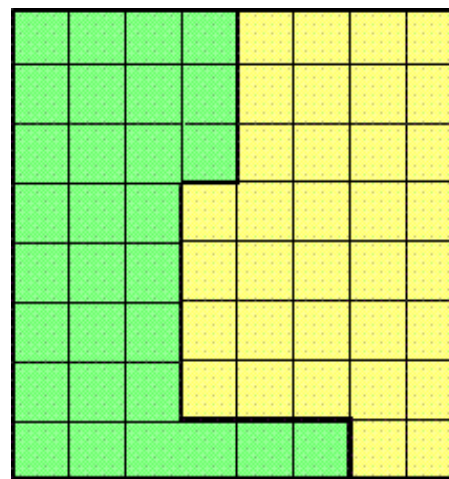
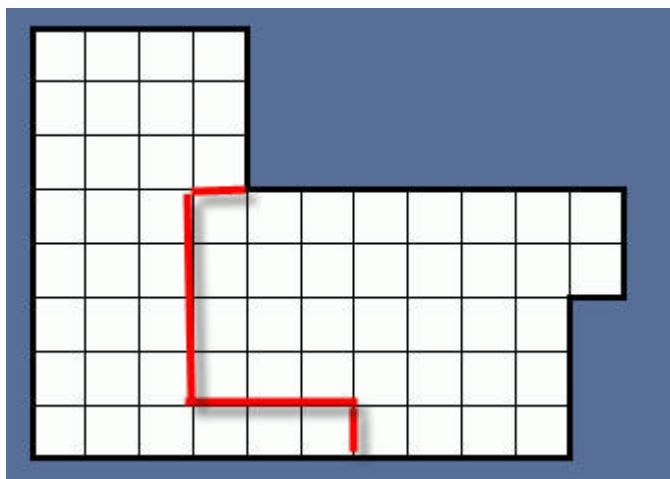
Koliko na kojoj grani ima ptica?

Na prvoj grani ima 5 ptica, a na drugoj 7.

2. Podijelite sliku na dva djela tako da čine 8×8 kvadrat (skicirajte)



Rješenje:



3. U autobusu se nalazi sedmorica dječaka. Svaki dječak ima sedam torbi. U svakoj torbi je sedam velikih mačaka. Svaka velika mačka ima 7 malih mačića. Svaka mačka ima 4 noge.

Koliko nogu se nalazi u autobusu?

10990

4. Nastavi niz...

1
11
21
1211
111221
312211
13112221

1113213211

5. $1+1+1+1+1+1+1+1+1+1-1+1+1+1+1+1 \times 0 = ???$

13

6. Ana i Marija imaju zajedno 44 godine. Marija je dva puta toliko stara koliko je Ana bila kada je Marija bila upola toliko stara koliko će Ana biti kada Ana bude tri puta toliko stara koliko je Marija bila kada je Marija bila tri puta toliko stara koliko je tada bila Ana. Koliko godina ima Ana, a koliko Marija?

a = Anine godine

m = Marijine godine

(1) $a + m = 44$

(2) $m = 2 * (a - x)$

(3) $m - x = 1/2(a + y)$

(4) $a + y = 3 * (m - z)$

(5) $m - z = 3 * (a - z)$

(2)-(5) $\Rightarrow 5a = 3m$

(1) $8a = 132 \Rightarrow$

Ana ima 16.5 godina, Marija ima 27.5 godina.

III. dio (Krimi-zagonetke)

1. Muškarac je upucan u autu. Na njegovoj odjeći nije bilo tragova baruta što znači da se ne radi o samoubojstvu i da je osoba koja ga je ubila bila izvan auta. Međutim, svi prozori su bili zatvoreni, a i vrata od auta su bila zaključana.

Nakon analize mjesta zločina, utvrđeno je da su jedine rupe od metka bile na čovjeku – dakle, na autu ništa, niti ogrebotine. Ključevi su pronađeni u autu. Kako je muškarac ubijen?

Auto je bio cabriolet te je ubijeni ubijen kroz otvoren krov.

2. Bio jednom okrugli dvorac. U njemu su bila tri kata. Na pravom katu vještica je kuhala napitak. Na drugom katu matematičar je učio u kutu. Na trećem katu je bio obješen čovjek. Nije počinio samoubojstvo ili tako nešto. Tko ga je ubio? Vještica ili matematičar? Zašto?

Ubojica je matematičar jer u okruglome dvorcu ne postoje kutovi.

3. U jednoj prostoriji su bili komadići razbijenog stakla i lokva vode. Ništa više. Kako su Ivica i Marica umrli u toj prostoriji?

Ivica i Marica su ribe koje su uginule jer se akvarij razbio.

-
4. U gradu usred pustinje dogodila se pljačka banke. Policija je krenula u istragu u pustinji. Prvo su pronašli vreće s pijeskom. Zatim razbacanu odjeću. Malo dalje putne kovčege. Još dublje u pustinji našli su gornju polovicu prelomljene šibice. Tada su pronašli mrtvu plavušu. Što se dogodilo?

Pljačkaši su pobjegli balonom. Kad ih je počela loviti policija, počeli su odbacivati teret kako bi mogli više poletjeti. Prvo su izbacili vreće, nakon toga odjeću, kovčege. Budući da su na kraju i dalje ostali preteški, odlučili su da osoba koja izvuče najkraću šibicu skače iz balona.

5. U sobi se nalazi ormar kojem su otpiljene noge. Na bočnoj strani ormara su kredom označene dvije crte. Na podu je prolivena voda i mrtvi patuljak zaposlen u cirkusu. Kako je umro?

Budući da je ormar bio snižen jer su mu noge bile odsječene, patuljak se ubio jer je mislio da je narastao, jer su crte označavale visinu patuljka.

IV. dio (razne)

1. "Ja Vam garantiram", kaže prodavač u pet shopu, "da će ova papiga ponoviti svaku riječ koju čuje." Kupac je donio papigu i shvatio da papiga ne ponavlja ni jednu riječ. Međutim prodavač nije lagao. Kako je to moguće?

Papiga je gluha.

2. Gđa. Bandek je kuhala u kuhinji kada je izgubila dijamantni prsten koji je upao u kavu. Prsten se nije smočio, zašto?

Kava je bila u prahu (nije kuhana).

3. Archibald je šetao 20 minuta po kiši, nije mu se smočila ni jedna vlas na glavi. Nije nosio šešir, nije imao kišobran i glava mu nije bila pokrivena. Odjeća mu se smočila, kako se to dogodilo?

Ćelav je.

4. Farmer Milan posjeduje 3 roze svinje, 4 smeđe svinje i jednu crnu svinju. Koliko Milanovih svinja može reći da su iste boje kao i druge svinje?

Svinje ne mogu govoriti

5. Tko bez jednog slova ostaje sam???

osam

6. Ima četiri slova, oduzmeš jedno, ostane pet.

Opet

7. Imam velika usta, glasan sam, ali nikoga ne ogovaram, iako imam posla sa svakakvim prljavštinama. Tko sam ja??!!

Usisivač

8. Što pripada tebi, a drugi to rabe više od tebe?

Tvoje ime

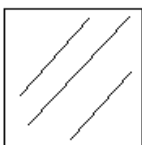
9. Servira se na stol, a nije ni za jesti ni za piti?

Ping pong loptica

10. Uđeš na jedna vrata a izađeš na dva vrata. Kada misliš da si izašao, zapravo si ušao....

Hlače

11. Nacrtaj kvadrat s tri crte.



12. Što je lakše od pera, ali se teško može držati?

Dah

13. Markova majka ima troje djece. Prvi se zove Ponedjeljak, drugi Utorak, a treći?

Marko

14. Što više od nje uzimaš sve je veća.

Rupa

15. Otac je imao dva sina, svaki sin imao je svog konja. Otac reče neka se utrkuju konjima i onaj čiji konj pobjedi neće dobiti ništa, a onaj čiji konj izgubi dobit će cijelo imanje. Naravno, stajali su na startu svako na svom konju i čekali sudnji dan jer niti jedan nije htio biti prvi. Nakon dugog čekanja otiđoše nekom seoskom pametnom čovjeku koji ih je savjetovao kako da obave utrku. Što im je savjetovao?

Da zamjene konje.

16. Postoje dvije sobe: u jednoj žarulja (ugašena), a u drugoj 3 prekidača. Iz prve sobe se ne može vidjeti druga soba i obrnuto. Kako saznati koji prekidač pali žarulju (istovremeno može biti pritisnut samo 1 prekidač i imaš pravo samo 1 put ući u sobu)?

Upalimo prvi prekidač i čekamo 15 min, ugasimo prvi prekidač i uključimo drugi. Odemo u sobu gdje je žarulja. Ako je žarulja upaljena, drugi prekidač pali žarulju. Ako je žarulja topla, a nije upaljena, pali ju prvi prekidač, ako nijedno od toga nije slučaj, onda žarulju pali treći prekidač.



Dino Ilić

POPIS POLAZNIKA PO RADIONICAMA

Olimpijci - Davor Prugovečki, dipl. ing., Adrian Satja Kurdija

R. br.	Ime i prezime	Škola	Raz.
1.	Ivan Lazarić	OŠ Centar, Pula	8.
2.	Mihael Liskij	OŠ P. Preradovića, Zagreb	8.
3.	Mislav Balunović	Gim. M. Mesića, Sl. Brod	1.
4.	Dominik Gleich	XV. gimnazija, Zagreb	1.
5.	Ivan Katanić	Gimnazija Požega, Požega	4.
6.	Ivica Kičić	V. gimnazija, Zagreb	4.
7.	Gustav Matula	XV. gimnazija, Zagreb	4.
8.	Matija Milišić	XV. gimnazija, Zagreb	3.

PYTHON – Nikola Dmitrović, prof.

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Marija Gegić	OŠ Retkovec, Zagreb	7.
2.	Petar Orlić	OŠ J. Račića, Zagreb	7.
3.	Mario Pejinović	OŠ Cvjetno naselje, Zagreb	8.
4.	Leon Starešinić	OŠ J. Račića, Zgreb	7.
5.	Branimir Filipović	OŠ Trnsko, Zagreb	8.
6.	Filip Herceg	Zagrebački računalni savez, Zagreb	7.
7.	Karlo Kovačić	Mladi informatičari Strahoninca, Strahoninec	6.
8.	Marin Drabić	Mladi informatičari Strahoninca, Strahoninec	1.
9.	Ivana Žužić	OŠ F. Frankovića, Rijeka	8.
10.	Dino Ilić	SŠER, Rijeka	1.
11.	Kristijan Vulinović	Gimnazija A. Mohorovičića, Rijeka	1.
12.	Marija Vančina	OŠ F. K. Frankopana, Krk	8.
13.	Tonko Sabolčec	OŠ I. G. Kovačić, Sv. Juraj na Bregu	8.
14.	Matteo Kinkela	Gim. A. Mohorovičića, Rijeka	1.
15.	Nikola Pintarić	OŠ I. G. Kovačića, Sv. Juraj na Bregu	7.

LOGO početni – Edi Topić

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Sven Ahčin	OŠ Strmec, Strmec	2.
2.	Emilia Haramina	OŠ F. K. Frankopana, Krk	4.
3.	Lucija Franolić	OŠ F. K. Frankopana, Vrh	5.
4.	Lucija Depikolozvane	OŠ F. K. Frankopana, Krk	4.
5.	Lorenzo Šamanić	OŠ F. K. Frankopana, Krk	4.
6.	Luka Braut	OŠ F. K. Frankopana, Krk	4.
7.	Roberta Bajčić	OŠ F. K. Frankopana, Vrh	5.
8.	Luka Gabriel Rošić	OŠ F. K. Frankopana, Vrh	4.
9.	Nikica Stijepić	OŠ F. K. Frankopana, Vrh	4.

LOGO napredni – Edi Topić

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Marina Samblić	OŠ F. K. Frankopana, Krk	5.
2.	Lovro Mršić	OŠ F. K. Frankopana, Krk	5.
3.	Fran Vasilić	OŠ F. K. Frankopana, Krk	5.
4.	Karlo Sintiћ	OŠ F. K. Frankopana, Krk	5.
5.	Josip Bajčić	OŠ F. K. Frankopana, Vrh	6.
6.	Niko Perićić	OŠ F. K. Frankopana, Vrh	6.
7.	Luka Topić	OŠ F. K. Frankopana, Malinska	5.
8.	Ena Topić	OŠ F. K. Frankopana, Malinska	6.
9.	Vilim Lendvaj	OŠ Popovača, Popovača	5.
10.	Lucija Žužić	OŠ Fran Franković, Rijeka	5.

LOGO za natjecatelje – Edi Topić

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Marija Gegić	OŠ Retkovec, Zagreb	7.
2.	Vedran Kurdija	OŠ Mladost, Zagreb	6.
3.	Petar Orlić	OŠ J. Račića, Zagreb	7.
4.	Mario Pejnović	OŠ Cvjetno naselje, Zagreb	8.
5.	Leon Starešinić	OŠ J. Račića, Zgreb	7.
6.	Vilim Lendvaj	OŠ Popovača, Popovača	5.
7.	Josip Klepec	I. OŠ Vrbovec, Vrbovec	5.

BASIC početni – Zrinka Gavran / Marijan Jurešić, dipl. ing.

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Roberta Bajčić	OŠ F. K. Frankopana, Krk	5.
2.	Emilia Haramina	OŠ F. K. Frankopana, Krk	4.
3.	Lucija Franolić	OŠ F. K. Frankopana, Vrh	5.
4.	Lucija Depikolozvane	OŠ F. K. Frankopana, Krk	4.
5.	Lorenzo Šamanić	OŠ F. K. Frankopana, Krk	4.
6.	Luka Braut	OŠ F. K. Frankopana, Krk	4.
7.	Josip Đorđić	OŠ F. K. Frankopana, Baška	5.
8.	Luka Gabriel Rošić	OŠ F. K. Frankopana, Vrh	4.
9.	Nikica Stijepić	OŠ F. K. Frankopana, Vrh	4.

BASIC napredni – Marijan Jurešić, dipl. ing.

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Marina Samblić	OŠ F. K. Frankopana, Krk	5.
2.	Lovro Mršić	OŠ F. K. Frankopana, Krk	5.
3.	Fran Vasilić	OŠ F. K. Frankopana, Krk	5.
4.	Karlo Sintić	OŠ F. K. Frankopana, Krk	5.
5.	Josip Bajčić	OŠ F. K. Frankopana, Vrh	6
6.	Silvija Kosić	OŠ F. K. Frankopana, Vrh	6
7.	Niko Peričić	OŠ F. K. Frankopana, Vrh	6
8.	Luka Topić	OŠ F. K. Frankopana, Malinska	5.
9.	Ena Topić	OŠ F. K. Frankopana, Malinska	6.
10.	Lucija Žužić	OŠ Fran Franković, Rijeka	5.

C za OŠ – Zrinka Gavran

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Petar Orlić	OŠ J. Račića, Zagreb	7.
2.	Mario Pejinović	OŠ Cvjetno naselje, Zagreb	8.
3.	Leon Starešinić	OŠ J. Račića, Zgreb	7.
4.	Josip Klepec	I. OŠ Vrbovec, Vrbovec	5.
5.	Ivan Petrinić	OŠ F. K. Frankopana, Baška	8.
6.	Vilim Lendvaj	OŠ Popovača, Popovača	5.

ALGORITMI OŠ – Nikola Dmitrović, prof.

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Branimir Filipović	OŠ Trnsko, Zagreb	8.
2.	Marija Gegić	OŠ Retkovec, Zagreb	7.
3.	Filip Herceg	Zagrebački računalni savez, Zagreb	7.
4.	Karlo Kovačić	Mladi informatičari Strahoninca, Strahoninec	6.
5.	Vedran Kurdija	OŠ Mladost, Zagreb	6.
6.	Petar Orlić	OŠ J. Račića, Zagreb	7.
7.	Mario Pejinović	OŠ Cvjetno naselje, Zagreb	8.
8.	Marin Drabić	Mladi informatičari Strahoninca, Strahoninec	1.
9.	Ivana Žužić	OŠ F. Frankovića, Rijeka	8.
10.	Leon Starešinić	OŠ J. Račića, Zgreb	7.
11.	Marko Lazarić	OŠ Centar, Pula	6.
12.	Josip Klepec	I. OŠ Vrbovec, Vrbovec	5.
13.	Nikola Pintarić	OŠ I. G. Kovačića, Sv. Juraj na Bregu	7.

VISUAL BASIC – Marijan Jurešić, dipl. ing.

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Marina Samblić	OŠ F. K. Frankopana, Krk	5.
2.	Lovro Mršić	OŠ F. K. Frankopana, Krk	5.
3.	Fran Vasilić	OŠ F. K. Frankopana, Krk	5.
4.	Karlo Sintić	OŠ F. K. Frankopana, Krk	5.
5.	Carlo Cola	OŠ F. K. Frankopana, Krk	8.
6.	Ivan Žmirić	OŠ F. K. Frankopana, Krk	7.
7.	Dino Ilić	SŠER, Rijeka	1.
8.	Kristijan Vulinović	Gimnazija A. Mohorovičića, Rijeka	1.

ALGORITMI SŠ – Adrian Satja Kurdija, Davor Prugovečki, dipl. ing.

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Mislav Balunović	Gimnazija M. Mesića, Sl. Brod	1.
2.	Hristijan Bogoevski	Private Yahya Kemal College, Skopje, Makedonia	3.
3.	Domagoj Čevid	V. gimnazija, Zagreb	2.
4.	Goran Flegar	Prva gimnazija Varaždin, Varaždin	4.
5.	Dominik Gleich	XV. gimnazija, Zagreb	1.
6.	Martin Gluhak	Prva gimnazija Varaždin, Varaždin	2.
7.	Matteo Kinkela	Gimnazija A. Mohorovičića, Rijeka	1.
8.	Zvonimir Medić	XV. gimnazija, Zagreb	2.
9.	Neven Miculinić	Udruga darovitih informatičara Rijeke, Rijeka	2.
10.	Ivan Paljak	Prva gimnazija Varaždin, Varaždin	3.
11.	Mihael Peklar	V. gimnazija, Zagreb	1.
12.	Drago Plečko	XV. gimnazija, Zagreb	1.
13.	Antun Razum	XV. gimnazija, Zagreb	3.
14.	Jure Ratković	Gimnazija L. Vranjanina, Zagreb	3.
15.	Ivan Sesar	XV. gimnazija, Zagreb	2.
16.	Marin Tomić	V. gimnazija, Zagreb	2.
17.	Tomislav Tunković	Gimnazija L. Vranjanina, Zagreb	3.

C++ / STL – Jasmin Velkić, ing.

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Branimir Filipović	OŠ Trnsko, Zagreb	8.
2.	Ivan Sesar	XV. gimnazija, Zagreb	2.
3.	Mihael Peklar	V. gimnazija, Zagreb	1.
4.	Karlo Kovačić	Mladi informatičari Strahoninca, Strahoninec	6.
5.	Marin Drabić	Mladi informatičari Strahoninca, Strahoninec	1.
6.	Ivana Žužić	OŠ F. Frankovića, Rijeka	8.
7.	Marko Lazarić	OŠ Centar, Pula	6.

PHP/HTML – Gordan Nekić

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Ivana Žužić	OŠ F. Frankovića, Rijeka	8.
2.	Marko Lazarić	OŠ Centar, Pula	6.
3.	Filip Herceg	OŠ Brestje, Zagreb	7.
4.	Carlo Cola	OŠ F. K. Frankopana, Krk	8.
5.	Ivan Žmirić	OŠ F. K. Frankopana, Krk	7.
6.	Dino Ilić	SŠER, Rijeka	1.
7.	Kristijan Vulinović	Gimnazija A. Mohorovičića, Rijeka	1.
8.	Karlo Kovačić	Mladi informatičari Strahoninca, Strahoninec	6.
9.	Nikola Pintarić	OŠ I. G. Kovačića, Sv. Juraj na Bregu	7.

Izrada ANDROID aplikacija – Jasmin Velkić, ing.

R. br.	Ime i prezime	Škola, Općina / Grad	Razred
1.	Ivana Žužić	OŠ F. Frankovića, Rijeka	8.
2.	Marko Lazarić	OŠ Centar, Pula	6.
3.	Vilim Lendvaj	OŠ Popovača, Popovača	5.
4.	Josip Klepec	I. OŠ Vrbovec, Vrbovec	5.
5.	Marin Drabić	Mladi informatičari Strahoninca, Strahoninec	1.
6.	Dino Ilić	SŠER, Rijeka	1.
7.	Kristijan Vulinović	Gimnazija A. Mohorovičića, Rijeka	1.
8.	Karlo Kovačić	Mladi informatičari Strahoninca, Strahoninec	6.
9.	Nikola Pintarić	OŠ I. G. Kovačića, Sv. Juraj na Bregu	7.
10.	Mihael Peklar	V. gimnazija, Zagreb	1.

R A D I O N I C E

OSNOVNA ŠKOLA "FRAN KRSTO FRANKOPAN"



Osnovna škola u Gradu Krku djeluje kao posebno područno razredno odjeljenje u sklopu OŠ "Fran Krsto Frankopan" Krk, najveće otočne škole u Primorsko-goranskoj županiji. Ona obuhvaća još šest osmogodišnjih i jednu četverogodišnju školu u mjestima: Omišalj, Malinska-Dubašnica, Vrh, Punat, Dobrinj, Vrbnik i Baška, sa zajedničkom upravom smještenom u krčkoj školi koja je ujedno i matična škola svim ostalima. Osnovna škola "Fran Krsto Frankopan" u Gradu Krku nalazi se na sjeveroistočnom rubu staroga grada omeđenog srednjovjekovnim gradskim bedemom.



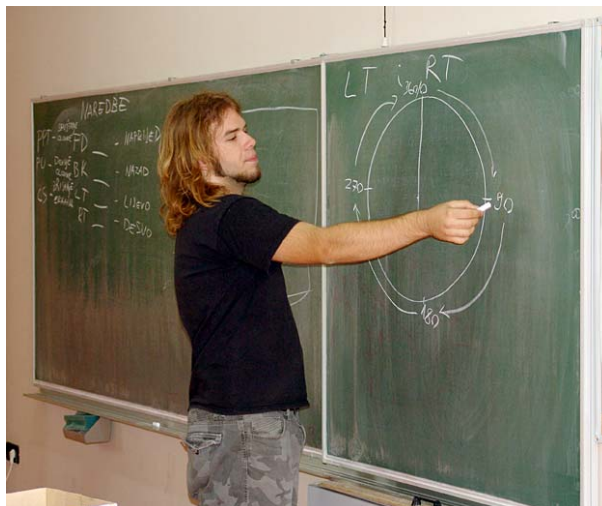
LOGO početni

Predavač: **Edi Topić**

Polaznici:

1. Sven Ahčin
2. Emilia Haramina
3. Lucija Franolić
4. Lucija Depikolozvane
5. Lorenzo Šamanić
6. Luka Braut
7. Roberta Bajčić
8. Luka Gabriel Rošić
9. Nikica Stijepić

Pohvaljeni: **Lucija Franolić i Luka Braut**



Radionicu LOGO početni pohađali su najmlađi nadareni informatičari, uglavnom s otoka Krka. Većina ih se nije susrela s LOGO programskim jezikom, pa se u ovih osam dana prešlo cjelokupno početničko pa i malo naprednog gradiva iz MSW LOGO-a.

Prva dva dana bila su posvećena osnovnim naredbama za kretanje, koje su svi polaznici veoma brzo usvojili. Zatim su naučili najvažniju naredbu, REPEAT, njezin brojač REPCOUNT, te njezine višestruke varijacije. Budući da ni tu nije bilo problema, polaznici su se uhvatili u koštac s procedurama, nepoznanicama i problemskim zadacima. Za kraj, da ne bi bilo sve crno-bijelo, naučili su raditi s bojama i animacijama.

Polaznici su bili veoma motivirani i uspješni, dijelom zbog pomalo natjecateljske atmosfere, a dijelom zbog predivnih slika, animacija i „trikova“ ovog programskog jezika. Što je najvažnije, naučili su postavljati i rješavati probleme natjecateljskog tipa.



Zadatak SUNCE :a :n

Potrebno je nacrtati sunce (mnogokut sa :n stranica veličine :a, kojemu su stranice zamijenjene s pravilnim mnogokutima).

Rješenje:

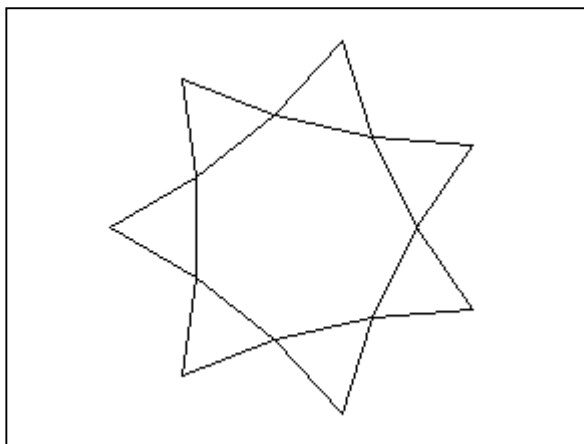
I. (pomoću procedure)

```
to SUNCE :a :n
  repeat :n[troukut :a fd :a rt 360/:n]
end

to TROKUT :a
  repeat 3[fd :a lt 120]
end
```

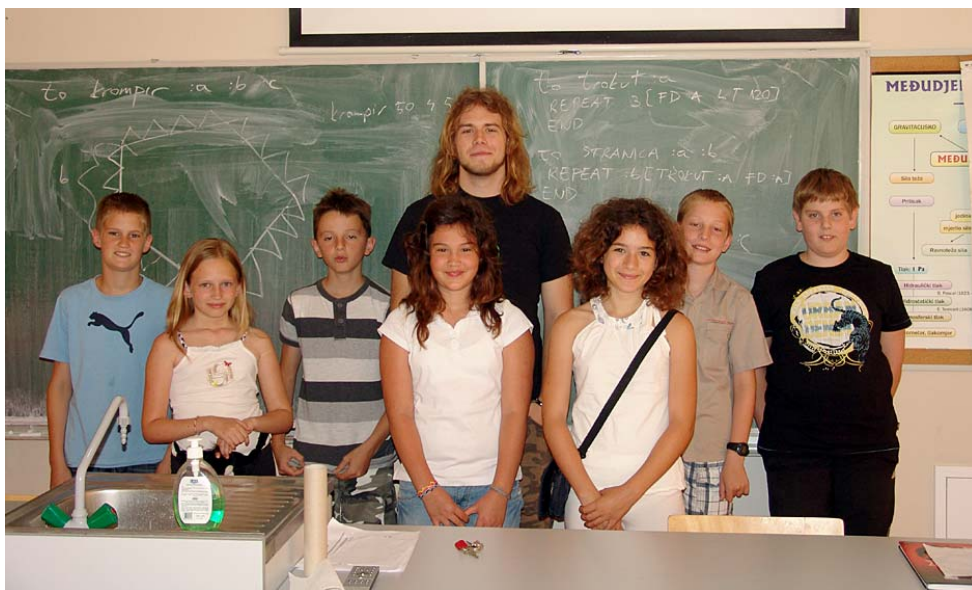
II. (pomoću dvostruke petlje)

```
to SUNCE :a :n
  repeat :n[ repeat 3[fd :a lt 120] fd :a rt 360/:n]
end
```



Primjer „fora“ slike:

```
cs repeat 360[repeat repeatcount[fd repeatcount rt repeatcount] rt repeatcount setpc
random 16 wait 1]
```

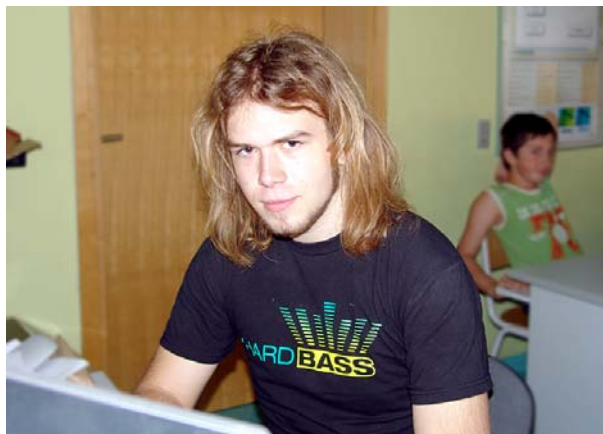


LOGO napredni

Predavač: **Edi Topić**

Polaznici:

1. Marina Samblić
2. Lovro Mršić
3. Fran Vasilić
4. Karlo Sintić
5. Josip Bajčić
6. Niko Peričić
7. Luka Topić
8. Ena Topić
9. Vilim Lendvaj
10. Lucija Žužić



Pohvaljeni: **Fran Vasilić, Lucija Žužić i Marina Samblić**

Radionica LOGO napredni okupila je sve osnovnoškolce koji su pokazali želju za višom razinom znanja iz programskog jezika LOGO. Na radionici su se obradili temeljni principi složenijih naredbi, struktura podataka te algoritmi u MSW LOGO-u.

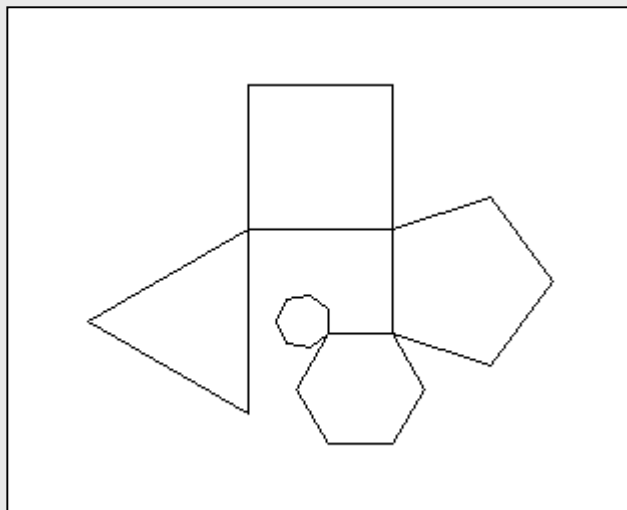
Radionica je bila podijeljena u tri dijela. Prvi dio se odnosio na kompliciranije primjene već naučenih naredbi, te ponavljanje i usavršavanje primjene naredbe MAKE. Drugi dio se odnosio na učenje novih struktura u LOGO-u, poput lista, uvjetnih naredbi, koordinatne grafike i analize. Treći dio je bila sinteza svog naučenog gradiva, u vidu pravljenja jednostavne igrice u MSW LOGO-u i završnog natjecanja.

Polaznici su uspješno savladali gradivo, što govori prosječna uspješnost završnog natjecanja od preko 70 posto! Također, polaznici su bili poprilično motivirani usvajajući nova znanja i primjenjujući to u nešto zabavno poput igrice.



ZADACI S NATJECANJA

1. Nacrtaj spiralu sa stranicama od n -terokuta (počevši od trokuta), tako da istovremeno smanjuješ stranicu $:a$ za $:b$ i povećavaš broj stranica za 1. Ako je stranica mnogokuta manja od $:s$, zaustavi proceduru. Procedura neka glasi SPIRALA $:A :B :S$



CS SPIRALA 92 20 10

Rješenje:

```
to spirala :a :b :s
  make "n 3
  while [:a>:s][repeat :n[fd :a lt 360/:n] fd :a rt 90
    make "a :a-:b make "n :n+1]
end
```

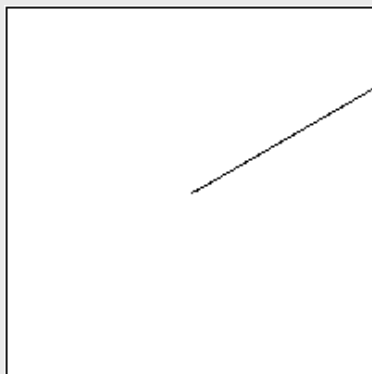
2. Napravi proceduru ISPIT $:a :b :x$, koja ispituje vrijednost nepoznanice $:x$, na sljedeći način:

- ako je $:x=1$, nacrtaj kvadrat veličine $:a$ sa središtem u središtu ekrana, i kružnicu veličine $:b$
- ako je $:x=2$, nacrtaj dva kruga, jedan veličine $:a$, drugi veličine $:b$
- ako je $:x=3$, nacrtaj pravokutnik širine $:a$ i visine $:b$ sa središtem u središtu ekrana

Rješenje:

```
to ispit :a :b :x
  if :x=1 [pu setxy -(:a/2) -(:a/2) ppt repeat 4[fd :a rt 90]
    pu home ppt circle :b]
  if :x=2 [circle :a circle :b]
  if :x=3 [pu setxy -(:a/2) -(:b/2) ppt repeat 2[fd :b rt 90 fd
    :a rt 90]]
end
```

3. Nacrtaj kvadrat veličine a sa središtem u središtu ekrana i nacrtaj zraku pod kutom k iz središta ekrana. Zraka ne smije izaći iz kvadrata. Procedura neka glasi ZRAKA $a :k$



CS ZRAKA 200 60

Rješenje:

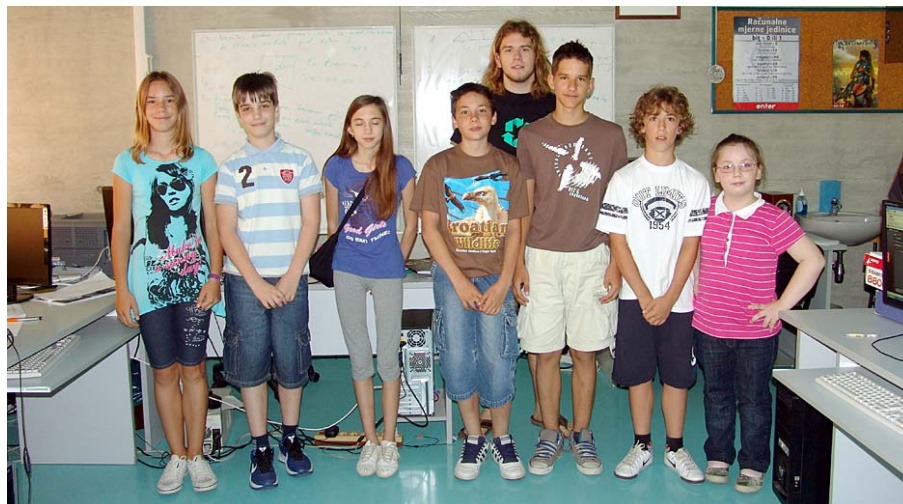
```
to zraka :a :k
  pu setxy -(:a/2) -(:a/2) ppt
  repeat 4[fd :a rt 90]
  pu home ppt rt :k
  repeat 1000[fd 1 if xcor>(:a/2) [stop] if xcor<-(:a/2) [stop]
    if ycor>(:a/2) [stop] if ycor<-(:a/2) [stop]]
end
```

4. Nacrtaj polje veličine $a \times b$ na način da je svaki piksel nasumično obojan drugačijom bojom (pomoću *random*). Prebroji crne piksele i ispiši koliko ih ima. Procedura neka glasi SARENO $a :b$.

(Opaska: ovaj zadatak se ne može direktno testirati, već je bitan samo kôd)

Rješenje:

```
to sareno :a :b
  repeat :a[repeat :b[setpc random 16 fd 1] pu bk :b rt 90 fd 1 lt 90 ppt]
  pu home ppt
  make "r 0
  pu
  repeat :a[repeat :b[if pixel=0 [make "r :r+ 1] fd 1]
    bk :b rt 90 fd 1 lt 90]
  print :r
end
```



LOGO za natjecatelje

Predavač: **Edi Topić**

Polaznici:

1. Marija Gegić
2. Vedran Kurdija
3. Petar Orlić
4. Mario Pejinović
5. Leon Starešinić
6. Vilim Lendvaj
7. Josip Klepec

Pohvaljeni: **Marija Gegić i Vedran Kurdija**



Ove godine LOGO natjecateljski je pohađalo sedmero vrhunskih LOGO natjecatelja, koji su pokazali nenadmašnu vještinu u rješavanju i najkompliciranijih zadataka. Iako su se neki zadaci rješavali i više od jednog dana, svi su uspješno savladani.

Radionica se koncentrirala na korelaciju s OŠ algoritmima, te primjenu istih na probleme tipične za natjecateljski LOGO.

Primjeri zadataka:

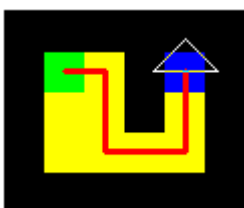
1. ZADATAK – Bajadero

LOGO kornjača se nedavno pretvorila u petu nindža kornjaču, „Bajadero“. Kako bi ušla u klan Tinejd Mjutnt Nindža Turtles, mora položiti ispit. Ispit je jako težak i ona traži Vašu pomoć pri rješavanju istog. Ispit se sastoji od labirinta (crno polje – zid, žuto polje – prazno, zeleno polje – start, plavo polje – cilj), a kornjača treba iscrtati svoj put do cilja crvenom bojom. Kornjača smije ići samo po praznom polju, startu i cilju. Ako put do cilja ne postoji, potrebno je ispisati „CILJ NE POSTOJI“.

Izgled labirinta sadržan je u listi :l, u obliku podlista koje predstavljaju retke labirinta. Simbolika za vrstu polja je sljedeća: 0 – prazno, 1 – prepreka, 2 – start, 3 – cilj.

Ukoliko postoji više rješenja, predložite bilo koje. Stranica polja neka bude 20 piksela. Kornjača se smije kretati samo ortogonalno, i to za 20 piksela gore, dolje, lijevo ili desno.

Primjer: CS BAJADERO [[1 1 1 1 1 1] [1 2 0 1 3 1] [1 0 0 1 0 1] [1 0 0 0 0 1] [1 1 1 1 1 1]]



Rješenje:

Problem se rješava implementacijom BFS algoritma. Kornjača se kreće rekursivno dok ne dođe do cilja ili ne prođe svakim putem. Procedure su: *bajadero* :l (glavna procedura), *boja* :a (funkcija koja vraća

boju s obzirom na unos), *crtaj :p* (funkcija koja crta polje i vraća dimenzije polja), *crtaj :r* (funkcija koja crta jedan red polja, a vraća veličinu polja), *rijesi :x :y :xp :yp* (rekurzivna BFS procedura).

```

to bajadero :l
  setpc 0
  setpensize [1 1]
  make "dim (crtaj :l) ;nacrtaj polje
  make "s (first :dim) ;deklariraj sve varijable
  make "v (last :dim)
  make "startx 0
  make "starty 0
  make "ciljx 0
  make "ciljy 0
  make "x 10
  make "y 10

  pu                                ;pronadi cilj i start
    repeat :v[
      repeat :s[
        setxy :x :y
        if pixel=2 [make "startx :x make "starty :y]
        if pixel=1 [make "ciljx :x make "ciljy :y]
        make "x :x+20]
      make "x 10 make "y :y+20]

  setxy :startx :starty                ;postavi se na cilj
  make "e (rijesi :ciljx :ciljy :startx :starty);pokreni rekurziju
  if :e=0 [print [CILJ NE POSTOJI]]    ;ako rekurzija vrati nulu, ispiši
  setpc 0
  setpensize [1 1]
end

to boja :a
  if :a=0 [op 6]                        ;žuta
  if :a=1 [op 0]                        ;crna
  if :a=2 [op 2]                        ;zelena
  if :a=3 [op 1]                        ;plava
end

to crtaj :p
  make "visina 0
  while [not :p=[]][make "sirina (crtajr (last :p)) make "visina :visina+1 pu
    fd 20 ppt make "p butlast :p] ;crtaj jedan redak i pomakni se gore
  op (list :sirina :visina)            ;vrati dimenzije
end

to crtajr :r
  make "h 0
  while [not :r=[]][make "h :h+1 setpc (boja (item 1 :r)) ppt
    repeat 4[fd 20 rt 90] pu rt 45 fd 10 ppt setfc (boja (item 1 :r))
    fill pu bk 10 rt 45 fd 20 lt 90 make "r butfirst :r ppt]
    ;crtaj redak, kvadrat po kvadrat
  rt 90 pu bk :h*20 lt 90 ppt          ;vrati se na početak reda
  op :h                                ;vrati visinu
end

to rijesi :x :y :xp :yp
  if (and xcor=:x ycor=:y)[op 1]        ;ako smo na cilju, lančano zaustavi rekurziju
  setpc 4                                ;crvena boja
  setpensize [3 3]
  pu

  setxy xcor+20 ycor                    ;proširi rekurziju udesno i vrati se
  ifelse (or pixel=6 pixel=1) [ppt setxy xcor-20 ycor setxy xcor+20 ycor pu
    if (rijesi :x :y xcor ycor)=1
    [op 1] setpc 6 ppt setxy :xp :yp pu setpc 4][setxy :xp :yp]
  setxy xcor-20 ycor                    ;proširi rekurziju ulijevo

```

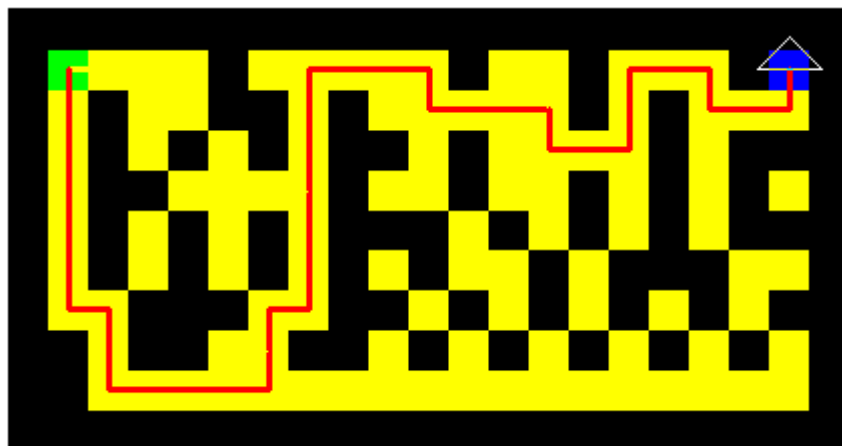


```

    ifelse (or pixel=6 pixel=1) [ppt setxy xcor+20 ycor setxy xcor-20 ycor pu
      if (rijesi :x :y xcor ycor)=1
        [op 1] setpc 6 ppt setxy :xp :yp pu setpc 4][setxy :xp :yp]
    setxy xcor ycor+20 ;gore
    ifelse (or pixel=6 pixel=1) [ppt setxy xcor ycor-20 setxy xcor ycor+20 pu
      if (rijesi :x :y xcor ycor)=1
        [op 1] setpc 6 ppt setxy :xp :yp pu setpc 4][setxy :xp :yp]
    setxy xcor ycor-20 ;dolje
    ifelse (or pixel=6 pixel=1) [ppt setxy xcor ycor+20 setxy xcor ycor-20 pu
      if (rijesi :x :y xcor ycor)=1
        [op 1] setpc 6 ppt setxy :xp :yp pu setpc 4][setxy :xp :yp]

    op 0 ;vrati nulu ukoliko se rekurzija maksimalno proširila bez rezultata
end

```



CS BAJADERO [[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1][1 2 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0 1 3 1][1 0 1 0 0 1 1 0
1 0 0 0 0 0 1 0 1 0 0 0 1][1 0 1 0 1 0 1 0 1 1 0 1 0 0 0 0 1 0 1 1 1][1 0 1 1 0 0 0 0 1 0 0 1 0 0 1 0 1 0 1 0 1][1 0 1 0 1
0 1 0 1 1 1 0 1 0 1 0 1 0 1 1 1][1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 1 1 0 0 1][1 0 0 1 1 1 0 0 1 1 0 1 0 1 0 1 0 1 0 1 1][1 1
0 1 1 0 0 1 1 0 1 0 1 0 1 0 1 0 1 0 1][1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1][1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1]]

2. ZADATAK – ALKENI

Alkeni su kemijski spojevi koji se uče na kraju osmog razreda. Sastoje se od ugljika (C) i vodika (H), i to tako da su atomi ugljika povezani jedan s drugim, a vodici su povezani s ugljikima.

Postoje mnoge vrste alkena, ali nas trenutačno, pošto smo jutros jeli margarin, zanimaju samo linearni (nerazgranati) alkeni. To su oni alkeni koji u svojoj strukturi nemaju ugljikov atom koji je povezan s više od dva susjedna ugljikova atoma.



Primjeri alkena

Struktura alkena prikazana je pomoću veznih crtica ili veza. To su crte koje spajaju dva prelamanja razlomljene crte (tj. ugljikove atome). U prvom i drugom primjeru veznih crtica je 8, a u trećem 4. Vezne crtice mogu biti jednostruke ili dvostruke (u drugim spojevima i višestruke), a u alkenima dvije dvostruke veze ne smiju biti jedna pored druge.

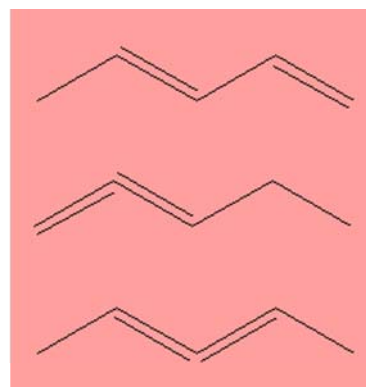
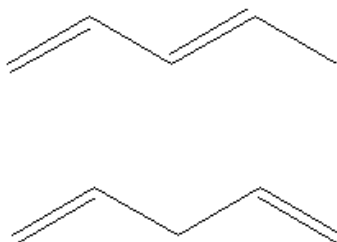
Vaš je zadatak nacrtati sve postojeće strukture jednog alkena s n prelamanja (kutova, ugljikovih atoma) i m dvostrukih veza. Strukture su iste ako je njihov poredak dvostrukih veza jednak ili obrnut. Potrebno je nacrtati samo jednu od dvije iste strukture.



Primjer istih struktura
(potrebno je nacrtati samo npr. gornju)

Vezna crta se crta kao linija duljine 40 piksela, a kut između dvije susjedne vezne crte je $120^\circ/-120^\circ$. n neće biti veći od 12, a m neće biti veći od 6.

Primjer: CS ALKENI 5 2



Pogrešne strukture (one koje se ne crtaju)

Rješenje nije komplicirano i zahtjeva samo malo više snalažljivosti i osnovno znanje o permutacijama. Ovdje se za permutacije koristi selektivni generator binarnih brojeva. Potrebno je pronaći sve zadovoljavajuće kombinacije jednostrukih i dvostrukih veza te ih nacrtati kako zadatak nalaže.



Procedure: *ALKENI* :n :m (glavna procedura), *BIN* :a :m (generator binarnog broja, koji usput provjerava zadovoljava li broj određena pravila), *LISTABIN* :brm :m :o („sakupljač“ zadovoljavajućih binarnih brojeva).

```

to alkeni :n :m
  make "x []
  make "n :n-1 ;broj veza
  make "r (listabin :n :m :n) ;napravi listu mogućih struktura
  make "brojac 1

  while [(count :r)+1>:brojac][
    make "pravi (item :brojac :r) ;provjeri ima li obrnutih struktura
    if (not (reverse :pravi)=:pravi)
      [make "r (remove (reverse :pravi) :r)] make "brojac :brojac+1]
  pu setxy 0 200 ppt ;postavi se na vidljivu poziciju

  while [not :r=[]][seth 60 make "p pos
    for [j 1 :n 1][make "h (item :j (first :r)) ;crtaj dvostruku ili jednostruku
    if :h=1 [lt 90 pu fd 5 ppt rt 90 fd 40 bk 40 rt 90 pu fd 5 ppt lt 90 fd 40]
    if :h=0 [ppt fd 40]
    ifelse (modulo :j 2)=0 [lt 60][rt 60]
    make "r (butfirst :r) pu setpos :p seth 0 bk 50 ppt]
end

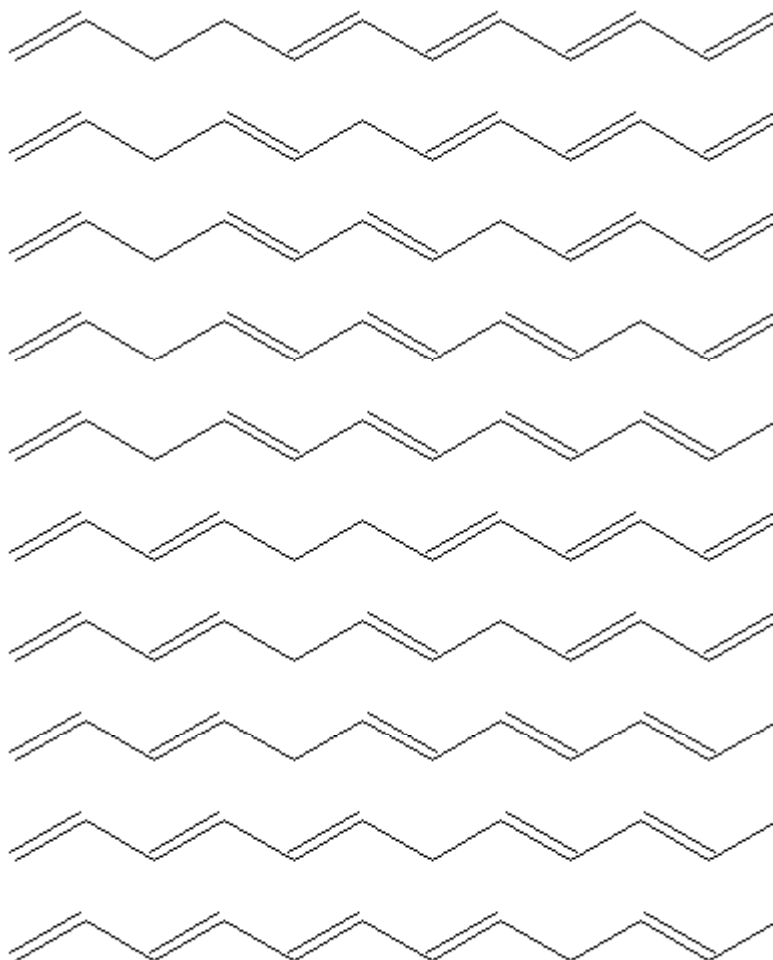
to bin :a :m
  make "x []
  make "br 0
  make "zadnja 0

  while [:a>0][
    make "z (modulo :a 2) ;provjeri: dvije susjedne dvostruke, previše dvostrukih
    if (and :zadnja=1 :z=1)[op []] make "zadnja :z make "br :br + :z
    ifelse (:br<(:m+1))[make "x (lput :z :x) make "a (:a-:z)/2]
      [op []] ifelse (:br=:m)[op reverse :x] [op []]
      ;ako nisu, vrati znamenke binarnog broja
  ]
end

to listabin :brm :m :o
  make "l []
  make "u (:brm-1)
  if (:u=0) [op []]
  make "poc power 2 (:brm-1)
    ;provjeri sve binarne
  make "kr power 2 :brm
    ;od 2:br-1 do 2:br
  make "temp []
  for [i :poc :kr-1 1]
    [make "temp (bin :i :m)
    if (not :temp=[])
      [while [(count :temp)<:o]
        [make "temp fput 0 :temp]
        make "l (lput :temp :l)] ]
  make "l (sentence :l
    (listabin :u :m :o))
    ;spoji sve binarne u jednu listu
  op :l
end

```





CS ALKENI 12 5

Ovo su bila dva teža zadatka s radionice, zahvaljujući kompleksnosti problema i težine implementacije rješenja u programski jezik LOGO. Unatoč tome, naši marljivi natjecatelji nisu posustajali i sa sigurnošću se može reći da ih ni jedan zadatak na natjecanjima neće smesti, te da će u srednjoj školi veoma lako nastaviti programerskim vodama.



Edi Topić

BASIC početni

Predavač: **Zrinka Gavran**

Polaznici:

1. Roberta Bajčić
2. Emilia Haramina
3. Lucija Franolić
4. Lucija Depikolozvane
5. Lorenzo Šamanić
6. Luka Braut
7. Josip Đorđić
8. Luka Gabriel Rošić
9. Nikica Stijepić

Pohvaljeni: **Lucija Franolić i Luka Braut**



Ove godine su na radionicu Basic početni došli svi učenici s istim predznanjem, tj. bez predznanja u programiranju u programskom jeziku QBasic. Cilj ove radionice bio je naučiti polaznike osnovama programiranja u programskom jeziku QBasic kako bi kasnije sami mogli nastaviti kôdirati u njemu i kako bi se zainteresirali za programiranje općenito.

Nakon uvodnog predavanja koje se temeljilo na tome da shvate kako treba kompjuteru "reći" da nešto napravi i pokazivanja početne sintakse počeli smo rješavati laganije zadatke. Grupa je napredovala podjednako brzinom što je olakšavalo predavanje.

Na početku radionice polaznici su učili kako se snalaziti na ekranu i općenito dobiti dojam o tome kako se izvodi program. Primjer takvog zadatka bi bio: "Napiši program koji u prvom redu ispisuje jednu riječ, u drugom dvije riječi, a u trećem redu jednu riječ."

Rješenje:

```
CLS
PRINT "zirafa"
PRINT "slon ";
PRINT "lav"
PRINT "tigar"
```

Kasnije su učili učitavanje brojeva i *if-else* provjeru. To su svi shvatili iz prve i moglo se brže preći na *for*-petlju.

Kroz radionicu su polaznici učili i sintaksu programskog jezika QBasic, ali i to da se zadaci mogu riješiti na drugačije načine, bolje načine. Primjer zadatka na kojemu su to mogli točno vidjeti je: „Napišite program koji učitava prirodan broj n i ispisuje sve neparne prirodne brojeve manje ili jednake od n .“

Prvo rješenje:

```
CLS
INPUT n
FOR i = 1 TO n
    IF i MOD 2 = 1 THEN
```

```

        PRINT i
    END IF
NEXT i

```

Drugo rješenje:

```

CLS
INPUT n
FOR i = 1 TO n STEP 2
    PRINT i
NEXT i

```

Treće rješenje:

```

CLS
INPUT n
FOR i = 1 TO n / 2
    PRINT i * 2 - 1
NEXT i

```



Razlika između prvog i drugog rješenja je u tome što je drugo rješenje duplo brže od prvoga – *for* petlja čini duplo manje koraka. Što se brzine tiče, između drugog i trećeg rješenja nema razlike, ali radili smo treće rješenje kako bi polaznici vidjeli da se zadatak može riješiti na različite načine, ovisno o tome što im se u tom trenutku čini kao bolja opcija.

Prva dva-tri dana bila su potrebna polaznicima da se naviknu na sintaksu QBasic-a – kada što treba napisati i na koji način, a kasnije su se već više mogli koncentrirati na rješavanja zadataka kroz koje su to onda i provježbali.

Prednost ovogodišnje grupe bio je njen broj – ove godine je u grupi bilo duplo manje polaznika i svi su imali jednako predznanje pa je grupa brže napredovala. To se lako može vidjeti po tome do kuda se došlo, ali i po broju zadataka koje je ova grupa riješila.

Na kraju radionice sam uputila polaznike na zadatke s bivših školskih, županijskih i državnih natjecanja kako bi što bolje utvrdili znanje stečeno na radionici. Najbolji polaznici su, za svoje zalaganje i izniman trud, dobili pohvalu.



Zrinka Gavran

BASIC napredni

Predavač: **Marijan Jurešić, dipl. ing.**

Polaznici:

1. Marina Samblić
2. Lovro Mršić
3. Fran Vasilić
4. Karlo Sintić
5. Josip Bajčić
6. Silvija Kosić
7. Niko Peričić
8. Luka Topić
9. Ena Topić
10. Lucija Žužić



Pohvaljeni: **Karlo Sintić, Fran Vasilić, Lucija Žužić i Marina Samblić**

Radionica BASIC napredni bila je namijenjena učenicima i natjecateljima koji su se već susreli s BASICom u svom radu. Razina znanja polaznika bila je vrlo heterogena, pa je pristup bio više individualan.

Na samom početku provedena je kratka anketa koliko učenici znaju o BASICu. Uspostavilo se da jedan dio njih ne zna niti naredbe grananja niti petlje, što je zahtijevalo da radionica započne s IF naredbom. Budući da je ova naredba izuzetno važna za bilo kakav nastavak programiranja, posvećena su joj 2 predavanja. Tijekom ta dva predavanja rješavani su različiti zadaci u kojima je težište stavljeno na grananje, od onih vrlo jednostavnih do zadataka težine županijskog natjecanja.

Sljedeća obrađivana tema bile su petlje (*for*, *do-while*). Petlje su isto tako esencijalne za programiranje, no i nešto kompliciranije od naredbi grananja te im je posvećeno ukupno 3 predavanja.

Nakon petlji uslijedila su polja (jednodimenzionalna i dvodimenzionalna) te svi standardni algoritmi s poljima: unos, ispis, pretraživanje, brisanje elementa, dodavanje elementa, spajanje dvaju sortiranih polja (*merge*), sortiranje te na kraju nekoliko zadataka s natjecanja kod kojih je težište stavljeno na polja. Na polja je utrošeno 2 predavanja.



Nakon polja bili su obrađeni stringovi, standardne BASICove funkcije za rad sa stringovima (*LEN*, *LEFT\$*, *RIGHT\$*, *MID\$*,...). Obrađeni su standardni algoritmi, okretanje stringa, palindrom,... za ovu temu potrošeno je jedno predavanje.

Na kraju, zadnje je predavanje bilo posvećeno potprogramima. Kroz nekoliko različitih primjera učenici su upoznati s osnovnim svojstvima potprograma, prednostima upotreba potprograma i slično. Učenici nisu imali završnu provjeru znanja upravo zbog velike heterogenosti grupe, ali su u globalu svi savladali predavano gradivo.

C za osnovne škole

Predavač: **Zrinka Gavran**

Polaznici:

1. Petar Orlić
2. Mario Pejinović
3. Leon Starešinić
4. Josip Klepec
5. Ivan Petrinić
6. Vilim Lendvaj

Pohvaljeni: **Ivan Petrinić, Josip Klepec i Vilim Lendvaj**



Pola polaznika ove radionice su bili totalni početnici u C++-u, a pola njih je već dugo kôdiralo u C++-u tako da su znali cijelu sintaksu. Zato se radionica podijelila na dva dijela – one koji uče C++ ispočetka i one koji rješavaju zadatke u C++-u.

Cilj ove radionice bio je naučiti polaznike osnovama programiranja u programskom jeziku C++ kako bi kasnije, sami doma mogli nastaviti kôdirati u njemu.

U početnoj grupi su bili učenici koji su već prije kôdirali u nekim drugim programskim jezicima (QBasic ili Pascal) tako da su razumjeli osnove samo su ih trebali ponoviti u C++-u.

Kao i uvijek sa polaznicima koji su tek počeli kôdirati bilo je problema sa shvaćanjem kako funkcionira *for* petlja i dvije ugniježdene *for* petlje. Za bolje razumijevanje toga su rješavali zadatak pomoću kojega su trebali učitati prirodan broj n i ispisati piramidu brojeva – u prvom redu broj 1, u drugom redu brojeve 1 i 2, ..., u n -tom redu brojeve 1, 2, ..., n .

Rješenje:

```
#include <cstdio>
using namespace std;

int n;

int main(){
    scanf("%d", &n);
    for (int j = 0; j < n; ++j)
    {
        for (int k = 0; k < j; ++k)
            printf("%d ", k + 1);
        printf("\n");
    }
    return 0;
}
```


Nakon što su svladali *for* petlju prešli smo na nizove. Oni koji su se već susreli s nizovima samo su se trebali naviknuti da u C++-u indeksi u nizu kreću od 0 umjesto do 1, kako su do sada radili. Oni kojima je niz bio novi pojam, najveći problem je bio shvatiti razliku između indeksa polja u nizu i vrijednosti polja u nizu. Razlika se može lako uočiti u sljedećem zadatku: Učitaj brojeve n i a i nakon njih niz od n brojeva koji će se sastojati od brojeva 0 do $n-1$, svaki broj će se pojaviti samo jednom. Potrebno je ispisati koji broj se nalazi na a -tom mjestu u nizu i na kojem mjestu u nizu se nalazi broj a .

Rješenje:

```
#include <cstdio>
using namespace std;

int main()
{
    scanf("%d%d", &n, &a);
    for (int j = 0; j < n; ++j)
        scanf("%d", &niz[j]);
    printf("%d\n", niz[a]); // ispisuje broj na a-tom mjestu u nizu
    for (int j = 0; j < n; ++j)
        if (niz[j] == a)
        {
            printf("%d\n", j); // ispisuje mjesto na kojemu se u nizu nalazi broj a
            break;
        }
    return 0;
}
```



U naprednoj grupi su bili učenici s velikim iskustvom u programiranju koji su bili u fazi prelaženja s QBasica ili Pascala na C++. Već su vrlo dobro shvaćali lagane zadatke i njihov najveći problem su bile sitne greške u sintaksi.

Prvih par dana napredna grupa je prevladala te sitne greške u sintaksi i onda smo krenuli učiti naprednije stvari u C++-u. Prva od njih su bili vektori. Jedan od lakših zadataka s vektorima je bio ovaj: napiši program koji unosi brojeve n i a i nakon njih niz od n brojeva. Program treba ispisati sve brojeve iz niza koji su djeljivi brojem a .



Rješenje:

```
#include <cstdio>
#include <vector>
using namespace std;

vector<int> V;
int n, a, x;

int main(){
    scanf("%d%d", &n, &a);
    for (int j = 0; j < n; ++j){
        scanf("%d", &x);
        if (x % a == 0)
            V.push_back(x);
    }
    for (int j = 0; j < V.size(); ++j)
        printf("%d ", V[j]);
    printf("\n");
    return 0;
}
```



Zrinka Gavran

C++ / STL

Predavač: **Jasmin Velkić, dipl. ing.**

Polaznici:

1. Branimir Filipović
2. Ivan Sesar
3. Mihael Peklar
4. Karlo Kovačić
5. Marin Drabić
6. Ivana Žužić
7. Marko Lazarić

Pohvaljeni: **Ivana Žužić i Marin Drabić**



STL (*Standard Template Library* ili standardna biblioteka predložaka) je nadogradnja programskog jezika C++ mnogim korisnim algoritmima i strukturama podataka često korištenim kako na natjecanjima, tako i inače kod pisanja različitog softwarea u C++-u.

Ova radionica već se tradicionalno održava u sklopu ljetnih kampova i zimskih škola HSIN-a, budući da predstavlja bitan korak u razvoju učenika od C početnika do naprednih korisnika jezika C++. Prema tome, ova je radionica u pravilu namijenjena učenicima koji su dosad barem donekle programirali u jeziku C, iako je za očekivati da i polaznici bez znanja C-a mogu uspješno sudjelovati u radionici. Ipak, potrebno je naglasiti da se na radionici ne obrađuju osnovna pravila sintakse programskog jezika C, već se kreće od naprednijih tema.

Prije upoznavanja s najkorištenijim algoritmima i strukturama podataka iz STL-a, potrebno je upoznati mehanizam jezika C++ koji je omogućio izgradnju STL-a – predloške (*templates*).

Predloške je najlakše objasniti kroz primjere:

Potrebno je napisati funkciju koja prima dva broja kao argumente, te ispisuje manji od dva broja, njihov zbroj i rezultat dijeljenja većeg broja s manjim. Funkciju želimo koristiti s argumentima tipa *int*, pri čemu želimo koristiti cjelobrojno dijeljenje, te s argumentima tipa *float* i *double*, gdje koristimo normalno dijeljenje.

Iako je rješenje navedenog problema u jeziku C trivijalno, ipak zahtijeva umnogostručivanje posla:

```
void obradi_int(int a, int b) {
    int manji, veci;
    if(a > b) {
        manji = a; veci = b;
    }
    else {
        manji = b; veci = a;
    }
    cout<< "Manjibroj: " <<manji<<endl;
    cout<< "Količnik: " <<veci / manji<<endl;
}
void obradi_float(float a, float b) {
    float manji, veci;
    if(a > b) {
```

```

        manji = a; veci = b;
    }
    else {
        manji = b; veci = a;
    }
    cout<< "Manjibroj: " <<manji<<endl;
    cout<< "Količnik: " <<veci / manji<<endl;
}
void obradi_double(double a, double b) {
    doublemanji, veci;
    if(a > b) {
        manji = a; veci = b;
    }
    else {
        manji = b; veci = a;
    }
    cout<< "Manjibroj: " <<manji<<endl;
    cout<< "Količnik: " <<veci / manji<<endl;
}

```

Kao što je vidljivo, za postizanje rješenja potrebno je napisati tri funkcije koje se razlikuju samo po tipovima podataka koje obrađuju. Moguće je donekle pojednostavniti gornje rješenje izbacivši funkciju koja prima *float* argumente i iskoristiti *double* verziju funkcije, no i u tom slučaju se radi samo o aproksimaciji. Isto tako, *double* verziju nije moguće koristiti umjesto prve funkcije, budući da *int* argumenti za operator dijeljenja podrazumijevaju cjelobrojno dijeljenje, dok *double* argumenti znače normalno dijeljenje (poziv funkcije *obradi_double* s cjelobrojnim argumentima neće rezultirati cjelobrojnim dijeljenjem jer će doći do implicitne pretvorbe tipova iz *int* u *double*).

U svakom slučaju, dolazi do umnogostručivanja posla, i to u najgorem obliku: "copy/paste". Naime, kopirajući kôd uvijek postoji mogućnost kopiranja neotkrivenih grešaka u kôdu, koje se kasnije kada su uočene možda neće ispraviti na svim mjestima gdje je kôd kopiran (pažljivi čitatelj će uočiti da sva tri gornja kôda imaju logičku pogrešku).

U C++-u ovaj se problem rješava vrlo jednostavno:

```

<template T>
void obradi(T a, T b) {
    Tmanji, veci;
    if(a > b) {
        manji = a; veci = b;
    }
    else {
        manji = b; veci = a;
    }
    cout<< "Manjibroj: " <<manji<<endl;
    cout<< "Količnik: " <<veci / manji<<endl;
}

```

Kôd funkcije napisan je neovisno o tipu podataka koji obrađuje, te se (u konkretnom slučaju) može primijeniti na svaki tip podataka koji podržava operacije usporedbe i dijeljenja. Poziv funkcije može izgledati ovako:

```

    obradi<int>(5, 2);
    obradi<double>(5.0, 2.0);

```

ili ovako:

```

    obradi(5, 2);
    obradi(5.0, 2.0);

```

gdje će kompajler prema tipovima podataka u pozivu funkcije odrediti koju verziju funkcije pozvati.

Kao što je ranije rečeno, osnovni elementi STL-a su algoritmi i strukture podataka, odnosno spremnici u STL žargonu. Mehanizam predložaka je ključan za elegantno generičko ostvarivanje ovih algoritama i struktura podataka upravo zbog činjenice da najveći broj njih ne ovisi o konkretnom tipu podataka, naravno uz neka razumna ograničenja.

Tako su, na primjer, spremnici (čiji je zadatak pohrana podataka istog tipa) u velikom broju slučajeva potpuno neovisni o konkretnom tipu podataka (npr. niz ili povezana lista), ili imaju male zahtjeve na tipove podataka koje spremaju (npr. skup implementiran binarnim stablom zahtjeva mogućnost usporedbe dva elementa istog tipa). Upravo zato je implementiranje ovih spremnika poželjno korištenjem mehanizma predložaka – kôd je potrebno napisati samo jednom i koristiti za različite tipove podataka.

Za algoritme vrijedi isto: algoritam za sortiranje je neovisan o tipu elemenata koje sortira, dok god ih može uspoređivati.



Prije upoznavanja s konkretnim algoritmima i spremnicima, potrebno je objasniti koncept ortogonalnosti algoritama i spremnika korištenjem mehanizma iteratora. Naime, algoritmi uvijek obavljaju neku operaciju nad skupom elemenata istog tipa. Takvi elementi se stoga nalaze u nekom spremniku, ponekad nizu, ponekad povezanoj listi, ponekad dvostruko povezanoj listi, binarnom stablu i sl. Neovisnost algoritma o konkretnoj implementaciji spremnika jednako je važna kao i neovisnost o tipu podataka elemenata.

Kao što je već rečeno, neovisnost o tipu podataka postiže se predlošcima, a neovisnost o tipu spremnika postiže se iteratorima.

Iterator je koncept koji omogućava kretanje (iteriranje) kroz spremnik neovisno o konkretnom tipu spremnika kroz koji se iterira. Osnovne operacije koje svaki iterator mora podržavati, kako bi bio iskoristiv u algoritmima su (u slučaju jednostavnog tzv. Forward iteratora):

- Pristup vrijednosti elementa spremnika na koji se iterator odnosi (dereferenciranje)
- Usporedba s drugim iteratorom (koristi se za provjeru kraja spremnika)
- Prebacivanje iteratora na sljedeći element spremnika

Dodatno, neki iteratori podržavaju i sljedeće operacije:

- Prebacivanje iteratora na prethodni element spremnika (Bidirectional iterator)
- Prebacivanje iteratora na proizvoljni element spremnika (Random access iterator)

Dok god svaki spremnik pristup svojim elementima dopušta putem iteratora, potpuno je svejedno koja je unutarnja implementacija tog spremnika.

Slijedi pregled najčešće korištenih algoritama i spremnika. Detaljan popis podržanih metoda, uz kratke opise implementacije moguće je pronaći na <http://www.cplusplus.com/reference/stl/>.

Spremnici

- *vector*

Vector je spremnik koji je implementiran nizom, nudi pristup elementima putem Random Access iteratora, brine se o alociranju memorije kako raste i sl. Složenost pristupa proizvoljnom elementu je $O(1)$, kao i dodavanje na kraj, ali brisanje i dodavanje u sredinu je $O(n)$. Primjeri upotrebe:

```
vector<int> v;
vector<int> v1(15, 8);
vector<int> v2(100, 5);
vector<int> v3(v2.begin(), v2.begin() + 15);

v.push_back(11);
v.push_back(12);

v3.reserve(1000);
v3.resize(10, 7);
v3.insert(v3.begin() + 4, 3, 2);
v3.insert(v3.begin() + 8, v1.begin() + 3, v1.begin() + 7);
v3.insert(v3.begin() + 8, l.begin(), l.end());
v1.erase(v1.begin() + 1, v1.begin() + 10);

for( vector<int>::iterator it = v.begin(); it != v.end(); ++it)
    cout << *it << endl;
```

Kôd je prilično jednostavan i jasan, opisi metoda su vidljivi na ranije priloženoj adresi. Potrebno je primijetiti korištenje iteratora u zadnjem dijelu kôda.

- *list*

List je spremnik implementiran korištenjem povezone liste. Omogućuje dodavanje i brisanje na bilo kojem mjestu liste u $O(1)$, ako je poznat iterator na element koji se briše ili ispred onoga koji se dodaje. Upotreba je slična kao u slučaju vektora

- *set*

Set je spremnik koji je implementiran binarnim stablom. Sve operacije su $O(\log n)$. Bitno svojstvo seta je da eliminira duplikate, odnosno, višestruko dodavanje istih elemenata će rezultirati samo jednim elementom u spremniku.

- *stack i queue*

Stack i Queue su implementacije stoga i reda, kao LIFO i FIFO struktura.

- *map i multimap*

Map je spremnik koji omogućuje spremanje parova (ključ, vrijednost), uz efikasnu $O(\log n)$ pretragu prema ključu. Multimap omogućava spremanje više vrijednosti za svaki pojedini ključ.

Primjer programa koji broji pojavljivanje svake riječi u tekstu i ispisuje ih u silaznom redoslijedu prema broju pojavljivanja:

```
map<string, int> m;
multimap<int, string, greater<int>> mm;

for(;;)
{
    string inp;
    getline(cin, inp);
```

```
if(inp == "kraj")
    break;

int pos;
do
{
    pos = inp.find(" ");
    if(pos == -1 && inp.length() > 0)
        m[inp]++;
    else
    {
        m[inp.substr(0, pos)]++;
        inp.erase(0, pos + 1);
    }
}
while(pos != string::npos);
}

for(map<string, int>::iterator it = m.begin(); it != m.end(); ++it)
    mm.insert(make_pair(it->second, it->first));
for(multimap<int, string>::iterator it = mm.begin(); it != mm.end(); ++it)
    cout << it->first << ": " << it->second << endl;
```

Algoritmi

Detaljan pregled algoritama STL-a dostupan je na <http://www.cplusplus.com/reference/algorithm/>.

U sklopu radionice obradili smo neke najčešće korištene:

- *sort*
STL implementacija quick sort algoritma sa $O(n \log n)$ složenosti. Zahtijeva Random Access iterator. Također postoji i *stable_sort*, koji garantira da će elementi s istom vrijednošću biti poredani onim redoslijedom u kojem su bili prije sortiranja.
- *lower_bound/upper_bound/equal_range*
Provodi binarno pretraživanje nad prethodno sortiranim spremnikom koristeći Random Access iterator
- *count_if/find_if*
Broji ili pronalazi prvi element koji zadovoljava neki uvjet. Uvjet se može specificirati kao funkcija koja prima kao argument element spremnika i vraća *true* ako je uvjet ispunjen.



Jasmin Velkić

PYTHON

Predavač: **Nikola Dmitrović, prof.**

Polaznici:

1. Marija Gegić
2. Petar Orlić
3. Mario Pejinović
4. Leon Starešinić
5. Branimir Filipović
6. Filip Herceg
7. Karlo Kovačić
8. Marin Drabić
9. Ivana Žužić
10. Dino Ilić
11. Kristijan Vulinović
12. Marija Vančina
13. Tonko Sabolčec
14. Matteo Kinkela
15. Nikola Pintarić



Pohvaljeni: **Karlo Kovačić, Matteo Kinkela, Tonko Sabolčec i Nikola Pintarić**

Na osnovu pozitivnih iskustava s jubilarnog 20. Kampa, drugu godinu za redom se na Kampu organizirala i radionica iz programskog jezika Python.

Ovogodišnji polaznici radionice su većinom biliiskusni osnovnoškolski natjecatelji s iskustvom programiranja u Pascalu i C++ bez prethodnih iskustava s Pythonom. Poznavanje tih jezika im je omogućilo da bez većih problema, počevši od osnova te preko složenijih elemenata dođu u mogućnost rješavati natjecateljske zadatke u Pythonu. Na taj način su polaznici, između ostalog, primjenjivali Python kao jezik koji može zamijeniti Basic i Pascal u učenju osnova programiranja te kao jedan od jezika na natjecanjima. Ovaj jezik je svojim karakteristikama ponovno privukao polaznike radionice koji su iskazali želju za njegovim daljnjim proučavanjem.



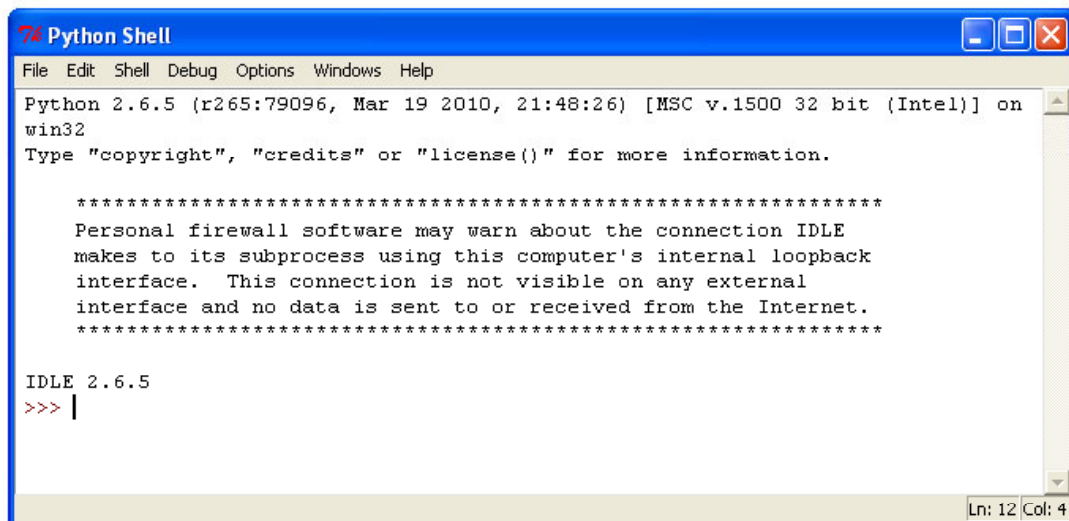
Python je besplatan programski jezik, s *Open Source* licencom, podržan na različitim operacijskim sustavima. Prije svega, Python je interpreter, odlikuje ga jednostavna sintaksa te podržava različite stilove programiranja, od strukturnog do objektno-orijentiranog. Prednost Pythonu daje "otvorena" licenca, te besplatno korištenje i preuzimanje programskog paketa i dodatnih modula kao i mogućnost rada na različitim platformama (Linux, Windows, Unix,...). Izvorni tvorac Pythona je Guido van Rossum, koji je 1991. godine osmislio osnovne principe. Danas Python razvija i održava organizacija "Python Software Foundation".

Teme koje su se poučavale na radionici su:

- Uvodna prezentacija, osnovne karakteristike jezika, njegova prošlost i budućnost, instalacija podrške za pisanje programa u Pythonu;
- Python – kao interpreter i programibilni kalkulator;
- Pojam varijable, standardni tipovi podataka, aritmetički i logički operatori;
- *If, for, while, break, continue* u Pythonu;
- Pojam stringa u Pythonu;
- Složeni tipovi podataka: lista (*lists*), N-torka (*tuple*), rječnik (*dictionary*);
- Karakteristične funkcije na listama, N-torkama i stringovima
- Rješavanje zadataka s ovogodišnjeg školskog, županijskog i državnog natjecanja u programiranju za osnovne škole.

Radno okruženje Pythona:

IDLE (*Integrated DeveLopment Environment*) je radno okruženje koje omogućava jednostavno pisanje kôda u Pythonu. Upisane naredbe se odmah interpretiraju u IDLE okruženju. Ako se program želi prvo napisati a zatim izvršavati, tada treba iz izbornika „File“ izabrati „New window“ te napisani i spremljeni program pokretati pomoću opcije „Run Module“ u izborniku „Run“. IDLE izvršava kôd u Pythonu u odvojenom procesu koji se restarta svakim pokretanjem „Run“ opcije u radnom okruženju.



PYTHON IDLE

Jednostavni tipovi podataka u Pythonu:

Podaci koje računalo sprema u memoriju mogu biti različitog tipa. To mogu biti neke brojčane vrijednosti koje računalo promatra iz perspektive količine memorije koju zauzimaju a nama one predstavljaju, npr., temperaturu mora u nekom periodu promatranja.

Varijabli u Pythonu nije potrebno prije korištenja navoditi tip podatka s kojim će raditi. Ali, tipovi podataka su potrebni kako bi se odredile operacije koje je moguće raditi na varijablama tog tipa te načini spremanja podataka u memoriju. Python ima pet standardnih tipova podataka. Dijelimo ih na jednostavne (brojevi i stringovi) te složene (liste, N-torke i rječnik).



Aritmetički operatori:

Aritmetički operatori omogućuje kreiranje izraza. Razlikujemo unarne i binarne operatore. Python raspolaže sa sljedećim operatorima:

Operator	Primjer	Objašnjenje
+	$x=a+b$	Zbraja vrijednosti a i b (varijable ili vrijednosti) te ih sprema u varijablu x
-	$a-b$	Oduzima vrijednosti a i b
**	$a**b$	Potencira vrijednost a na b
*	$a*b$	Množi vrijednosti a i b
%	$x=a\%b$	U varijablu x sprema ostatak pri cjelobrojnom dijeljenju vrijednosti a s b . Može pronaći ostatak i pri dijeljenju dva realna broja ($2.5\%1.3=1.2$).

Posebnu pažnju treba obratiti na dva operatora dijeljenja „/“ i „//“. Operator „/“ će izvoditi cjelobrojno dijeljenje ako su oba operanda cjelobrojnog tipa te obično dijeljenje ako je barem jedan od operanda realan. Operator „//“ na cijelim brojevima izvodi uobičajeno cjelobrojno dijeljenje dok na realnim brojevima nakon dijeljenja zanemaruje decimalni dio u rezultatu. Primjer upotrebe ovih operatora:

<pre>>>> 9/2 4</pre>	<pre>>>> 8.5/2 4.25</pre>	<pre>>>> 9//2 4</pre>	<pre>>>> 6.5//2 3.0</pre>
-------------------------------	------------------------------------	--------------------------------	------------------------------------

Ako se u izrazu pojavljuje više operatora, tada se redoslijed izvršenja određuje na osnovu prioriteta. Prioriteti imaju horizontalnu i vertikalnu hijerarhiju koja je zadana na sljedeći način:

Prioritet	Operator	Redoslijed izvršenja
1	()	Od unutarnjih prema vanjskim zagradama
2	**	
3	// * / %	S lijeva na desno u izrazu
4	+ -	S lijeva na desno u izrazu

Uvjetni operatori:

Uvjetni operatori omogućuju kreiranje logičkih izraza koji nakon evaluacije općenito mogu poprimiti samo dvije vrijednosti: *istinu* i *laž*. Logički izrazi se koriste u petljama i naredbi odlučivanja, ali i prilikom određivanja raznih uvjeta koje algoritam treba ispunjavati.

- Relacijski operatori

Izrazi koji koriste relacijske operatore, nakon evaluacije mogu poprimiti cjelobrojnu vrijednost jedan ili nula, ovisno je li izraz evaluiran kao istina ili laž. Relacijski operatori su: jednako (==), različito (!= ili <>), veće (>), manje (<), manje ili jednako (<=) te veće ili jednako (>=). Pravila evaluacije za pojedini od ovih operatora su jednostavna i slična u svim programskim jezicima.

- Logički operatori

Nad operandima logičkog tipa možemo primjenjivati logičke operatore. Logički operatori su: *and* – konjunkcija (I), *or* – disjunkcija (ILI) te *not* – negacija (NE). Pravila koja Python koristi pri evaluaciji logičkih izraza s logičkim operatorima *and* i *or* su malo drugačija od uobičajenih. Opišimo ih na primjeru ako su *A* i *B* neki logički izrazi.

A and B – rezultat ove logičke operacije je laž ako je logički izraz *A* lažan (logički izraz *B* se tada i ne evaluira). Ako je *A* istinit, tada je rezultat vrijednost izraza *B*;

A or B – rezultat ove logičke operacije je istina ako je izraz *A* istinit. Ako je *A* lažan, tada je rezultat vrijednost izraza *B*.

- Bitovni operatori

Bitovni operatori djeluju na bitove i na cijele brojeve u bitovnom zapisu. Neki od njih mogu i mijenjati brojeve na razini bitova. Djelovanje bitovnih operatora je zadano tablicom.

Operator	Primjer	Objašnjenje
~ (NOT)	~x	$\sim 1=0$ i $\sim 0=1$ ($\sim x=-(x+1)$)
& (AND)	x&y	$1\&1=1$, inače je rezultat 0
(OR)	x y	$0 0=0$, inače je rezultat 1
^(XOR)	x^y	$1\wedge 1=0$ i $0\wedge 0=0$, inače je rezultat 1
<<	x<<y	Pomiče bitove u bitovnom zapisu broja <i>x</i> za <i>y</i> mjesta u lijevo. Ustvari, množi <i>x</i> s 2^y
>>	x>>y	Pomiče bitove u bitovnom zapisu broja <i>x</i> za <i>y</i> mjesta u desno. Ustvari, cjelobrojno dijeli <i>x</i> s 2^y

- **Operatori pripadnosti**
Operatori pripadnost, *in* i *not in*, provjeravaju pripadnost elementa nekoj sekvencijalnoj strukturi kao što je string, lista ili N-torka i o tome vraćaju odgovarajuću vrijednost 1 ili 0.
- **Operatori istovjetnosti**
Operatori istovjetnosti, *is* i *is not*, uspoređuju memorijske lokacije dvaju objekata i vraćaju o tome odgovarajuću vrijednost 1 ili 0.

Prioriteti operatora:

Python poštuje određene prioritete pri evaluaciji izraza koji imaju uvjetne i ostale operatore. Ti prioriteti se mogu mijenjati korištenjem zagrada. Između operatora postoji vertikalna hijerarhija (od grupe operatora s najmanjim prioritetom prema grupi operatora s najvišim prioritetom) i horizontalna hijerarhija (s lijeva na desno). Operatori u grupi istog prioriteta se u izrazu izvršavaju ovisno o mjestu pojavljivanja, s lijeva na desno.

NAJMANJI PRIORITET	=
	not → and → or
	in, not in, is, is not, <, <=, >, >=, <>, !=, ==
	→ ^ → &
	<<, >>
	+, -
	*, /, //, %
	+x, -x, ~x
NAJVEĆI PRIORITET	**

Zadnjeg dana radionice smo organizirali natjecanje koje je pokazalo veliki napredak u odnosu na početne dane radionice. Na kraju, polaznici su tijekom radionice ulagali dodatne napore da pretraživanjem interneta i metodom pogreški i pokušaja zajedno otkrivaju dodatne mogućnosti koje pruža Python. U tim pokušajima, kao i na završnom natjecanju su se najviše istaknuli polaznici Karlo Kovačić, Matteo Kinkela, Tonko Sabolčec i Nikola Pintarić koji su za svoje napore i nagrađeni na radionici.



Nikola Dmitrović

VISUAL BASIC

Predavač: **Marijan Jurešić, dipl. ing.**

Polaznici:

1. Marina Samblić
2. Lovro Mršić
3. Fran Vasilić
4. Karlo Sintić
5. Carlo Cola
6. Ivan Žmirić
7. Dino Ilić
8. Kristijan Vulinović

Pohvaljeni: **Carlo Cola, Dino Ilić,
Lovro Mršić i Kristijan Vulinović**



Visual Basic je radionica gdje se učenici susreću s nekim znanjima koja im neće koristiti u samim natjecanjima, ali su ta znanja danas primjenjiva. Učenici se upoznaju s tehnologijama koje su danas standard i na taj način dobivaju osjećaj da rade nešto što bi mogli primijeniti. Upravo stoga je ova radionica dosta popularna među polaznicima.

Obradene teme:

- Izrada Windows forms aplikacije
- Postavke formi i kontrola
- Sintaksa koda
- Button kontrola
- CheckBox kontrola
- ListBox kontrola
- Label kontrola
- TextBox kontrola
- Progressbar kontrola
- Globalne varijable
- Funkcije i sub-ovi
- Moduli



Tijekom prvog dijela radionice učenici su se upoznali sa sučeljem Visual Basica, načinom kreiranja projekta, spremanje projekta,... Isto tako, velika je pažnja posvećena objektima grafičkog korisničkog sučelja. Objasnjeno je da svaki od tih objekata ima svoja nekakva svojstva te radnje koje se nad objektima mogu izvršavati. Naglašeno je da se ovdje radi o dosta drugačijem obliku programiranja, od onoga s kojim su se učenici susreli. Radi se o programiranju gdje je zapravo "program" odgovor na neki događaj, tzv. programiranje upravljano događajima (*event-driven programming*). Za vrijeme čitavog trajanja radionice inzistirano je na pojmovima objekt, svojstvo i metoda jer je to upravo ono na čemu se bazira današnje programiranje. Kako se je radilo o uglavnom mlađim polaznicima nije se ulazilo u dubinu objektno orijentiranog programiranja već se je o objektima govorilo samo kroz elemente grafičkog korisničkog sučelja.

Tijekom radionice učenici su kreirali jednostavan program za rad s tekstualnim datotekama (pisanje teksta, spremanje, otvaranje, oblikovanje,...) kroz ovaj primjer upoznali su osnovne elemente grafičkog korisničkog sučelja (okvir za tekst, padajući izbornik, izbornik, dijaloškim prozorima,...).

Drugi dio radionice bilo je upoznavanje s bazama podataka. U tom dijelu bilo je obrađeno kreiranje tablica u MS Accessu, jednostavni upiti (*INSERT*, *SELECT*, *DELETE*, *UPDATE*) te nešto kompliciraniji upiti gdje se selektiraju podaci iz dviju tablica.



U naprednijem dijelu radionice naučili su kako povezati bazu podataka s Visual Studiom na primjeru programa za vođenje podataka o polaznicima, voditeljima i radionicama na Kampu. Prvo smo osmislili i kreirali bazu podataka sa svim tim podacima, a zatim smo kreirali program s višestrukim formama koji je implementirao osnovne funkcionalnosti (dodavanje polaznika, dodavanje predavača, pretraživanje polaznika, dodjeljivanje polaznika u grupu,...).

Po završetku trajanja radionice bila je provedena provjera znanja, gdje su učenici trebali kreirati bazu podataka s proizvodima te jednostavnu aplikaciju za dućan. Svi učenici su kreirali bar osnovnu funkcionalnost programa.

Svi su polaznici upoznati s osnovama rada te vjerujem da će mnogi od njih nastaviti s radom aplikacija u Visual Basicu-u.



Izrada ANDROID aplikacija

Predavač: **Jasmin Velkić, dipl. ing.**

Polaznici:

1. Ivana Žužić
2. Marko Lazarić
3. Vilim Lendvaj
4. Josip Klepec
5. Marin Drabić
6. Dino Ilić
7. Kristijan Vulinović
8. Karlo Kovačić
9. Nikola Pintarić
10. Mihael Peklar

Pohvaljeni: **Nikola Pintarić i Kristijan Vulinović**



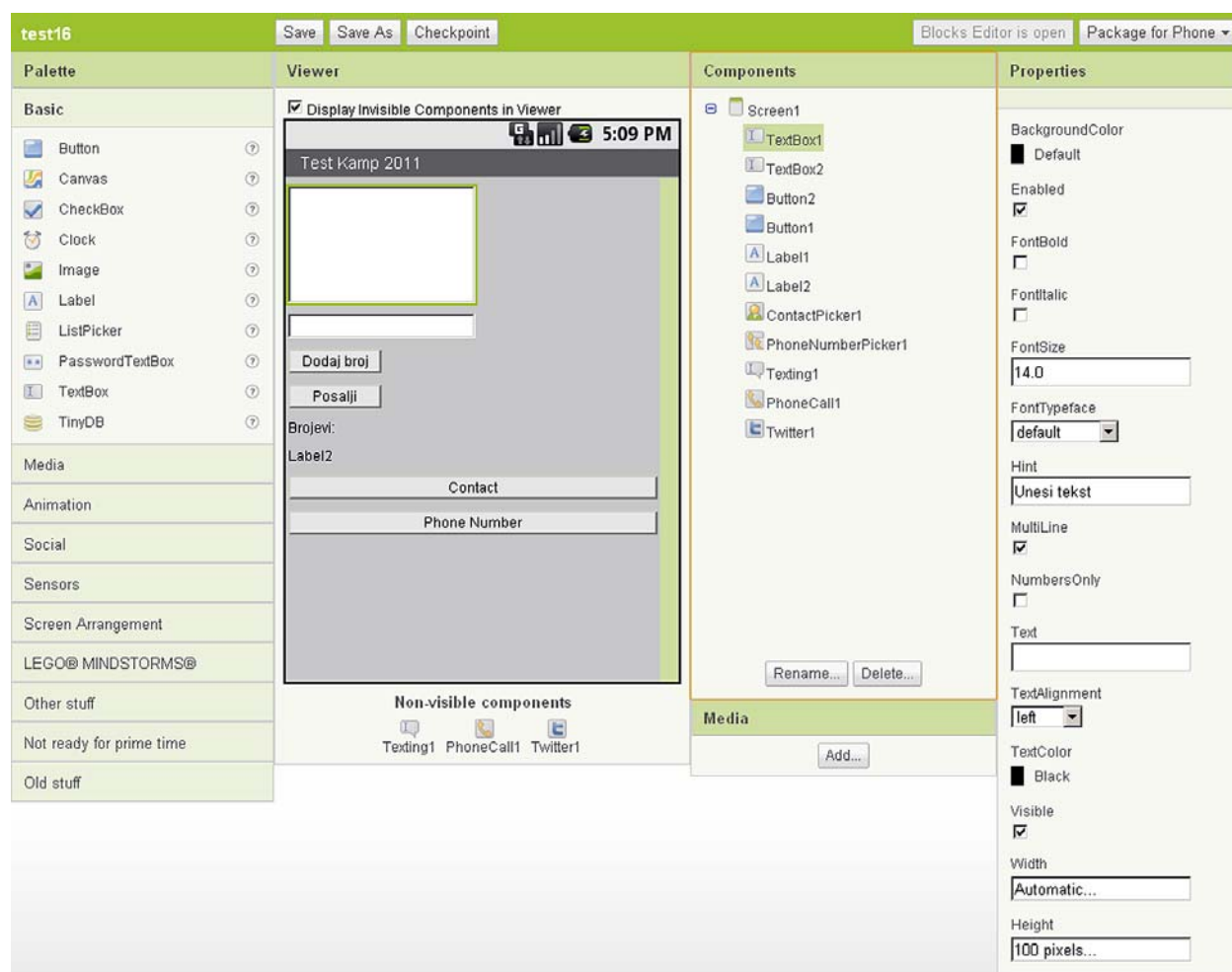
Ovogodišnja novost na Kampu mladih informatičara Hrvatske je radionica za izradu Android aplikacija. Kako se oslobodio termin u predavanjima zbog nedovoljne zainteresiranosti učenika za Linux radionicu, odlučili smo ponuditi nešto novo, zanimljivo i aktualno. Kako većina polaznika u današnje vrijeme posjeduje *Smartphone*, osmislili smo radionicu koja bi upoznala učenike s osnovama programiranja aplikacija za mobilne telefone opremljenim Android operacijskim sustavom. Iako se standardno za profesionalni razvoj Android aplikacija koristi Java *Android SDK* (uobičajeno u *Eclipse* okruženju), smatrali smo da trebamo sustav prilagođeniji početnicima. U tu svrhu odlučili smo se za Googleov Beta alat *Appinventor*.

APP INVENTOR

Appinventor je skup alata koji na vrlo jednostavan i pristupačan način objedinjuje razvojno okruženje za izradu Android aplikacija. Sastoji se od web baziranog korisničkog sučelja, iznimno privlačnog Java editora i emulatora Android telefona. Također, posjeduje mogućnost izravnog spajanja telefona putem USB porta za pokretanje i testiranje mobilnih aplikacija.

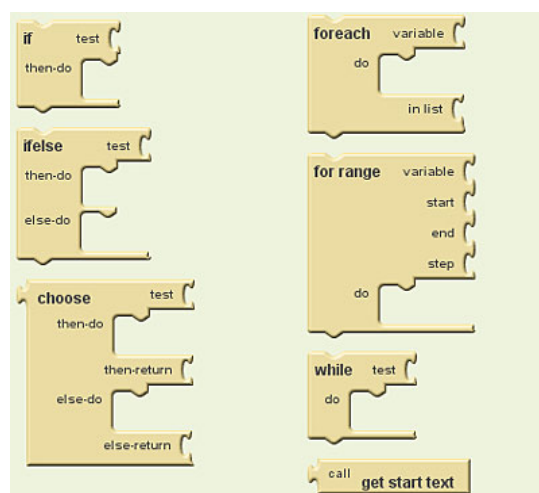


Glavno korisničko sučelje, osim za rukovođenje projekata, prvenstveno služi za vizualnu obradu i dizajn projekta – aplikacije. Slično kao i u drugim vizualnim razvojnim alatima, ovdje se definiraju objekti koji se pojavljuju u grafičkom sučelju novonastale aplikacije.



Glavno korisničko sučelje i primjer vizualnog dizajna aplikacije

Blocks Editor – editor kôda aplikacije pisan u Javi na vrlo jedinstven i intuitivan način predstavlja programski kôd. Linije kôda programskog jezika zamijenjene su blokovima oblika puzzli, koji sprječavaju upotrebu krive sintakse pa čak i umjeren broj logičkih pogrešaka. *Blocks Editor* omogućava grafičko programiranje, postepenu gradnju elemenata kao i svojevrsnu slobodu "pisanja". Dijelove programa moguće je po želji rasporediti po radnoj površini ili grupirati radi lakšeg snalaženja. Osim glavnih naredbi, moguće je izabrati napredne funkcije koje se mogu odnositi na sve objekte unutar aplikacije. Svaki od objekata još ima i svoje specifične naredbe, poput: *button.enabled*, *button.heigth*,...i specifične događaje koji se mogu aktivirati: *button.click*, *button.focus*,... kod objekta tipa "gumb".



Set naredbi za grananje, provjere i petlje

Emulator – mala aplikacija koja virtualizira Android operacijski sustav u svrhu testiranja projekta. Pokreće se Androidovo grafičko sučelje s limitiranim mogućnostima i kompajlirana aplikacija koju smo napravili. Iako operacijski sustav, a samim time i pripadajuće aplikacije su optimizirane za upravljanje dodiranjem prsta, moguće je testirati mišem i tipkovnicom. Stoga, preporučljivo je konačnu verziju svakako testirati na stvarnom telefonu.



Emulator – simulira Android OS



Izvršavanje aplikacije "Asocijacija" izrađene na Kampu

OBRADIVANE TEME

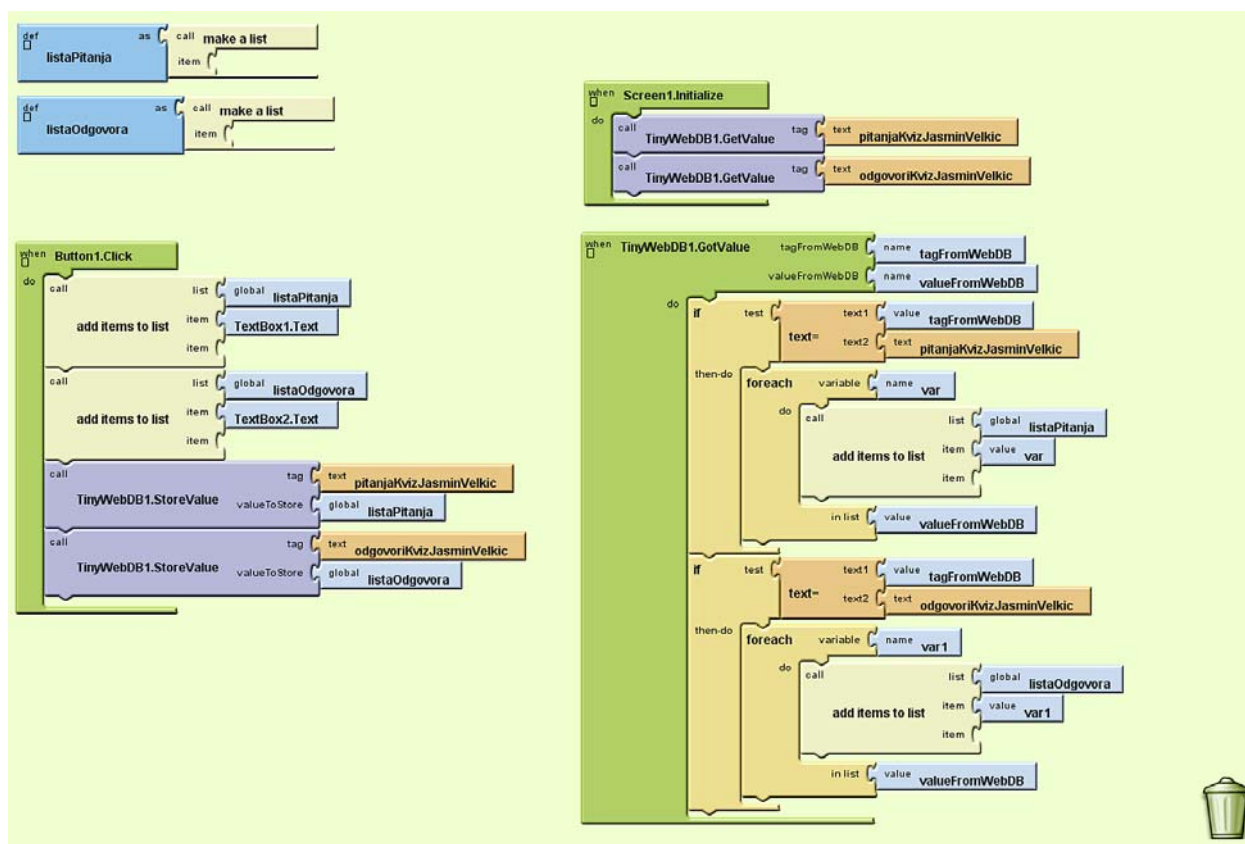
Radionica je okupila desetak polaznika, većinom mladih učenika. Svi su imali predznanja iz programiranja tako da nije bilo potrebe za teoretskim objašnjavanjem programerske logike i opisa naredbi već njihova implementacija na jedinstven i drugačiji način unutar paketa Appinventor.

Prvi sat održano je uvodno predavanje. Objašnjen i demonstriran je rad unutar Appinventor sučelja, pokrivene su sve dostupne opcije, dizajn projekta i vizualna obrada objekata. Zatim smo proučili *Blocks Editor*, prostor u kojemu se slaganjem „kockica“ ostvaruje programski kôd. Polaznicima je dano da sami testiraju razne kombinacije naredbi i ujedno da se priviknu na „vizualno“ programiranje.

U nastavku radionice pokušali smo napraviti prvi kompleksniji program. Pošto je "gumb" jedan od najvažnijih i najupotrebljavanijih objekata, započeli smo sa svojstvima, razjasnili što su to "dogadaji – events" i nastavili s konkretnim programiranjem objekta. Pa smo tako za početnički primjer napravili aplikaciju koja zbraja dva broja.

Kako bi mogli raditi s promjenjivim vrijednostima (varijablama), obradili smo objekte za unos i ispis teksta. Kao primjer napravili smo aplikaciju koja dodaje uneseni broj na kraj ispisanog teksta u novom polju. Za konstrukciju raznovrsnijih stvari, obradili smo liste, varijable, grananje, *for/while* petlje i sl. Također smo koristili i naprednije objekte poput ubacivanje i manipulacija slika, *timer-event*, *checkbox*, *imagesprite* za pokretnu animaciju, itd. Za primjer upotrebe kompleksnijih objekata, napravili smo aplikaciju "Asocijacije" i igricu "Krtica".

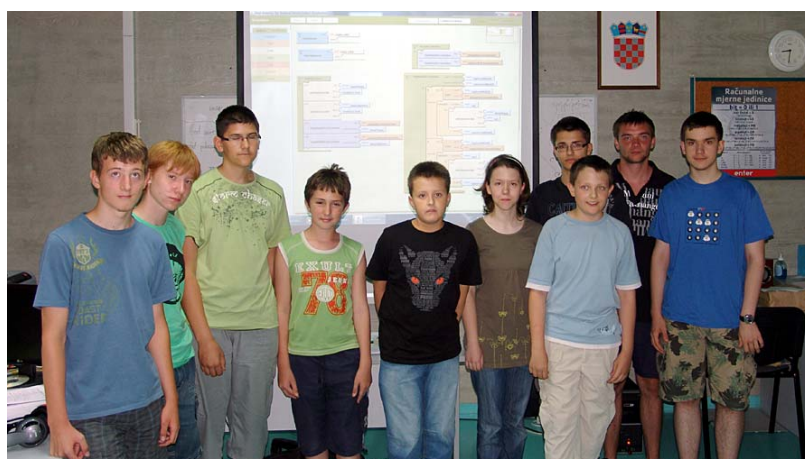
Naprednije aplikacije zahtijevale su spremanje i čitanje podataka iz stalne memorije. U tu svrhu koristili smo objekt *TinyDB* koji ima mogućnost čitanja i pisanja u memoriju Android telefona. Time smo mogli unaprijediti našu aplikaciju "Asocijacije" tako što smo оформili bazu pitanja i odgovora čime je postala raznovrsnija.



Blocks Editor – primjer programa kojim se administrira baza za igru "Asocijacije"

Zadnja dva sata odvojili smo za samostalnu izradu aplikacija. Svatko od polaznika je trebao osmisliti originalnu ideju uz primjenu novostečenih znanja. Ideja je bilo raznih. Iako su najviše prevladavale igrice, našla se i jedna enciklopedija, napredni kalkulator, unaprijeđena verzija asocijacija i sl.

Kao nešto novo na Kampu, mislim da su učenici dobro prihvatili ovu radionicu i brzo se snašli u novom okruženju. Vjerujem, a i mnogi su osobno rekli, kako će sami nastaviti s upoznavanjem i proširivanjem znanja o izradi Android aplikacija. Najbolji među njima, Kristijan Vulinović i Nikola Pintarić dobili su pohvale za iznimno zalaganje i trud.



HTML/PHP

Predavač: **Gordan Nekić**

Polaznici:

1. Ivana Žužić
2. Marko Lazarić
3. Filip Herceg
4. Carlo Cola
5. Ivan Žmirić
6. Dino Ilić
7. Kristijan Vulinović
8. Karlo Kovačić
9. Nikola Pintarić



Pohvaljeni: **Ivana Žužić, Marko Lazarić, Nikola Pintarić i Kristijan Vulinović**

Radionicu PHP pohađali su učenici različitih uzrasta i predznanja. Vidjevši da je većini polaznika to bio prvi susret s PHP-om, u nekoliko dana obrađene su sve osnovne pojedinosti PHP-a i neke napredne funkcije te spajanje PHP-a s *MySQL* bazom.

Obrađene teme:

- PHP otvaranje *tagova* u HTML-u i naredba *echo*
- *if*-grananje
- *for* petlja
- *while* petlja
- *include* funkcija
- kôdiranje funkcija
- *session*, *cookies*
- *mail* funkcija
- *rand* funkcija
- *mysql*
- spajanje PHP-a i *mysql*-a
- Login funkcija koja je otporna na SQL injekcije (Sigurnost)
- Registracijska funkcija s automatskim generiranjem lozinke
- Administrator panel za upravljanje stranicom

Realizacija radionice:

Radionica je započela uvodom u PHP uz objašnjenje osnovnih naredba. Sljedeći dan započeli smo sa samostalnim projektom tako da je svaki polaznik radio svoju vlastitu stranicu. Treći dan objasnili smo osnove *MySQL* baze te spajanje PHP-a na *MySQL* bazu.

Primjer kôda:

```
//Spajanje na bazu

$db_host="localhost"; $db_user="korisnik"; $db_pass="lozinka"; $db_name="Imebaze";

$db_id = mysql_connect($db_host, $db_user, $db_pass) or die("Nemogu se spojiti na Bazu");

mysql_select_db($db_name, $db_id) or die("Baza nije pronađena");

// kraj koda
```



Sljedeći dan napravili smo registracijsku stranicu na kojoj su se korisnici mogli registrirati. Skripta bi korisnike automatski zapisala u bazu i poslala nasumično generiranu lozinku korisnika na e-mail. Peti dan polaznici su s lakoćom shvatili *MySQL* naredbe te su uz moju pomoć uspješno napravili login funkciju koja je prijavljivala korisnike. Obradili smo i temu kako se možemo zaštititi od SQL injekcija.

Šesti dan polaznici su preradili stranicu tako da su samo registrirani korisnici imali sve mogućnosti. Na kraju su polaznici dobili zadatke kako bi se provjerilo njihovo znanje po završetku radionice.

Svi su polaznici upoznati s osnovama rada te vjerujem da će mnogi od njih nastaviti s radom stranica u PHP-u.



Gordan Nekić

ALGORITMI za OŠ

Predavač: **Nikola Dmitrović, prof.**

Polaznici:

1. Branimir Filipović
2. Marija Gegić
3. Filip Herceg
4. Karlo Kovačić
5. Vedran Kurdija
6. Petar Orlić
7. Mario Pejinović
8. Marin Drabić
9. Ivana Žužić
10. Leon Starešinić
11. Marko Lazarić
12. Josip Klepec
13. Nikola Pintarić



Pohvaljeni: **Branimir Filipović, Ivana Žužić, Marija Gegić i Vedran Kurdija**

Većina polaznika ove radionice je iskazivala veliki interes i sposobnost u shvaćanju i primjenjivanju algoritama koji su se radili tijekom radionice. Pored unaprijed planiranih algoritama, polaznici su iskazivali želju za dodatnim radom te su neki algoritmi rađeni upravo prema njihovim dodatnim željama. Ideja radionice je bila forsirati shvaćanje i primjenu algoritama krenuvši od onih jednostavnih prema složenijim. Poučavani algoritmi su kodirani u svim programskim jezicima koji su zastupljeni na natjecanjima (Basic, Pascal, C++, a pokazno i u Pythonu).

Teme koje su se poučavale na radionici su:

- Osnovni algoritmi
 - o algoritam za pečenje palačinki – polaznici su trebali definirati korake ovog životnog algoritma te međusobno otkrivati pogreške i nelogičnosti u njegovoj provedbi;
 - o čudno traženje maksimuma – pronaći maksimalnu vrijednost na što neobičniji način;
 - o uvod u rekurzije – suma prvih n prirodnih brojeva;
 - o Peanova aritmetika – zbrajanje, množenje i potenciranje;
- Rekurzije
 - o kreirati sve troznamenkaste brojeve koji se sastoje od 1, 2, 3, 4 i 5;
 - o kreirati sve permutacije niza 12345;
 - o problem kraljice – na koliko načina na šahovsku ploču možemo staviti 8 kraljica tako da se nijedan par tih kraljica ne napada;
- Dinamičko programiranje
 - o Mjenjačnica – problem optimizacije količine novca u uvjetima proizvoljne promjene novca iz jedne valute u drugu uz poznat dnevni omjer tih valuta;
 - o Piramida – pronalaženje optimalnog puta uz maksimizaciju dodatne vrijednosti;
 - o Tunel – pronalaženje optimalnog puta uz minimizaciju dodatne vrijednosti;

- Uvod u teoriju grafova
 - o BFS – objašnjenje algoritma i implementacija;
 - o DFS – objašnjenje algoritma i implementacija;
 - o BFS – objašnjenje algoritma i implementacija;
 - o Guliver – DMIH 2007
 - o Tramvaj – Županijsko 2002.

ZADATAK Mjenjačnica // Domagoj Kusalić: „Napredno programiranje i algoritmi u C i C++“

Banka je odlučila kako će sljedećih n dana mijenjati omjer dolara i eura u svojim mjenjačnicama. Saznali smo tu informaciju i želimo što više zaraditi. Na raspolaganju imamo 1000 eura. Koliko najviše eura možemo imati n -tog dana ako svaki dan po želji možemo mijenjati novac iz jedne u drugu valutu.

Rješenje:

Zadatak je klasična dinamika. Označimo s $D[x]$ maksimalnu količinu dolara koju možemo imati x -ti dan, ako su nam svi novci u dolarima te slično za eure u oznaci $E[x]$. Isto tako, označimo s $koeficijent[x]$ omjer dolara i eura na x -ti dan. Tražimo relaciju koja će povezati stanje na x -ti i $x-1$ dan. Relacija je:

$$D[x] = \max(D[x-1], E[x-1] * koeficijent[x]);$$

$$E[x] = \max(E[x-1], D[x-1] * (1/koeficijent[x]));$$

To znači da sljedeći dan, ako sve novce imamo u dolarima, možemo imati najviše onoliko dolara koliko smo imali prošli dan, ili pak možemo eure od prošlog dana pretvoriti u dolare. Slično vrijedi i za najviše eura u trenutnom danu. Početni uvjeti su $E[0]=1000$ i $D[0]=0$.

ZADATAK Piramida // Domagoj Kusalić: „Napredno programiranje i algoritmi u C i C++“

Unosi se prirodan broj n . U sljedećih n redaka nalaze se pozitivni cijeli brojevi tako da se u x -tom retku nalazi x brojeva. Cilj je u toj piramidi brojeva pronaći put od vrha do dna, kojemu je suma brojeva najveća. Počinjemo u prvom retku i možemo se kretati samo prema dolje ili dolje-desno.

Rješenje:

Definirajmo stanje dinamike kao sumu najboljeg puta koji počinje u vrhu i završava u x retku i y stupcu tj. mjestu (x,y) . To stanje računamo na osnovu prijašnjih stanja koristeći relaciju:

$$DP[x][y] = \max(DP[x-1][y-1], DP[x-1][y] + unos[x][y])$$

Naravno, potrebne su dodatne provjere o valjanosti svake lokacije.

Odabrati pohvaljene sudionike radionice nije bilo lako jer je većina polaznika kvalitetno i s puno entuzijazma radila na zadanim algoritmima. Posebno treba istaknuti nekoliko polaznika koji su bez problema razumjeli i implementirali algoritme koji su se do prije 5-7 godina radili na pripremama olimpijskih timova!



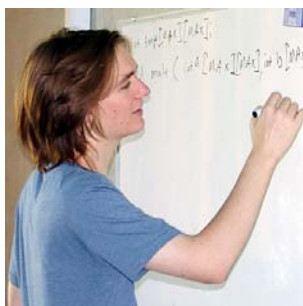
Nikola
Dmitrović

ALGORITMI za SŠ

Predavači: **Adrian Satja Kurdija, Davor Prugovečki, dipl. ing., Gustav Matula i Ivica Kičić**

Polaznici:

1. Mislav Balunović
2. Hristijan Bogoevski
3. Domagoj Čavid
4. Goran Flegar
5. Dominik Gleich
6. Martin Gluhak
7. Matteo Kinkela
8. Zvonimir Medić
9. Neven Miculinić
10. Ivan Paljak
11. Mihael Peklar
12. Drago Plečko
13. Antun Razum
14. Jure Ratković
15. Ivan Sesar
16. Marin Tomić
17. Tomislav Tunković



Pohvaljeni: **Domagoj Čavid, Hristijan Bogoevski, Jure Ratković i Neven Miculinić**

Obrađene su sljedeće teme:

- Tehnika dinamičkog programiranja (DP)
- Napredno binarno pretraživanje
- Logaritamska struktura (*Fenwickovo stablo*)
- Algoritmi u koordinatnoj ravnini
- *Grahamov* algoritam za nalaženje konveksne ljuske (*convex hull*)
- Uporaba matrica u rekurzijama
- Množenje i brzo potenciranje matrica
- *Gaussova* metoda eliminacije
- Prefiks stablo (*Trie*)

Kroz nešto više od tjedan dana, radionica algoritama za srednje škole obuhvatila je mnoge važne teme koje se pojavljuju na informatičkim natjecanjima u srednjoj školi, na natjecanjima kao što su HONI, HIO, Izborne pripreme i međunarodne olimpijade. Važno je napomenuti da to nipošto nije jedina svrha ovih algoritama: oni se jednako široko primjenjuju u cjelokupnoj informatičkoj praksi (iz koje su uostalom i proizašli), u izradi raznih korisničkih programa i slično.

Dinamičko programiranje neizostavan je element ovakve radionice, jer je to jedan od najvažnijih pristupa rješavanju raznih problema. Pokazan je čuveni „*knapsack*“ problem, kao i neke njegove primjene i modifikacije.

Binarno pretraživanje (eng. *binary search*) ne koristi se samo da bismo pronašli element u sortiranom nizu: ta je, doduše, primjena najčešća, ali ova je metoda zapravo mnogo šira i jača. Ponekad je naime vrlo teško ili nemoguće zadatak riješiti razmišljajući kako bismo za zadani ulaz A izračunali izlaz B, pa zato pokušavamo riješiti problem razmišljajući u drugom smjeru ($B \rightarrow A$), tj. kako bismo za neki potencijalni izlaz B provjerili u kojoj mjeri on zadovoljava ulaz A. Na taj način održavamo donju i gornju granicu između kojih se nalazi konačno rješenje i, koristeći gore spomenutu provjeru, korak po korak ih približavamo dok se one ne spoje – tada smo pronašli traženo rješenje.



Logaritamska struktura ili *Fenwickovo stablo* iznimno je važan alat koji omogućava efikasno vršenje operacija na nizu: mijenjanje elementa i upit za zbroj, minimum ili maksimum na nekom intervalu niza. Ispostavlja se da je primjena ove strukture veoma široka i obuhvaća iznenađujuće velik broj zadataka.

Kada su objekti zadani u koordinatnoj ravnini (što nije rijedak slučaj u praksi, a često se javlja i na informatičkim natjecanjima), potrebno je poznavati algoritme za rad s takvim objektima – najčešće je riječ o točkama, pravcima, dužinama, trokutima i općenito poligonima. Polaznici su, primjerice, naučili računati površinu bilo kojeg poligona u ravnini (ne nužno konveksnog), a kao jedan od važnijih algoritama istaknut je „*Graham scan*“ koji služi za nalaženje konveksne ljuske danog skupa točaka (to je najmanji konveksan poligon koji ih sve obuhvaća), jer se „*covex hull*“ koristi vrlo često.

Razne rekurzivne relacije mogu se svesti na operacije s matricama. Inače su matrice predmet proučavanja linearne algebre, ali njihove je primjene na informatičkim zadacima vrlo korisno naučiti jer se često koriste kada uporaba obične rekurzivne relacije nije dovoljno brza: problem tada svodimo na množenje, odnosno brzo potenciranje matrica. Brzo potenciranje općenito je korisno i kada nije riječ o matricama.



Na matrice se prirodno nastavlja problem rješenja sustava jednačbi (na koji se svode mnogi zadaci), jer se on rješava transformacijama matrice, i to *Gaussovom* metodom eliminacije koju su polaznici naučili i primijenili.

Važno je bilo ne zaobići rad sa stringovima, jer se oni svuda pojavljuju. Tako su polaznici naučili koristiti strukturu zvanu prefiks stablo (eng. *trie*), koja uglavnom služi za brzo pronalaženje neke riječi (ili njezinog prefiksa) u danom skupu riječi (npr. u rječniku), ali ima i šire primjene, što su polaznici i uočili u nekoliko odabranih zadataka.

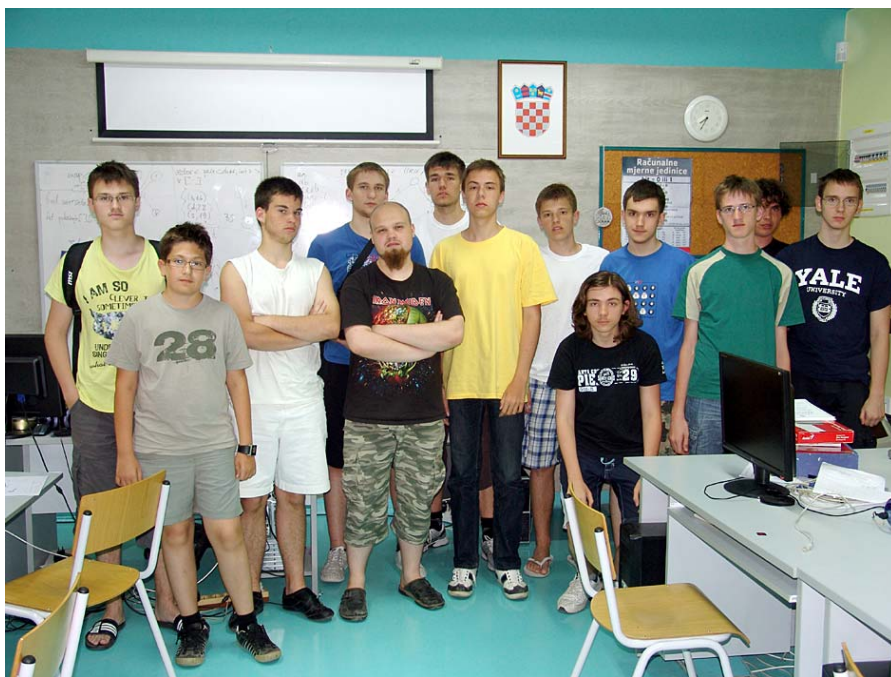
Svaka od ovih tema bila je popraćena brojnim primjerima zadataka na koje se ta metoda primjenjuje. Potom su zadatke rješavali na računalima i testirali na internetu. Evo i primjera zadatka koji su polaznici rješavali:

Dano je $N \leq 100\,000$ točaka u koordinatnoj ravnini (unose se njihove koordinate). Potom je dano $Q \leq 100\,000$ upita: u svakom upitu dana je jedna nova točka za koju treba odgovoriti postoji li trokut, sastavljen od početnih N točaka, koji sadrži danu točku.

Slijedi opis algoritma kojim se rješava zadatak:

Ključno je primijetiti da, ako za upit postoji trokut u kojem se tražena točka nalazi, tada ćemo uvijek moći smjestiti taj trokut na konveksnu ljusku zadanih točaka. Stoga zapravo za svaki upit moramo provjeriti nalazi li se unutar konveksne ljuske. Najprije pronađemo konveksnu ljusku danih N točaka koristeći *Graham scan*. Algoritam koji za svaki upit prolazi po svim dužinama konveksnog poligona i provjerava da li se zadana točka nalazi s iste strane svake dužine, nije dovoljno efikasan jer ima vremensku složenost $O(Q \cdot N)$.

Slijedeći algoritam je dovoljno efikasan. Iz najniže točke konveksne ljuske povučemo sve dijagonale. Trebamo odrediti između koje dvije dijagonale se nalazi upit (dijagonale promatramo kao polupravce). Te dvije dijagonale možemo pronaći binarnim pretraživanjem, jer je u provjeri lako odgovoriti s koje je strane nekog polupravca dana točka. Kad smo ustvrdili koje su to dvije susjedne dijagonale, dovoljno je provjeriti nalazi li se upit unutar trokuta koji zatvaraju najniža točka konveksne ljuske i krajnje točke nađenih susjednih dijagonala. Složenost ovog algoritma je $O(N \log N + Q \log N)$.



Adrian Satja Kurdija

OLIMPIJCI

Predavači: **Davor Prugovečki, dipl. ing. i**
Adrian Satja Kurdija

Polaznici:

1. Ivan Lazarić
2. Mihael Liskij
3. Mislav Balunović
4. Dominik Gleich
5. Ivan Katanić
6. Ivica Kičić
7. Gustav Matula
8. Matija Milišić



Kao i prošlih godina, u sklopu ovogodišnjeg Kampa se održavala radionica Olimpijci. Ova radionica je prvenstveno namijenjena pripremama hrvatskog tima za međunarodna natjecanja (CEOI i IOI). Dodatno, zajedno s glavnim timom hrvatske reprezentacije pripremala se i mlada reprezentacija za nastup na JBOI-u u Rumunjskoj. Tako su voditelji priprema bili Adrian Satja Kurdija (mladi tim) i Davor Prugovečki (glavni tim). Osim spomenutih timova koji će predstavljati Hrvatsku na olimpijadama, radionicu je pohađao i poseban gost iz Makedonije, Hristian Bogoevski, koji je također olimpijac. Njemu se ovakva radionica jako svidjela i općenito rad u društvu s bivšim i trenutnim hrvatskim olimpijcima.

Pripreme su se odvijale dva puta dnevno, u jutarnjem terminu od 8:30 do 13:00 i poslije podnevnom terminu od 18:00 do 20:30. U međuvremenu su polaznici imali slobodno vrijeme za lagani odmor koji su najčešće provodili na plaži. Polaznici su rješavali zadatke koji su bili dodjeljivani na dnevnoj bazi, pri čemu su pojedine dane rješavali zadatke s pojedinog natjecanja (prvi ili drugi dan neke olimpijade), a drugih dana su rješavali tematski zadatke, odnosno skup zadataka koji se odnose na istu tematiku. Zadaci su obuhvaćali razne teme (dinamičko programiranje, teorija grafova, kombinatorika, geometrija, ad hoc, ...) i bili su iz sljedećih izvora:

- Poljske olimpijade
- IOI
- SPOJ i ZATEMAS
- CodeForces
- Državno natjecanje Srbije
- Državno natjecanje Singapura
- Studentska natjecanja HSIN-a



Nakon jutarnjih radionica polaznici su diskutirali o rješenjima pojedinih zadataka te su na poslijepodnevnim radionicama nastavili rješavati jutarnje zadatke s novim informacijama i zaključcima. Od početka radionice se vodila evidencija o riješenim zadacima u obliku tablice na ploči. Time se na neki način održavao natjecateljski duh među polaznicima jer je svako uspješno rješenje pojedinca motiviralo preostale polaznike.

Sva rješenja zadataka su vrlo brzo bila testirana od strane online evaluatora (CodeForces, SPOJ, Poljske olimpijade) i od strane voditelja (prije zadavanja zadataka su voditelji Adrian i Davor pripremili testno okruženje na svojim računalima sa službenim test primjerima te je i to testiranje bilo relativno brzo). Zadaci s državnog natjecanja Singapura su bili srednje težine, ali je na svakom zadatku bilo potrebno obratiti pozornost na detalje. Dodatno, drugi zadatak s tog natjecanja (u nastavku i objašnjen - Tutor) je tipičan zadatak koji se rješava principom dinamičkog programiranja, ali nije lagan ukoliko se najbolji rezultat ide tražiti od kraja (tada je teško osmisliti stanje takvo da se ne prekorači memorijska složenost).

Osim rješavanja zadataka na radionici olimpijci, polaznici su povremeno održavali predavanja naprednih algoritama srednjoškolcima. Tako je Ivica Kičić objašnjavao logaritamsku strukturu (*Fenwickovo stablo*) i njezinu primjenu, dok je Gustav Matula objašnjavao kako koristiti matrice (što mogu predstavljati, kako se potenciraju) i koja im je primjena (izračun rekurzivne funkcije za veliki N). Ova predavanja su pomogla polaznicima da također i sami bolje shvate pojedine algoritme i unaprijede svoju sposobnost prijenosa znanja.

Neki od zadataka koje su polaznici rješavali na radionici olimpijci:

Island (Poljska olimpijada 2008/2009, 3. etapa)

U zadatku je zadana mapa gradova koji su posloženi u konveksan poligon tako da je grad 1 dolje lijevo, grad N dolje desno, a gradovi između njih su sortirani po rednim brojevima. Potrebno je odrediti najkraći put između gradova 1 i N , pri čemu je dozvoljeno kretati se samo po cestama i smije se prelaziti s jedne na drugu ako se one sijeku. Primijetimo da je dovoljno za svaki grad uzeti u obzir samo cestu prema gradu s najvećim indeksom. Tako možemo krenuti od grada 1 i njegove ceste te redom dodavati ostale ceste i graditi put čiji će oblik izgledati kao dio konveksne ljuske. Kako pri dodavanju nove ceste možemo brisati samo zadnje dodane ceste praktično je koristiti *stack* koji će nam garantirati linearnu složenost. Dakle, glavni algoritam je složenosti $O(N)$, dok se zbog pojedinih sortiranja radi pripreme podataka (koje se može izbjeći) složenost povećava na $O(N \log N)$. U nastavku se nalazi rješenje polaznika Ivana Katanića:

```
#include <cstdio>
#include <vector>
#include <cmath>
#include <algorithm>

using namespace std;

const long double eps = 1e-10;

struct point {
    long double x, y;
    point( long double _x = 0, long double _y = 0 ) { x = _x; y = _y; };
};

struct line {
    long double A, B, C; // Ax + By = C
    line( long double a, long double b, long double c ) { A = a; B = b; C = c; };
    line( const point &P, const point &Q ) {
        A = P.y - Q.y;
        B = Q.x - P.x;
        C = A*P.x + B*P.y;
    };
};

int ccw( const point &A, const point &B, const point &C ) {
    long double p = ( B.x-A.x )*( C.y-A.y ) - ( B.y-A.y )*( C.x-A.x );
    return p < -eps ? -1 : p > eps;
}

long double dist( const point &A, const point &B ) {
    return sqrt( ( B.x-A.x )*( B.x-A.x ) + ( B.y-A.y )*( B.y-A.y ) );
}
```

```

point presjek( const line &P1, const line &P2 ) {
    long double d = P1.A*P2.B - P2.A*P1.B;
    return point( (P2.B*P1.C - P1.B*P2.C)/d, (P1.A*P2.C - P2.A*P1.C)/d );
}

const int MAXN = 100005;

vector< int > E[MAXN];
point a[MAXN];
int v1[MAXN], v2[MAXN];

point presjek( int x, int y ) {
    line l1 = line( a[ v1[x] ], a[ v2[x] ] );
    line l2 = line( a[ v1[y] ], a[ v2[y] ] );
    return presjek( l1, l2 );
}

int main( void ) {
    int n, m;
    scanf( "%d %d", &n, &m );
    for( int i = 0; i < n; ++i ) {
        int x, y;
        scanf( "%d %d", &x, &y );
        a[i] = point( x, y );
    }

    for( int i = 0; i < m; ++i ) {
        int a, b;
        scanf( "%d %d", &a, &b ); --a, --b;
        if( a > b ) swap( a, b );
        E[a].push_back( b );
    }

    m = 0;

    int c = 0;
    for( int i = 0; i < n; ++i ) {
        sort( E[i].begin(), E[i].end() );
        int l = n-1;
        for( int j = E[i].size()-1; j >= 0; --j, --l )
            if( E[i][j] != l ) break;

        if( l <= c || l <= i ) continue;

        while( m >= 2 && ccw( a[i], presjek( m-2, m-1 ), a[l] ) <= 0 ) --m;
        v1[m] = i, v2[m++] = l;
        c = l;
    }

    long double ans = 0;
    point last = a[0];
    for( int i = 0; i+1 < m; ++i ) {
        point p = presjek( i, i+1 );
        ans += dist( last, p );
        last = p;
    }
    ans += dist( last, a[n-1] );

    printf( "%.10Lf", ans );
    return 0;
}

```

Tutor (olimpijada Singapura 2010. godine)

Zadatak traži da se za određeno vrijeme T odredi maksimalna količina novca C koji se može skupiti, a da se pritom mogu koristiti 3 operacije.

Varijable zadatka:

- 1) vrijeme $T \leq 1000$
- 2) *learningRate* – skalira rezultat svake 'train' operacije
- 3) *paybackRate* – skalira rezultat svake 'teach' operacije
- 4) cijena svake od 4 knjiga

Operacije zadatka:

- 1) *teach* - traje 2 sekunde, a profit koji daje je određen formulom:
 $10 + \min(20, \text{knowledge}) * \text{paybackRate}$
- 2) *train* - košta 20\$, povećava *knowledge* za 1, a trajanje je određeno formulom:
 $\max(1, (\text{int})(8 / \max(1, \text{book} * \text{learningRate})))$
- 3) *buy* - *i*-to kupovanje se odnosi na cijenu *i*-te knjige, a trajanje je *i* sekundi
(i = 0, ..., 3 za pojedinu knjigu)

Zadatak se rješava dinamičkim programiranjem. Uočimo da je dovoljno da za određeno vrijeme, *knowledge* i zadnju knjigu koju smo kupili pamtimo samo najveću količinu novca koja se do tada može skupiti.

To će nam biti stanje, a prijelazi su tada definirani gore navedenim operacijama. Maksimalan broj stanja je $1000 * 20 * 4$, a tolika je i vremenska složenost jer je prijelaz složenosti $O(1)$. U nastavku se nalazi rješenje polaznika Matije Milišića:

```
#include <stdio>
#include <cstring>
#include <algorithm>

using namespace std;

int a[5];
int dp[1055][22][5];

inline void update( int &r, int c ) { r = max( r, c ); }

int main( void ) {
    int t, l, p;
    scanf( "%d %d %d", &t, &l, &p );
    for( int i = 0; i < 4; ++i )
        scanf( "%d", &a[i] );

    memset( dp, -1, sizeof( dp ) );
    dp[0][0][0] = 0;

    int ans = 0;
    for( int i = 0; i <= t; ++i )
        for( int j = 0; j <= 20; ++j )
            for( int k = 0; k <= 4; ++k ) {
                if( dp[i][j][k] == -1 ) continue;

                int c = dp[i][j][k];
                ans = max( ans, c );

                if( k < 4 && a[k] <= c ) update( dp[i+k][j][k+1], c - a[k] );
                if( c >= 20 ) {
                    int tt = max( 1, 8/max(1, k*l) );
                    update( dp[i+tt][j+1][k], c - 20 );
                }

                update( dp[i+2][j][k], c + 10 + j*p );
            }

    printf( "%d\n", ans );
    return 0;
}
```

Bicikl (ZATEMAS online baza zadataka):

U jednoj dalekoj državi treba napraviti stazu za biciklističku utrku. Država ima *N* gradova označenih brojevima od 1 do *N*. Jedini je uvjet da staza počinje i završava u gradu 1. U stazi se mogu ponavljati i gradovi i ceste (pa tako i grad 1). Staza treba prolaziti kroz što više gradova (svaki grad kroz koji prolazi brojimo samo jednom, bez obzira na broj prolazaka). Napišite program koji računa broj tih gradova.

Ograničenja:

n - broj gradova = manji od 300 000

k - broj cesta = manji od 300 000

k linija po 2 broja A i B , koji označavaju da postoji jednosmjerna cesta od A do B

Najjednostavnije rješenje je da se rekurzivno granamo na sve gradove i gledamo putove na kojima smo uspjeli doći do grada 1, ali to bi bilo presporo zbog veličine test primjera i previše cesta na koje bi se rekurzija granala.

Bolje rješenje je da jednom rekurzijom nađemo sve gradove do kojih možemo doći od grada 1, a onda "okrenemo" sve ceste i opet pustimo rekurziju iz grada 1. Time smo našli sve gradove do kojih možemo doći iz grada 1 i sve one iz kojih možemo doći do grada 1. Svi gradovi koje smo obišli u obje rekurzije se pribrajaju rješenju, jer ako možemo od grada 1 doći do nekog grada G , i ako možemo od tog istog grada G doći do grada 1, onda se taj grad nalazi na putu.

U nastavku se nalazi rješenje polaznika Mihaela Liskija:

```
#include <iostream>
#include <vector>

using namespace std;

vector<int> ve1[300010], ve2[300010];
int bio1[300010], bio2[300010], zbr=0;

void rek2 (int tr){
    if (bio2[tr]==1) return;
    bio2[tr]=1;
    if (bio1[tr]==1) zbr++;
    for (int i=0; i<ve2[tr].size(); i++){
        rek2(ve2[tr][i]);
    }
    return;
}

void rek1 (int tr){
    if (bio1[tr]==1) return;
    bio1[tr]=1;

    for (int i=0; i<ve1[tr].size(); i++){
        rek1(ve1[tr][i]);
    }

    return;
}

int main(){
    int n, k, p1, p2;

    cin >> n >> k;

    for (int i=0; i<k; i++){
        cin >> p1 >> p2;
        ve1[p1].push_back(p2);
        ve2[p2].push_back(p1);
    }

    rek1(1);
    rek2(1);

    cout << zbr << endl;

    return 0;
}
```

Asfalt (Srpska informatička olimpijada 2011)

Zadan je povezan neusmjeren graf od N čvorova i M bridova ($N \leq 20000$, $M \leq 200000$). Poznato je da svaki čvor n ima neku jedinstvenu, nama nepoznatu vrijednost $f(n)$ iz intervala $[1, N]$. Prema toj vrijednosti 'asfaltiramo' brid između čvorova a i b ako $f(b)$ maksimalnu vrijednost od svih susjeda a . Poznato nam je koji su bridovi asfaltirani, zanima nas koliko postoji čvorova n takvih da $f(n)$ može poprimiti vrijednost N , u nekom valjanom pobrojavanju. Također, garantirano je da postoji barem jedno rješenje.

Rješenje:

Primijetimo da svaki čvor osim onog s najvećim f ima točno jedan maksimum, stoga asfaltirani bridovi čine stablo. Ukorijenimo stablo u proizvoljnom čvoru n . Zanima nas je li moguće dodijeliti vrijednosti f čvorovima stabla tako da nigdje ne dođemo do kontradikcije. Sljedeći uvjet, uz onaj da postoji barem jedno rješenje, nužan je i dovoljan za provjeru pobrojivosti čvorova:

Ukoliko ne postoji neasfaltirani brid koji spaja čvor s njegovim pretkom u ukorijenjenom stablu, tada postoji rješenje u kojem funkcija f poprima maksimum u n .

Iz samog uvjeta proizlazi *brute-force* rješenje koje za korijen uzima svaki čvor i tada provjerava ispravnost istoga.

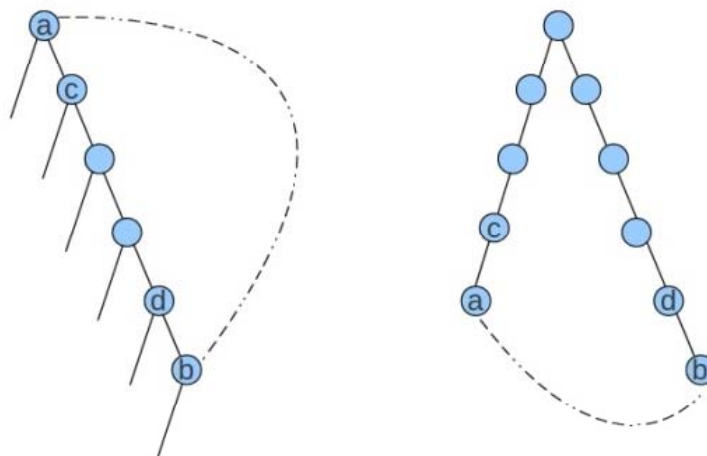
Sljedeći algoritam odgovara na nešto općenitije pitanje:

Ako ukorijenimo stablo u nekom čvoru n , koliko ima parova (čvor, predak(čvor)), dakle "loših" bridova koji su povezani neasfaltiranim bridom?

Pretpostavimo da nam je poznato rješenje za čvor 1. Tada izračunamo za sve bridove stabla promjenu tog broja ako prelazimo njime (primijetite da će taj broj imati različit predznak u ovisnosti o smjeru). Jednim DFS-om izračunamo ulazno (*discovery*) i izlazno (*finish*) vrijeme, na temelju kojeg možemo odgovarati na pitanja tipa: "Je li a predak od b ?" ili "Ako je a predak od b , koje dijete od a je također predak od b ?".

Prvo pitanje moguće je riješiti u složenosti $O(1)$, a drugo u složenosti $O(\log N)$. Uz brzu implementaciju odgovora na ova pitanja problem je moguće riješiti u složenosti $O(N + M \log N)$.

Razlikujemo dva slučaja (skica).



U prvom slučaju promatramo neasfaltirani brid koji spaja čvorove od kojih je jedan predak drugoga. Ako prelazimo iz čvora a u čvor c , broj "loših" bridova smanjuje se za 1. Ako prelazimo iz čvora d u čvor b , broj "loših" bridova povećava se za 1.

U drugom slučaju gledamo brid koji spaja čvorove od kojih nijedan nije predak nijednog. Ako prelazimo iz čvora a u čvor c , broj "loših" bridova smanjuje se za 1. Ako prelazimo iz čvora d u čvor b , broj "loših" bridova smanjuje se za 1.

Nakon izračunavanja tih "delta" vrijednosti na bridovima, problem lako riješimo jednim DFS-om.

Implementacija polaznika Gustava Matule:

```

#include <cstdio>
#include <cstring>

#include <vector>
#include <algorithm>

using namespace std;

#define MAXN 20005

int N, M;
int a, b, t;

int dad[ MAXN ];
int dis[ MAXN ];
int fin[ MAXN ];
int pivot, now;

int delta[ MAXN ];
vector< int > T[ MAXN ];
vector< pair< int, int > > E;

int cnt;
vector< int > Sol;

inline bool cmpf( int a, int b ) {
    if( a == pivot ) return 0;
    if( b == pivot ) return 1;
    if( dis[a] != dis[b] ) return ( dis[a] < dis[b] );
    return ( a < b );
}

void dfs0( int n ) {
    dis[n] = ++now;

    for( int i = 0; i < T[n].size(); ++i ) {
        if( T[n][i] == dad[n] ) continue;
        dad[T[n][i]] = n;
        dfs0( T[n][i] );
    }

    pivot = dad[n];
    sort( T[n].begin(), T[n].end(), cmpf );

    fin[n] = ++now;
}

inline bool is_ancestor( int a, int b ) {
    return ( dis[a] <= dis[b] && dis[b] <= fin[a] );
}

int fc;

inline void first_child( int a, int b ) {
    int lo = 0, hi = T[a].size()-1-( a != 1 ), mid;

    while( lo < hi ) {
        mid = ( lo + hi ) >> 1;
        if( fin[T[a][mid]] < dis[b] ) lo = mid+1;
        else hi = mid;
    }

    fc = T[a][lo];
}

void dfs1( int n ) {
    if( cnt == 0 ) Sol.push_back( n );
    for( int i = 0; i < T[n].size(); ++i ) {
        if( T[n][i] == dad[n] ) continue;
        cnt += delta[ T[n][i] ];
        dfs1( T[n][i] );
    }
}

```



```

    cnt -= delta[ T[n][i] ];
}
}

int main( void )
{
    scanf( "%d%d", &N, &M );

    for( int i = 0; i < M; ++i ) {
        scanf( "%d%d%d", &a, &b, &t );
        if( t == 0 ) E.push_back( make_pair( a, b ) );
        else {
            T[a].push_back( b );
            T[b].push_back( a );
        }
    }

    dfs0( 1 );

    for( int i = 0; i < E.size(); ++i ) {
        if( is_ancestor( E[i].first, E[i].second ) ) {
            first_child( E[i].first, E[i].second );
            ++delta[E[i].second];
            --delta[ fc ];
            ++cnt;
        }
        else if( is_ancestor( E[i].second, E[i].first ) ) {
            first_child( E[i].second, E[i].first );
            ++delta[E[i].first];
            --delta[ fc ];
            ++cnt;
        }
        else {
            ++delta[ E[i].first ];
            ++delta[ E[i].second ];
        }
    }

    dfs1( 1 );

    sort( Sol.begin(), Sol.end() );

    printf( "%d\n", (int)Sol.size() );

    for( int i = 0; i < Sol.size(); ++i )
        printf( "%d ", Sol[i] );

    putchar( '\n' );

    return 0;
}

```

Powerful Array, Codeforces

Zadan je niz a_i duljine N . Za zadani podniz $a_L, a_{L+1}, \dots, a_R$ potrebno je izračunati vrijednost sljedećeg izraza:

$$S = b_1 K_1^2 + b_2 K_2^2 + \dots + b_k K_k^2$$

gdje je $b_1, b_2, \dots, b_k$ skup različitih članova zadanog podniza, a K_i broj pojavljivanja broja b_i u tom podnizu. Potrebno je odgovoriti na Q takvih upita. $N, Q \leq 100000$, $a_i \leq 1000000$

Rješenje se temelji na tome da ćemo izračunati rezultat za jedan upit, a onda ga proširiti ili skratiti do sljedećeg upita. Tako obrađujemo sve upite po redu.

Primijetimo da, ukoliko trenutni obrađeni podniz proširimo ili skratimo za jedno polje, rješenje za novi podniz možemo izračunati u konstantnoj složenosti. Naime, kod takve transformacije mijenja se učestalost samo jednog elementa, za točno jedan, pa rezultat lagano ispravljamo. Budući da upite možemo obrađivati u proizvoljnom poretku, možemo odabrati takav poredak da je ukupan broj promjena reda veličine $O((N + Q) \cdot \sqrt{N})$.

Podijelit ćemo upite s obzirom na njihove početke u blokove veličine $\sqrt{N}$. Unutar svakog bloka upite sortiramo po desnome kraju, u rastućem poretku. Ovakvom podjelom garantiramo da će za svaki upit popravljanje lijevog kraja biti napravljeno u najviše $\sqrt{N}$ koraka, dok će za cijeli jedan blok ukupan broj popravljavanja desnog kraja biti najviše N . Kako blokova ima $\sqrt{N}$, a transformacija upita Q , ukupna složenost je $O((N + Q) \cdot \sqrt{N})$. Podjelu na blokove činimo na način da sve upite sortiramo po blokovima, pa po desnome kraju. Dodatnu optimizaciju možemo napraviti tako da parne blokove sortiramo uzlazno po kraju, a neparne silazno. Implementacija polaznika Ivice Kičića:

```
#include <cstdio>
#include <cstdlib>
#include <cstring>
#include <cmath>
#include <algorithm>

using namespace std;

typedef long long ll;

#define MAX 200004
#define VALUE 1000004

int niz[MAX];
int A[MAX], B[MAX], block[MAX];
int Q[MAX];
int N, M, sqrtN;
int X, Y;
int cnt[VALUE];
ll sol[MAX];
ll result;

inline bool cmp( int x, int y ) {
    return block[x] < block[y]
        || block[x] == block[y] && ( block[x] & 1 ? B[x] > B[y] : B[x] < B[y] );
}

inline void ADD( int x ) {
    result += (2 * (++cnt[niz[x]]) - 1) * (ll)niz[x];
}

inline void DEC( int x ) {
    result -= (2 * (--cnt[niz[x]]) + 1) * (ll)niz[x];
}

inline void move( int nx, int ny ) {
    while ( X > nx ) ADD( --X );
    while ( X < nx ) DEC( X++ );
    while ( Y < ny ) ADD( ++Y );
    while ( Y > ny ) DEC( Y-- );
}

int main(void) {
    scanf("%d%d", &N, &M);
    sqrtN = (int)(sqrt( N ) + 1);

    for ( int i = 1; i <= N; ++i )
        scanf("%d", niz + i);

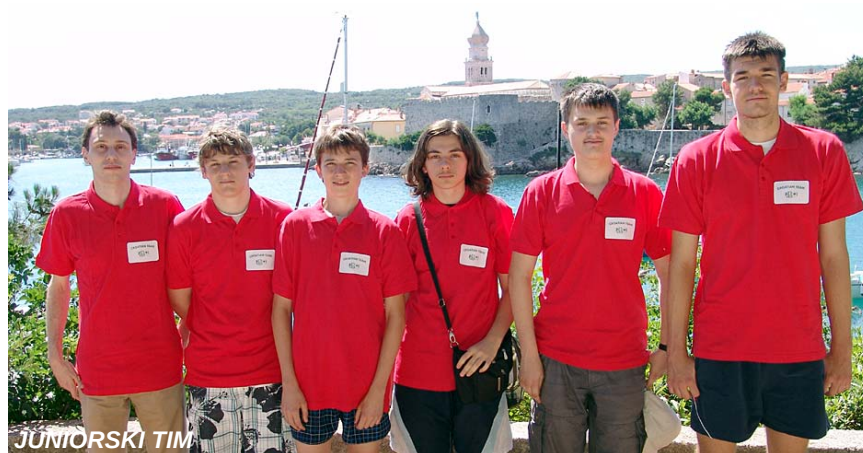
    for ( int i = 0; i < M; ++i ) {
        scanf("%d%d", A + i, B + i);
        block[i] = A[i] / sqrtN;
        Q[i] = i;
    }

    X = Y = 1; ADD( 1 );
    sort( Q, Q + M, cmp );
    for ( int k = 0; k < M; ++k ) {
        move( A[Q[k]], B[Q[k]] );
        sol[Q[k]] = result;
    }

    for ( int i = 0; i < M; ++i )
        printf("%I64d\n", sol[i]);
    return 0;
}
```

Hrvatski srednjoškolski tim u sastavu: Ivan Katanić, Ivica Kičić, Gustav Matula i Matija Milišić zastupat će Hrvatsku na 18. srednjoeuropskoj informatičkoj olimpijadi 2011 – Central European Olympiad in Informatics – CEOI 2011 u Poljskom gradu Gdinja, od 7. do 12. srpnja 2011. godine i na 23. međunarodnoj informatičkoj olimpijadi – International Olympiad in Informatics – IOI 2011 u gradu Pattaya City, Tajland, od 22. do 29. srpnja 2011. godine.

Juniorski tim u sastavu: Ivan Lazarić, Mihael Liskij, Mislav Balunović i Dominik Gleich nastupit će na 5. juniorskoj balkanskoj informatičkoj olimpijadi – Junior Balkan Olympiad in Informatics – JBOI 2011 u Rumunjskom gradu Bistrita, od 3. do 9. srpnja 2011., gdje je Hrvatska pozvana kao cijenjeni gost.



Smatram da su intenzivnim radom na Kampu, olimpijci dobro usvojili i provježbali potrebno gradivo koje se pojavljuje na međunarodnim natjecanjima. Nadamo se dobrim rezultatima!

Davor Prugovečki



21. KAMP MLADIH INFORMATIČARA HRVATSKE
Krk, 18. - 28. lipnja 2011.

BILTEN

Autori priloga i fotografija

Sudionici Kampa 2011.

Urednica

Vlasta Šeparović, dipl. ing.
vlasta@hsin.hr

Tehnički urednik

Aron Bohr

Likovno rješenje korica

Ivo Šeparović, dipl. ing.
Aron Bohr

Izdavač

Hrvatski savez informatičara
Zagreb, Dalmatinska 12, tel./fax 01/48 48 770
www.hsin.hr, e-mail: hsin@hsin.hr

Za izdavača odgovara

Ivo Šeparović, dipl. ing., tajnik HSIN-a
isepar@hsin.hr

Naklada

100 primjeraka

Tisak

Hrvatski savez informatičara

ORGANIZATOR



HRVATSKI SAVEZ INFORMATIČARA

SUORGANIZATORI



OŠ FRANA KRSTE FRANKOPANA KRK

K I O K
KLUB INFORMATIČARA
OTOKA KRKA

POKROVITELJSTVO I POTPORA



HRVATSKA ZAJEDNICA
TEHNIČKE KULTURE



GRAD KRK
OPĆINE I GRADOVI
OTOKA KRKA



**GLAVNI SPONZOR PROGRAMA
HSIN-a u 2011. GODINI**



HSIN
10000 ZAGREB, Dalmatinska 12, HRVATSKA
Telefon/fax: + 385/01/48 48 770, 48 48 768
e-mail: hsin@hsin.hr
www.hsin.hr