

Др Филип Марић

МАТЕМАТИКА У ОБРАЗОВАЊУ БУДУЋИХ ИТ СТРУЧЊАКА

Увод

У претходних десетак година присутан је веома значајан раст сектора развоја информационих технологија (тзв. ИТ сектора) у Србији. Вредност извоза софтверских производа, телекомуникација, рачунарских и информацијских услуга је током претходних пет година растао годишње за 20–25% и у 2019. години извоз је премашио милијарду и четиристо милиона евра. Због свега овога плате запослених у ИТ сектору су значајно изнад републичког просека. Наравно, развој ИТ сектора није значајан само за појединце који у њему учествују, већ и за друштво у целини, јер данас ИТ сектор значајно доприноси бруто друштвеном производу Србије.

Осим веће конкурентности услед нижих трошкова у односу на земље Западне Европе и Северне Америке (пре свега због ниже цене рада), изузетно значајан фактор успешног развоја ИТ сектора у Србији је традиција образовања на пољу математичких и техничких дисциплина. Универзитет у Београду је међу првих 500 на Шангајској листи, постоји значајан број добрих државних факултета широм земље на којима се школују програмери и постоје приватни факултети који такође нуде квалитетно образовање на овом пољу. Иако постоји значајан простор за осавремењавање и унапређивање наставе, традиционално средњошколско образовање у области природних и математичких наука (пре свега на природно-математичким смеровима у класичним гимназијама и у специјализованим школама попут Математичке гимназије) најбољим ученицима пружа јако добру основу за даље школовање и рад у свим областима природних, техничких и инжењерских дисциплина¹. Факултетско образовање сигурно није довољно за успешну каријеру у ИТ сектору и сигурно је потребно унапређивање практичних вештина кроз праксу и целоживотно учење. С друге стране, иако постоји значајан број запослених који успешно раде и без завршеног факултета, факултет јесте

¹С друге стране, проблем оваквог, донекле елитистичког приступа настави лежи у томе што велики број ученика, који није у стању да испрати предвиђени план и програм, прилично рано, често још у основној школи, у потпуности испадне из система и задржи се само на основном нивоу постигнућа, па чак и испод тога.

компаративна предност, јер пружа ширину знања коју појединац тешко може да стекне потпуно самостално, а у неким компанијама је и формалан услов за више и руководеће позиције.

Успешан развој ИТ сектора допринео је и увођењу обавезне наставе информатике и основа рачунарског програмирања у старије разреде основне школе, као и повећање број наставних тема из области програмирања и рачунарских наука у настави информатике и рачунарства у гимназијама. Иако су ови предмети конципирани као општеобразовни и посвећени су савладавању информатичких компетенција које могу бити значајне свим академским грађанима (на пример, и будућим лекарима и будућим правницима) они могу допринети повећаном опредељивању ученика за област информатике, рачунарства и програмирања.

Значај математичког образовања за бављење информатиком

Ако се по страни остави сегмент рачунарства у склопу наставе информатике, за развој способности потребних за успешно бављење професионалним програмирањем кључно је квалитетно математичко образовање. Учењем математике, ученици стичу важне вештине потребне за касније бављење рачунарством и програмирањем. С једне стране, математичким образовањем стичу се конкретна знања и развија се математички апарат који има директну примену у решавању практичних задатака, а с друге стране математичко образовање доприноси развоју општих когнитивних способности и развоју одговарајућег приступа решавању проблема које је корисно у свим информатичким дисциплинама. Наведимо само неколико примера концепата који се развијају у настави математике, а који су веома релевантни за бављење програмирањем.

Алгоритамски поступци

Рачунари нису интелигентне машине, већ само изводе низ корака које им програмери задају у облику програма који се извршава. Зато вештина програмирања представља вештину осмишљавања и потпуно прецизног описивања алгоритамских поступака. Такозвани алгоритамски начин размишљања (енгл. *algorithmic thinking, computational thinking*) важан је приступ решавању проблема у савременом свету.

Математика обилује алгоритамским поступцима и ученици се кроз наставу математике обилато срећу с њима. Већ од млађих разреда основне школе, када уче како да сабирају, одузимају, множе и деле бројеве „потписивањем“, ученици уче како да спроведу прецизне алгоритме. У старијим разредима ОШ ученици уче алгоритме за растављање природних бројева на просте чиниоце, проналажење НЗД и НЗС, сабирања и множења полинома итд, у средњој школи уче алгоритме дељења полинома итд. Осим ових „именованих“ алгоритама, већина бављења математиком подразумева спровођење неких алгоритамских поступака који помажу да се нови задаци реше. На пример, „алгоритам“ упрошћавања рационалних алгебарских израза укључује примену низа познатих алгебарских једнакости

(нпр. квадрат бинома, дистрибутивност, скраћивање разломка, . . .) и презаписивање израза док се не сведе на неки једноставнији облик, који се даље не може упрошћавати (на неку нормалну форму). „Алгоритам“ израчунавања одређеног интеграла укључује „алгоритам“ увођења смене у интеграл, „алгоритам“ парцијалне интеграције, „алгоритам“ примене Њутн-Лајбницевог формуле и слично. Дакле, ученици који добро владају математиком имају веома добро разумевање појма алгоритамског поступка и његовог прецизног и поступног извођења.

Формални језик

Иако су са развојем вештачке интелигенције рачунари постали способни да издвоје релевантне информације из текстова задатих на природном језику, програмирање и даље подразумева запис алгоритма на потпуно прецизним, програмским језицима (C, C++, C#, Java, Python, SQL, . . .), који имају веома стриктну синтаксу и прилично прецизно описану и семантику. Ученици се таквим обликом прецизног записа и са таквим вештачким, симболичким језицима први пут срећу у математици.

Још у млађим разредима основне школе учи се писање израза на основу текста проблемског задатка (тзв. решавање „текстуалних“ задатака). У самом почетку бављења математиком, кроз решавање једначина уводи се појам променљиве који означава непознате величине и од ученика се захтева формални запис релација задатих текстом задатка (није довољно да ученик само израчуна тражени резултат, већ се инсистира да коректно, симболички запише поступак израчунавања). Изрази и формуле, који представљају основу математичког језика се практично у непромењеном облику користе и у програмирању. Скоро да је немогуће замислити рачунарски програм у ком се не јавља неки израз којим се задаје израчунавање, тј. трансформација вредности променљивих (већ први програми које ученици пишу су такви да се читају неке улазне величине, на њих се примени одређена формула, записана математичким изразом и на крају се испише резултат).

Прецизност која се тражи у програмирању превазилази прецизност која се учи и захтева у математици, што може представљати значајан проблем на почетним нивоима програмерског образовања и што многе почетнике одбије од програмирања (један пропуштени размак или симбол ; доводи до тога да програм не може ни да се покрене).

Математичка логика

Иако се исказна логика (исказни везници) и предикатска логика првог реда (квантификатори и везане променљиве) формално уводе тек у првом разреду гимназије, они су у неформалном облику увелико присутни у настави математике у старијим разредима ОШ и ученици не могу да разумеју текст уџбеника ако овим појмовима не владају добро на интуитивном нивоу.

Програмирање захтева јако добро разумевање основних логичких појмова. На пример, рад са датумима, који је неизбежан у разним практичним апликацијама, подразумева и проверу да ли је година преступна. Дефиниција преступне године је да је то „она година која је дељива са 4, а није дељива са 100, осим ако није дељива са 400“. Запис овог услова у програмском језику, нпр. „(godina % 4

== 0 and godina % 100 != 0) or godina % 400 = 0“, захтева добро и дубоко разумевање исказне логике, конјункције и дисјункције. Даље, писање упита над базама података, што је поново неизбежан корак у већини савремених апликација захтева добро разумевање концепата логике првог реда и квантификације. На пример, један реалан упит може бити да се у школском информационом систему међу ученицима којима су закључене све оцене пронађу они који имају јединицу из неког предмета – универзална и егзистенцијална квантификација у овом услову је прилично очигледна.

Разлагање проблема на једноставније потпроблеме

Сваки рачунарски програм захтева коришћење (а често и креирање) хијерархије апстракција, посвећених решавању посебних, јасно дефинисаних подзадатака чијом се комбинацијом долази до решења оригиналног проблема. Ово разлагање проблема је изразито важно за савладавање комплексности, јер се комплекснији задаци могу решити само ако се разложе на једноставније задатке.

Један од основних механизма разлагања програма на мање целине је дефинисање засебних функција. Наиме, функције представљају централни појам како у математици, тако и у програмирању. На пример, све математичке формуле (нпр. формуле за израчунавање обима, површине и запремине) описују функционалне зависности резултата од улазних величина. С временом, изрази којима се описују израчунавања постају све комплекснији и осим основних аритметичких оператора почињу да обухватају и примену сложенијих функција (кореновање, степеновање, логаритми, тригонометријске функције). Сложенији изрази се граде композицијом једноставнијих. Рачунарски програми се често могу схватити као функције које на основу вредности улазних величина израчунавају жељене резултате. Сложене функције изражавају се комбиновањем једноставнијих функционалности, најчешће механизмом композиције – резултати примене једне функције постају улазни параметри наредне (ово је нарочито изражено у тзв. функционалном програмирању). На пример, у првом кораку функција сортирања уређује списак особа на основу њиховог ЈМБГ, а у другом, засебном кораку, примењује се више пута функција која алгоритмом бинарне претраге проверава да ли се на списку налазе неке конкретне особе.

Још један облик апстракције у програмирању представљају структуре података, које су често ефикасне програмске реализације одговарајућих математичких концепата. На пример, већина савремених програмских језика пружа програмерима структуру података која се назива скуп, која представља коначан скуп, и структуру података асоцијативни низ (каже се и речник или мапа), која представља програмску реализацију функције над неким коначним доменом (сваком елементу коначног домена придружена је нека конкретна вредност). На пример, ако у програму желимо да сваком студенту придружимо његов број поена, употребити бисмо асоцијативни низ који слика индекс студента у број поена, а ако бисмо желели да пребројимо колико различитих студената постоји у неком низу, креирали бисмо скуп студената и читали број његових елемената.

Израчунавање и процена вредности

Бављење рачунарством често захтева егзактно израчунавање или, још чешће, процену неких величина и количина. На пример, често је потребно проценити колико ће времена бити потребно неком програму да изврши одређено израчунавање, да ли ће сви подаци који ће се прикупити моћи да стану у расположиву меморију и слично. То често захтева вештину израчунавања коначних збирова и производа, сумирање конвергентних редова и слично. Примери који су веома једноставни, а често се срећу у пракси су израчунавање збира аритметичког и геометријског низа и реда.

С друге стране, за разлику од традиционалне математике где се скоро по правилу захтева на апсолутној прецизности, процене у примењеним дисциплинама могу да буду и мало грубље и да захтевају да се нека величина оцени са одређеном допуштеном грешком². На пример, да би ученици могли да процене колико се различитих вредности може представити помоћу 32 бита, потребно је проценити вредност 2^{32} . Уз знање да је $2^{10} = 1024$ приближно једнако $10^3 = 1000$ и елементарно познавање правила степеновања, лако се увиђа да је $2^{32} = 2^2 \cdot 2^{30} = 4 \cdot (2^{10})^3$, што је приближно једнако $4 \cdot (10^3)^3 = 4 \cdot 10^9$, тј. да је тражена вредност нешто мало више од четири милијарде. Слично се онда број корака потребних за бинарну претрагу сортираног низа од милион елемената или најмања висина бинарног дрвета у које може да се складишти милион елемената процењује као логаритам за основу 2 броја милион, што се поново, на основу инвентивне примене једноставних правила логаритмовања лако може проценити на око 20.

Иако се у традиционалној настави математике овакве процене ретко изучавају (без претходне припреме јако мали број студената Математичког факултета уме да процени вредност $\log_2 10^6$, јер то није тип проблема с којим се суочавају у настави математике), сав потребан математички апарат се обрађује у средњим школама, тако да је студенте потребно само додатно усмерити и указати им на такав специфичан начин размишљања (а наравно, сасвим би имало смисла да се процене величина изучавају и у основношколској и средњошколској настави математике). Сложеније процене овог типа укључују, на пример, примену диференцијалног и интегралног рачуна у израчунавању коначних сума, процену сума одређеним интегралима и слично, што су све јако важне математичке технике, које помажу решавању проблема значајних за рачунарство.

Рачунарска графика и геометрија

Област рачунарске графике је одувек била веома популарна грана рачунарства, пре свега због својих примена у дигиталној анимацији, али и у индустрији видео-игара. Рачунарска графика није могућа без доброг познавања геометрије, нарочито аналитичке геометрије. Сви објекти који креирају неку сцену тј. цртеж се представљају својим координатама и величинама (нпр. круг представљамо координатама његовог темена и величином полупречника). Познавање линеарне

²Познат је пример Енрика Фермија, чувеног физичара, који је својим студентима постављао питања попут оног да процене број лоптица за голф које могу да стану у авион Боинг 747 или да процене број клавир-штимера на свету и његове изреке да је боље умети тачно проценити неку величину, него је израчунати прецизно, али нетачно.

алгебре, вектора и матрица додатно олакшава и поједностављује све класичне задатке у рачунарској графичи. На пример, креирање анимације у којој се врши неко кружно кретање објеката на сцени се веома једноставно описује дејством матрица ротације (са правилно распоређеним синусима и косинусима угла ротације) на векторе координата тих објеката.

Анализа и обрада података

У савременом свету смо окружени подацима и велики изазов савременог друштва је да из мноштва података издвоји релевантне податке и информације на основу којих се могу извести закључци и донети корисне одлуке. Подаци се данас (зло)употребљавају у различитим доменима – од анализе ДНК секвенци и протеина, која је основа за синтетисање савремених лекова, до анализе понашања корисника на друштвеним мрежама и усмеравању политичких и маркетиншких кампања ка појединачним особама на основу података о њиховим активностима на интернету. ИТ гиганти, попут компанија Гугл и Фејсбук, направили су савремене пословне империје пре свега на чињеници да поседују, анализирају и користе податке милиона, па и милијарди својих корисника. Захваљујући учењу из података, направљени су алгоритми који препознају глас, аутоматски превode с једног на други језик, препознају слике и детектују објекте на њима и слично.

Наука о подацима (енгл. *data science*) је интердисциплинарна област која се фокусира на извлачење знања из велике количине структурираних, али и неструктурираних, неорганизованих података. У тесној је вези са областима истраживања података (енгл. *data mining*) и великих података (енгл. *big data*), али и са облашћу машинског учења (енгл. *machine learning*).

Све набројане области захтевају примену рачунарских алгоритама и програмирања, али и веома озбиљну примену математичких знања, пре свега из области вероватноће и статистике, као и нумеричке математике (нпр. области апроксимације). Није могуће проучавати податке ако се не зна шта је аритметичка средина, шта је стандардна девијација, шта је медијана, шта је случајна променљива, шта је математичко очекивање и слично. Многи математичари, нарочито из области статистике нашли су своја нове послове у области науке о подацима и често се говори да је практично нестала граница између савремене, примењене статистике с једне и науке о подацима и машинског учења с друге стране.

Јаче повезивање предмета и решавање конкретних проблема

Јака природна повезаност између математике, природних и техничких наука (укључујући и информатику и рачунарство) треба да буде присутна и видљива кроз веома јаке корелације између одговарајућих школских предмета. У многим образовним системима препозната је област *STEM* (енгл. *science, technology, engineering and mathematics*), која подразумева холистички приступ излагању математике, природних наука (биологије, хемије, физичке географије), техничких и инжењерских наука (електротехнике, машинства, грађевине, хардверског и

софтверског инжењерства и слично) и информатике и рачунарства. У Србији су први покушаји интеграције неких предмета из ове области учињени кроз увођење изборног предмета „Примењене науке“ у гимназији. Успостављање корелација наставе математике са другим STEM дисциплинама (у нашем образовном систему, то су пре свега информатика, физика, биологија и хемија) може допринети јачем мотивисању ученика и дубљем сагледавању свих корелираних области. Разумно је претпоставити да је синергија у овом случају више од простог збира. На пример, велики број проблема из области рачунарства није уопште могуће решити без познавања адекватног математичког знања и решавање математичких задатака традиционално представља одличан полигон за увежбавање програмирања и алгоритама. С друге стране, применом рачунара лакше се решавају многи математички проблеми и може се допринети много квалитетнијем разумевању одређених математичких појмова (пре свега кроз увођење визуелизације и експерименталне методе у наставу математике).

Иако свака од набројаних дисциплина има своје специфичности и веома је дискутабилно шта би потпуна интеграција донела, овај приступ може бити одлична допуна традиционалном наставном моделу, нарочито на часовима пројектне наставе. На интегрисаним часовима могуће је са ученицима решавати комплексније, конкретне, проблеме из реалног света. Наиме, у савременом свету, који се непрестано и брзо мења и постаје све сложенији, осим усвајања фундаменталних знања и вештина, важно је код ученика развити способност њихове примене на решавање проблема, који захтевају комбиновање знања из различитих области. Апстрактна теоријска знања могу се боље разумети и продубити ако се сагледају кроз призму својих конкретних примена.

Традиционална настава математике често бива „шаблонизирана“ и нагласак се ставља само на израчунавање, тј. на спровођење алгоритама и то најчешће потпуно ручно. На пример, ученици решавају системе од три једначине с три непознате или рачунају детерминанте димензије четири с конкретним бројевима, „пешке“, чак без употребе најобичнијих калкулатора. Корелације са програмирањем и информатиком омогућавају да се већина таквих рутинских послова аутоматизује, па је могуће решавати и комплексније проблеме чије решавање без употребе рачунара не би било могуће или би захтевало превише труда и времена. На пример, уз употребу рачунара могуће је решавати системе линеарних једначина с неколико десетина непознатих и једначина, што се може употребити у изградњи неког прилично прецизног модела проблема из реалног живота, који се моделује таквим системима.

У традиционалној настави, алгоритми се често описују само имплицитно (обично кроз низ урађених примера и решених задатака) и од ученика се ретко тражи да прецизно опишу алгоритам, већ је углавном довољно само да умеју да га спроведу. У проблемски оријентисаној настави, описивање алгоритама постаје централна вештина и уместо да само примењује унапред задате поступке, од ученика се очекује да сам формулише алгоритам решавања проблема, изведе све потребне формуле (комбиновањем формула које је раније срео), па чак и да на крају тај алгоритам испрограмира и претвори у функционални систем.

У традиционалној настави средњошколцима се углавном даје потпуно апстрактна формулација задатка, па се код њих не развија довољно способност изградње математичког модела. У збиркама задатака скоро да нема више „текстуалних задатака“ (какви су карактеристични за основну школу, нарочито у млађим разредима) и формулације су углавном потпуно апстрактне (на пример, „Израчунај вредност израза“, „Упрости израз“, „Реши систем“, „Израчунај лимес“, „Нађи извод“, „Израчунај интеграл“ и слично). У проблемски оријентисаној настави, математичко моделовање проблема, такође постаје централни део задатка. Од ученика се очекује да разуме које су улазне величине битне за решавање проблема, и да препозна односе између улазних величина и резултата и моделује их на одређени начин. На пример, од ученика се може тражити да конструише функцију која израчунава резултате на основу улазних величина, или да опише везе између познатих и непознатих вредности у облику једначина чијим ће се решавањем пронаћи тражена решења. Такође, од ученика се тражи да препозна који математички апарат му може помоћи у решавању конкретног задатка, што је прилично некарактеристично у традиционалној настави, где су сви задаци у збиркама јасно подељени по областима и где се тешко могу наћи задаци који захтевају од ученика да комбинује више техника решавања. Проблемски оријентисана настава инсистира на практичној употребљивости уведених концепата на решавање конкретних проблема. На пример, уместо механике ручног израчунавања детерминанти, много важније постаје то да ученик може да препозна да је детерминанта мера површине, тј. запремине паралелограма, односно паралелепипеда који образују њени вектори колона и да онда израчунавање површина или запремина уме да сведе на проблем израчунавања детерминанти.

Наравно, све ово ставља много већи изазов пред ученике (али и наставнике) него у случају када се решавају само апстрактни задаци, изоловани од општег контекста примене и специфично припремљени само у циљу увежбавања неке једне конкретне технике. То сигурно значи да ће слабији ученици имати потешкоћа у оваквим активностима. С друге стране, рад у тиму, вршњачко подучавање и свест да се у склопу формалног образовања решавају конкретни проблеми, чији смисао ученици одмах разумеју може довести до веће мотивације ученика, па тиме и до бољих резултата.

Закључак

Квалитетно математичко образовање и разумевање основних математичких појмова, као и математичке прецизности у изражавању представља добру основу за бављење рачунарским наукама и програмирањем и оно мора у великом обиму, с великим фондом часова бити заступљено у основним школама, гимназијама природно-математичког смера и специјализованим одељењима профилисаним као информатици (у гимназијама и у техничким школама). Захваљујући дугој традицији математичког образовања у Србији у основним и средњим школама, која добрим ђацима, који успеју да савладају широк дијапазон уведених апстрактних математичких концепата, у великој мери постоје предуслови потребни за даље

школовање и професионални развој будућих стручњака у области програмирања, рада са подацима, машинског учења, вештачке интелигенције и слично. Евентуална унапређивања образовног процеса требало би да традиционални модел наставе допуне интегрисаном наставом из више STEM области, где би фокус био на решавању конкретних проблемских задатака из свакодневног живота, математичком моделовању, алгоритамском размишљању и интеграцији и примени тренутно веома фрагментисаних, апстрактних и теоријских знања.

Математички факултет, Београд

E-mail: filip@matf.bg.ac.rs