

Izdelava in reševanje domačih nalog pri računalniškem programiranju

Creating and Doing Homework at Computer Programming

Gašper Strniša

Tehniški šolski center Kranj, Kidričeva 55, 4000 Kranj,
gasper.strnisa@gmail.com

Povzetek

Prvi koraki pri učenju računalniškega programiranja, so za marsikoga zelo naporni, saj je ta proces sestavljen tako iz logičnega razmišljanja, kot tudi faktografskega učenja. Za uspešno osvojitve novih znanj, je potrebno precej motivacije in ustreznih nalog (vaj), ki pomagajo pri razumevanju celotnega koncepta. V prispevku je predstavljeno problemsko stanje učenja in poučevanja programiranja v srednjih šolah, ter teoretična izhodišča, predlog rešitve in možnost za nadgradnjo omenjenih rešitev, ki dvigujejo nivo znanja programiranja naših dijakov.

Ključne besede: računalniško programiranje, domače naloge, motivacija, dijaki, življenjske spretnosti

Abstract

The first steps in learning computer programming can be very difficult for many students since the process is a combination of logical thinking and factographic knowledge. To gain new knowledge successfully a lot of motivation and appropriate tasks (exercises) are needed to help students fully understand the concept as a whole. In the paper the problems of learning and teaching computer programming in high schools are discussed together with the theoretical frameworks. A solution to the problem and an upgrade to the already existing solutions raising the level of computer programming knowledge of our students are also suggested.

Keywords: computer programming, homework, motivation, students, life skills

1 Uvod

Učenje programiranja marsikomu povzroča velike težave. Gre namreč za učni proces, ki je sestavljen tako iz logičnega razmišljanja, kot tudi faktografskega učenja. Bistvo

programiranja je torej v reševanju določenih problemov, katerih rešitve se predstavi s sintakso enega izmed množice programskih jezikov.

Sintaksa programskega jezika je tista, ki se jo je sposoben naučiti prav vsak. Večji problem predstavlja njena uporaba, v pravilnem in smiselnem zaporedju, še zlasti ko gre za reševanje konkretnih primerov iz realnega življenja. Takšni primeri največkrat predstavljajo odmik od šolskih primerov, pri katerih se bodoče programerje seznanja z najenostavnejšimi vajami, ki so prilagojene osnovnemu razumevanju programiranja.

Izziv pedagoškega procesa pri poučevanju programiranja, je ustvariti premik od faktografskih znanj k ustvarjalnemu reševanju kompleksnih primerov.

2 Opis problemskega stanja

Resnost problematike, se pri učenju oziroma poučevanju programiranja v srednji šoli, konkretno v tehniški gimnaziji – smer računalništvo, pokaže v primeru, ko za omenjeni predmet sploh ne obstaja nobenega učbenika. Osnovo učenja programiranja, bi zlahka primerjali z učenjem matematike, kjer je poleg razumevanja teoretičnih načel, potrebno narediti tudi veliko vaj za utrjevanje snovi in kompleksnejši pregled nad določenimi problemi. Kako naj se torej dijaki naučijo dobro programirati, če sploh nimajo nobenih vaj, katere bi reševali?

Seveda morajo za zgoraj opisani problem odgovornost prevzeti učitelji, ki naj bi pripravili zadostno količino ustreznih vaj, katere bi potem posredovali svojim dijakom. V teh primerih morajo gradiva pripraviti učitelji sami, za kar pa porabijo ogromno časa. Glede na to, da morajo na tak način učitelji pripraviti gradiva za več predmetov se lahko zgodi, da za pripravo porabijo večino svojega prostega časa v kolikor ne želijo, da bi drastično padla kakovost njihovih predavanj (Strniša, 2009). Nov problem se pojavi, ker se znanje programiranja (v kolikor si dijaki izberejo računalništvo kot izbirni maturitetni predmet), preverja tudi na maturi. Kako naj torej posamezni učitelji, na nivoju cele države, zagotovijo enako mero znanja vsem dijakom?

Velik napredek v tej smeri sicer predstavljajo elektronska gradiva, ki pa imajo tako prednosti, kot tudi svoje slabosti. Največja opažena slabost je v nepovezanosti vsebin med posameznimi poglavji, saj so le te prispevali različni avtorji, ne da bi se ozirali eden na drugega.

3 Teoretična izhodišča

3.1 Življenjske spretnosti in informacije

Razlikovati je potrebno med poučevanjem življenjskih spretnosti, ki so skoraj vedno uporabne, in posredovanjem informacij, ki so uporabne le, če učenci uvidijo njihovo vrednost ali če jih lahko učitelji prepričajo, da si jih je vredno vtisniti v spomin. Razlika med spretnostmi in informacijami je jasna: spretnosti so to, kar uporabljate, informacije so to, kar veste. (Glasser, 1994:a)

Za vzbuditev zanimanja dijakov in motivacijo tako pri sodelovanju kot nadaljnjem raziskovanju oz. učenju, učitelji najlažje dosežejo s povezavo obravnavane snovi in konkretnih

primerov iz realnega življenja. Problem se pojavi le, ker dijaki in učitelji omenjenim primerom postavljajo različno uporabno vrednost, zaradi česar slednji v dobri veri po zanimivih predavanjih ne dosežejo pravega učinka. Tako je naloga učitelja, da učne snovi ne prilagaja primerom iz prakse, kateri se zdijo zanimivi njemu, temveč da ugotovi interes dijakov in na tej osnovi gradi izvedbo učnih vsebin (Strniša, 2010).

Učenci so tisti, ki končno presojujejo, kako pomembna je ta informacija zanje. Če jo ocenjujejo kot pomembno, bodo na čim boljši način naredili, kar se od njih zahteva. (Glasser, 1994:b)

3.2 Motivacija za učenje

Učenci so marsikaj sposobni narediti brez učiteljeve pomoči. Predvsem si moramo biti povsem na jasnem, kje so meje učenčeve volje do dela, vedoželjnosti in zanimanja in kje se začnejo ambicije staršev in učiteljev. Če učenci niso preobremenjeni, radi sprejemajo zadolžitve zunaj predpisanega učnega načrta (Kunaver, 1992).

Ustvarjalnost učencev temelji na notranji motivaciji, na zanimanju, lahko bi rekli »na notranjem ognju«. Zunanja motivacija (pohvale, nagrade, kazni) je lahko začetna spodbuda, ki pa sama ne zadošča. Ustvarjalni učitelj skuša navdušiti učence za nove vidike, za »igro idej«, ki naj postane učenčev prevladujoči stil razmišljanja. (Likar et al., 2004).

Zakaj torej ne bi dali dijakom manj suhoparnih vaj iz šolskih učbenikov in več raziskovalnih, dolgoročnejših zaposlitev, ki bi jih pritegnile?

3.3 Domače naloge

Dhoritijeva pravi, da domača naloga pri večini učencev vzbudi misel na monotono pisanje vaj in guljenja. Domače študijsko okolje je pri večini učencev siromašnejše in je lahko celo antiproduktivno. Zato moramo izredno paziti, da se učenčevo navdušenje, ki smo ga dosegli v razredu, doma ne izrodi zaradi mehničnega in zato tudi dolgotrajnega izvajanja domačih nalog.

Domače naloge so včasih tako duhomorne, da bi še odraslega povsem zamorile, kaj šele otroka. Prav zato na začetku ni primerno dajati veliko domačih nalog, ker utegnejo bolj demotivirati učence kot jim pomagati, saj od njih običajno zahtevajo kompleksnejša znanja, kot jih na tej stopnji od njih sploh lahko pričakujemo. (Dhority, 1992)

4 Izdelava in reševanje lastnih domačih nalog

Glede na opaženo problemsko stanje in znana teoretična izhodišča, je eden od načinov za lažje učenje in boljše razumevanje programiranja v izdelavi in reševanju lastnih domačih nalog. Izdelava lastne domače naloge pomeni, da morajo dijaki sami napisati besedilno nalogo, katero potem realizirajo v programskem jeziku. Na ta način so primorani razmišljati o celotnem konceptu naloge, kar jim bo nekoč zagotovo koristilo. Prav gotovo se bo vsak, ki se bo odločil za delo programerja, nekoč srečal s naročniki, ki mu bodo zgolj opisali svoje želje za programe. Ob tem je zelo pomembno, da si programer sam podrobno napiše »nalogo« in jo potem tudi ustrezno realizira.

V letošnjem šolskem letu, se je ta način izkazal kot zelo uspešen, saj so dijaki tretjega letnika tehniške gimnazije, sami morali napisati lastne domače naloge in jih nato še rešiti. Glede na

primerjano skupino dijakov, ki so tretji letnik obiskovali preteklo leto, so se rezultati ob preverjanjih znanja močno popravili.

Takšna obveznost s strani dijakov, je zagotovo dobrodošla tudi za učitelja, ki programiranje poučuje. S pregledom besedilnih nalog, ki so jih dijaki pripravili, je namreč mogoče pridobiti pregled nad temami, katere dijake dejansko zanimajo. Svoja predavanja v prihodnje torej lahko prilagaja primerom, kateri se zdijo zanimivi dijakom in na tej osnovi gradi izvedbo učnih vsebin. S tem doseže vzbuditev zanimanja dijakov in motivacijo tako pri sodelovanju kot nadaljnjem raziskovanju oz. učenju.

5 Nadgradnja omenjene rešitve

Seveda bi se zgolj izdelavo in reševanje lastnih domačih nalog dalo še nekoliko nadgraditi. Ideja je predvsem v dodatni motivaciji dijakov, katero bi lahko dosegli na več različnih načinov. Prva novost, ki bo uvedena v novem šolskem letu, bo objava domačih nalog na Googlovih dokumentih, kamor bo vsak dijak moral naložiti svoje domače naloge. Naložili jih bodo pod skupnim uporabniškim računom, katerega že uporabljajo za prejemanje e-poštnih sporočil, namenjenih njihovem razredu. Na ta način, bodo lahko vsi videli dela svojih sošolcev, s čimer se bo ustvarila velika baza nalog. Druga pozitivna lastnost takšnih objav je v različnih stilih zapisov nalog. Dijaki bodo na ta način dobili različna navodila, ki ne bodo zapisana le v stilu, kot ga uporablja učitelj.

Kot motivacijski faktor, bodo najboljše naloge objavljene na Facebook profilu, imenovanemu RSO Profesor, ki je bil ustvarjen kot didaktično orodje (Strniša, 2011). Na tem profilu bo poleg same naloge objavljen tudi njen avtor, kar naj bi med dijaki spodbudilo tekmovalnost pri pisanju kakovostnih domačih nalog. Slika 1 prikazuje objavo domače naloge dijaka Nejca T., na Facebook profilu RSO Profesor.

Facebook ponuja tudi mnogo možnosti, ki bodo objavam predstavljale dodano vrednost. Najboljša izmed njih je zagotovo podajanje komentarjev na objavo. Tukaj bodo tako dijaki, kot tudi učitelj lahko dodatno diskutirali o rešitvah naloge, dodatnih možnostih ali pa o morebitnih napakah. Potem so tu še možnosti: Všeč mi je, Deli z ostalimi, Označevanje avtorja,...

facebook

Iskanje

Domov

Profil

Poišči prijatelje

Uporabniški račun

TEHNIŠKI ŠOLSKI CENTER KRANJ

ZA POKLICE PRIHODNOSTI

Brskaj po zapiskih

Zapiski prijateljev

Zapiski o straneh

Moji zapiski

Moji osnutki

Zapiski o meni

Skoči k Prijatelju ali na Stran

Add tags

Naroči se

Moji zapiski

Uredi nastavitve uvažanja

Program za vodenje hokejskih evidenc (Nejc T.)

Urejanje

Rso Profesor, 28. junij 2011 ob 22:11

Napišite program, v katerem ustvarite razred Hokej. Vsebuje naj lastnost "Država", ki ji preko konstruktorja določite poljubno vrednost. Vsebuje naj tudi metodo, ki bo vrnila povprečno število golov. Preko argumentov, boste vanjo vnesli gole in število odigranih tekem posameznega igralca.

Sedaj ustvarite razred HokejNaLedu, ki deduje lastnosti iz prejšnje naloge. Kot svojo lastnost mu določimo "Prva liga", saj nas zanimajo zgolj prvoligaška tekmovanja posameznih držav. Vsebuje naj tudi lastno metodo za izpis države, izpiše pa naj tudi klub igralca, katerega boste vpisali preko argumenta.

Na koncu ustvarite še razred Izpis, v katerega zapišete glavno metodo in ustvarite 3 objekte za konkretne igralce. Z vnosom podatkov o igralcih zaključite program, ki bo vrnil smiselne izpise.



Všeč mi je · Komentiraj · Deli z ostalimi · Brisanje

Slika 1: Facebook profil RSO Profesor, z objavljeno nalogo dijaka Nejca T.

6 Zaključek

Izdelava lastnih domačih nalog je pri računalniškem programiranju zagotovo smiselna, saj si na ta način dijaki sami postavijo problem, ki ga nato poizkušajo rešiti. V kolikor se jim pri tem zatakne, bodo verjetno bolj intenzivno iskali rešitev, kot če bi jim bila naloga vsiljena in bi se že pri prvi težavi zaradi pomanjkanja motivacije ustavili. Z reševanjem lastnih nalog, je odpravljena tudi možnost prepisovanja od sošolcev, saj je malo verjetno, da sta si dva zamislila enako nalogo.

Pozitivno spodbujanje tekmovalnosti, ki za nagrado nima številčne vrednosti v redovalnici, pač pa objavo naloge dijaka na učiteljevem Facebook profilu, bi prav tako moralo stimulatивно vplivati na toleranten odnos do izdelave in reševanja domačih nalog. Prvi ugodni rezultati med primerjanimi skupinami so se že pokazali, prav to pa je gonilo za nadaljnje raziskovanje učnih načinov in metod, ter uvedbo novih idej, ki naj bi dvignili nivo znanja dijakov.

359

Viri in literatura:

- Dhority, L. (1992): Ustvarjalne metode učenja. Umetnost uporabe sugestije za celostno učenje, Alpha Center, Ljubljana.
- Glasser, W. (1994:a): Mlademu učitelju na pot, Regionalni izobraževalni center, Radovljica.
- Glasser, W. (1994:b): Dobra šola – Vodenje učencev brez prisile, Regionalni izobraževalni center, Radovljica.
- Likar, B. [et. al.], (2004): Inovativnost v šoli : [od ustvarjalnega poučevanja do inovativnosti in podjetnosti], Inštitut za inovativnost in tehnologijo - Korona plus, Ljubljana.
- Kunaver, D. (1992): Učitelj v dobri šoli, Zavod Republike Slovenije za šolstvo in šport, Ljubljana.
- Strniša, G. (2009): Uporaba e-gradiv za nove srednješolske učitelje. Vzgoja in izobraževanje v informacijski družbi – Zbornik 12. mednarodne multikonference. Uredili: Rajkovič, V. [et. al.], Ljubljana 16. oktober 2009. Ministrstvo za šolstvo in šport [et. al.], Ljubljana.
- Strniša, G. (2010). Poker z vidika učenja in poučevanja pri predmetu računalništva. V: Zbornik mednarodne konference Splet izobraževanja in raziskovanja z IKT – SIRIKT. (Kranjska gora, 14. – 17. april 2010), Lenarčič, A. Miška, Ljubljana.
- Strniša, G. (2011). Facebook kot didaktično orodje. V: Zbornik mednarodne konference Splet izobraževanja in raziskovanja z IKT – SIRIKT. (Kranjska gora, 13. – 16. april 2011), Lenarčič, A. Miška, Ljubljana.