

UVOD

Ciljevi tehničkog i tehnološkog obrazovanja odražavaju zainteresovanost društva za naučnu i tehničku pismenost, za društveno-ekonomski razvoj, za pripremanje ljudi za široku primjenu naučnih dostignuća i za obučavanje onih koji će ta dostignuća dalje razvijati i usavršavati.

Samostalni rad učenika uz pomoć nastavnika, jedan je od osnovnih ciljeva savremene nastave. Učenika treba postepeno voditi do samostalnog rada, sistematski i kontinuirano ga osposobljavati. Samostalan rad se može definisati kao organizovana zaposlenost učenika u kontrolisanim uslovima na rješavanju postavljenih zadataka.

Individualna nastava predstavlja najbolje rješenje za izvođenje nastave. Vršiti se izbor zadataka (sadržaja) koji najviše odgovaraju određenom učeniku. Individualizovati nastavu, znači, orijentisati se na realne tipove učenika, uzeti u obzir razlike među njima, uskladiti ih, i mijenjati metode i postupke pedagoškog djelovanja prema tim razlikama, omogućiti učenicima da napreduju prema svom tempu i mogućnostima.

Softver koji se koristi u oblasti obrazovanja predstavlja savremeni način učenja, i naziva se obrazovni računarski softver. On obuhvata programske jezike i alate, određenu organizaciju nastave i učenja, koja se bazira na logici i pedagogiji. Pod obrazovnim računarskim softverom odrazumjevaju se računarski programi koji se mogu koristiti u okviru nastave koji pomažu i usmjeravaju u fazama individualne nastave.

Modelovanje i simulacija su značajni faktori efikasnosti nastave i učenja, posebno u nastavi informatike. Uzimajući u obzir da je broj časova za obradu nastavnih sadržaja minimalan, da treba dominirati praktičan rad i vježbe, kombinujući ih sa davanjem potrebnih teoretskih znanja za predviđeni razred, i da učenik treba da usvoji nastavno gradivo, može se zaključiti da je primjena modela i simulacije u nastavi i procesu učenja veoma važna.

Zato je važno da već sad počnemo pedagoški, metodički i nastavno oblikovati materijale kroz koje ćemo te mlade ljude odgajati i educirati za budućnost temeljenu na informacijsko komunikacijskoj tehnologiji.

Nastava je planski i organizovan proces odgoja i učenja koje provodi stručna osoba radi sticanja znanja, sposobnosti, vještina i navika prema utvrđenim nastavnim planovima i programima u odgovarajućem prostoru i sa odgovarajućom tehnologijom. Nastava je organizirani institucionalni i vaninstitucionalni interakcijski stvaralački napor u savlađivanju propisanih sadržaja i sticanja znanja, sposobnosti i navika. Ona doprinosi osposobljavanju za dalji permanentni rad s namjerom da se razvije kritička, slobodna, autonomna, stvaralačka, humana, obrazovna, multikulturna, multietnička, i multikonfesionalna učenikova ličnost koja je otvorena za sve sadašnje i buduće progresivne promjene u sebi i društvu (sredini) u kojem živi. Nastava je osnovna odrednica didaktike.

Zadatak izučavanja informatike u osnovnoj školi je pružiti učenicima/učenicama osnovna znanja o informatičkim pojmovima, naučiti ih osnove korištenja računara, osnove programiranja i što je naročito važno, naučiti ih da stečena znanja primjene u svakodnevnom životu.

1. NASTAVNI PLAN

Naš cilj kao vaspitno-obrazovne institucije jeste upravo taj, da što više djece motivišemo da pokrenu svoju kreativnost, da se pronadu i usmjere i prodube svoja interesovanja prema određenim oblastima života na zemlji. Nastava informatike je u startu u prednosti nad ostalim predmetima, jer nudi nov pristup učenju (individualni-“ja i kompjuter“, a ne samo frontalni-“mi protiv nastavnika“), nudi šarene i lijepe sadržaje koji mogu da privuku i zadrže dečiju pažnju. Zato treba posvetiti veliku pažnju oblikovanju sadržaja koji se nude učenicima, kroz nastavni plan i program. Postojeći planovi i propisane teme nisu toliko zastarjeli, koliko bi mogli da budu bolje raspoređeni. Veoma je teško u 7. razredu na primjer za predviđenih 35 časova, kvalitetno ispredavati predviđeno: Hardware, OS Windows, MS Word, MS Excel, Internet. Ovaj set programa se smatra osnovnom kompjuterskom obukom i potrebnom podlogom za dalje bavljenje računarima. Takođe se postavlja problem koji programski jezik uvesti u 8 razred, da li baš Qbasic, da li uopšte uvoditi svima programiranje? Basic je upravo pravi pedagoški alat za učenje programiranja i ulaženje u taj svijet. Njegova struktura programa je lijepa, jednostavna, školska i veoma logična (početak, tijelo programa i kraj). Basic ima takođe programske struktura (linijsku, razgranatu, cikličnu) koje su veoma logične, lako shvatljive i ako se objasne na dobrim i ilustrativnim primerima ostaju u glavi za cijeli život. Basic jeste dobar za početak i sigurno ga treba proučavati, ali samo kao uvod u Delphi, Visual Basic ili neko drugo, kompleksnije, aktuelno objektno programiranje. Jednostavno treba odvojiti nivoe znanja na korisnički i stvaralački-programerski. Kod programiranja mislim i na samo programiranje u Visual paketima i na bavljenje multimedijom, koja takođe iziskuje poznavanje logike programiranja.

U radu sa učenicima uzrasta od 11 do 15 godina može se izdvojiti iskustvo da postoje pojedinci koji realno veoma lijepo i lahko usvajaju sadržaje osnovne programskog jezika i programiranje u Qbasic-u, ali mi moramo uzeti u obzir prosječnog učenika koji nije, statistički gledano, posebno zainteresovan za te sadržaje i čije intelektualne mogućnosti i nivo svjesnosti ovog uzrasta možda ne dozvoljavaju i ne stimulišu brzinu usvajanja gradiva i mogućnost logičkog povezivanja naučenog u cjelinu. Uzrast od 14-15 godina, odnosno osmi i deveti razred osnovne škole dozvoljava, u prosjeku gledano da se početni sadržaji programskog jezika Qbasic-a usvajaju sa finom dinamikom i bez problema.

U tom uzrastu učenici su se već susreli sa kompjuterom i već imaju zauzet stav. Plan i program nastave koji se pred njih iznosi često ih dovodi u situaciju da se razočaraju i dosađuju na časovima, smatrajući da to što se prezentuje nije vrijedno njihove pažnje i vremena. Pa o čemu pričamo, pa ja sve to znam! ! !. U ovoj situaciji je nastavnik prepušten sebi i na njemu je da, ulažući nepotrebno dodatni napor, nastavu prezentuje na način koji će ipak zadržati i takve učenike.

NASTAVNI PLAN I PROGRAM za OSMI I DEVETI razred devetogodišnje osnovne škole (Programiranje u Basic-u)

SADRŽAJ	ZNANJE I RAZUMIJEVANJE	SPOSOBNOSTI	VRIJEDNOSTI, STAVOVI, PONAŠANJE	AKTIVNOST UČENIKA	AKTIVNOST NASTAVNIKA
PROGRAMIRANJE	<i>Nedostaci govornog</i>	Učenici su sposobni	Shvataju značaj	Uključuju se u	Izlaže nastavno

PRIMJENA SOFTVERSKIH MODELA U NASTAVI INFORMATIKE-PROGRAMIRANJE U QBASIC-u

Osnovne karakteristike programskih jezika. Proceduralni programski jezici. Objektno orijentirani programski jezici. Neproceduralni programski jezici Metodologija programiranja. Proceduralno i neproceduralno komuniciranje sa računarom. Razvoj proceduralnog komuniciranja. Struktuirani pristup razvoju programa.	jezika za potrebe komuniciranja sa računarom. Programski jezici <i>niskog nivoa</i>. <i>Proceduralni (problemski orijentirane)</i> programski jezici (BASIC, Pascal, FORTRAN, COBOL, C itd). <i>Objektno orijentirani</i> programski jezici: C++, Java, Object Pascal. <i>Neproceduralni</i> (deklarativni, opisni) programski jezici. <i>Kompajleri i interpreteri.</i> <i>Proceduralno i neproceduralno komuniciranje sa računarom.</i> <i>Razvoj proceduralnog komuniciranja.</i>	objasniti osnovne karakteristike programskih jezika. Sposobni su praviti razlike između vrsta (grupa) programskih jezika: jezici niskog nivoa (mašinski jezik), proceduralnih, objektno orijentiranih i neproceduralnih programskih jezika. <i>Sposobni su objasniti struktuirani pristup razvoju programa.</i>	programiranja sa aspekta rješavanja mnogih problema koje čovjek bez računara vjerovatno nebi nikada riješio. Posebno se interesuju za savremene programske jezike koji su od praktičnog značaja.	diskusije o programskim jezicima. Analiziraju razlike između kompajlera i interpretera. Postavljaju pitanja u vezi sa struktuiranim pristupom razvoju programa, te u kojim programskim jezicima se primjenjuje ovaj pristup.	gradivo i povezuje ga sa gradivom o programiranju koje je obrađeno u šestom razredu. Vodi računa da se ponavlja samo ono gradivo koje je potrebno za razumijevanje ovog nastavnog gradiva. Organizuje vježbe i prati u kojoj mjeri učenici izvršavaju zadatke.
Programski jezik BASIC. Osnovni elementi BASIC-a Osnovne naredbe BASIC-a. Programske strukture.	Razumiju načine rada u BASIC-u (direktni i indirektni). <i>Alfabet BASIC-a. Tipovi podataka.</i> <i>Konstante, promjenljive i izrazi.</i> Struktura ponavljanja. FOR petlja, IF THEN i GOTO.	Sposobni su pokrenuti program BASIC Sposobni su primijeniti naredbe koje pripadaju strukturi ponavljanja.	Primijenjuju verziju BASIC-a koja im je na raspolaganju.	Interesuju se kaje probleme je moguće riješiti koristeći BASIC. Uredno vode pribilješke. Izvođe predviđene vježbe	Izlaže nastavno gradivo. Nastavnu građu izlaže na način kojim će kod učenika izazvati pažnju i podstaći interesovanje.

SADRŽAJ	ZNANJE I RAZUMIJEVANJE	SPOSOBNOSTI	VRIJEDNOSTI, STAVOVI, PONAŠANJE	AKTIVNOST UČENIKA	AKTIVNOST NASTAVNIKA
PROGRAMIRANJE Složene strukture podataka. Funkcije i podprogrami.	Indeksirane promjenljive - nizovi. Rad sa podacima.	Sposobni su pokrenuti program BASIC Sposobni su primijeniti	Primijenjuju verziju BASIC-a koja im je na raspolaganju. Primijenjuju	Interesuju se kaje probleme je moguće riješiti koristeći BASIC. Uredno vode	Izlaže nastavno gradivo. Nastavnu građu izlaže na način kojim će kod

Grafika i zvuk.		naredbe koje pripadaju strukturi ponavljanja.	funkcije i podprograme.	pribilježbe. Izvode predviđene vježbe	učenika izazvati pažnju i podstaći interesovanje.
Izrada programa po zadatku i samostalno.	Postavljanje zadatka, izrada algoritma i pisanje programa. Dokumentiranje i testiranje programa.	Sposobni su riješavati probleme uz pomoć računara. Također znaju formulisati i postaviti zadatak za koji je moguće napisati program.	Imaju potpuno izgrađen pozitivan stav prema računarima (uključujući kompletan hardver i softver) te se u svom radu oslanjaju na usluge koje im računar pruža.	Aktivno učestvuju u izvođenju vježbi. Demonstriraju svoja rješenja. Pomažu drugim učenicima.	Primjere programiranja bira pažljivo, one koji imaju edukativnu vrijednost i koji su u korelaciji sa sadržajima iz drugih nastavnih predmeta.

1.1.Prijedlog moguće razrade po nastavnim jedinicama

UVOD U PROGRAMIRANJE

Nastavna jedinica	Obrazovna postignuća
Rješavanje problema uz pomoć računara Od ideje do programa	Naučiti postaviti I riješiti problem. Razvijati vještine osmišljavanja rješenja problema kroz upotrebu osnovnih algoritamskih struktura
PROGRAMIRANJE	
Osnovne karakteristike programskih jezika	Usvojiti osnovne činjenice o programskim jezicima, te razlike među njima. Uočiti potrebu za uvođenjem programa prevodioca I njihovom ulogom. Uvesti I usvojiti pojam programiranja.
Metodologija programiranja	
Programski jezik BASIC	Naučiti šta je BASIC, način rada, alphabet BASIC-a, tipove podataka, promjenljivih, izraza I koje su osnovne BASIC naredbe.
Osnovne naredbe BASIC-a	
Programske structure: Linijska I razgranata struktura	Razvijati vještinu razumjevanja I izrade pseudokoda algoritma I grafičkog prikaza algoritma prema zadatom pseudokodu
Programske structure: Ciklična struktura – FOR petlja	
Ciklična struktura: WHILE, DO ...LOOP petlje	
Složene structure podatka	Usvojiti pojam niza I upoznati njegove karakteristike I primjenu kroz zadatke
Potprogrami I funkcije	Upoznati pojam potprograma I funkcija u programiranju I njihovu primjenu
Grafika I zvuk	Upoznati se sa mogućnošću crtanja u Qbasic-u,

	te vrstom grafičkih ekrana. Naučiti naredbe za crtanje I zvuk
Rad sa podacima	Uoznati naredbe za rad sa podacima

2. OPĆENITO O BASIC-U

BASIC je programski jezik. Služi za pravljenje jednostavnih programa i simulacija. Kao i većina drugih programskih jezika, za programiranje koristi naredbe i varijable koje zajedno čine programski kod kojim se piše program. Godine 1990. evoluirao je u Visual Basic, a danas je već aktualna verzija Visual Basic Net. Napravljen od strane Microsoft kompanije. Riječ BASIC nastala je od početnih slova riječi **B**eginner's **A**ll-purpose **S**ymbolic **I**nstruction **C**ode (višenamjenski simbolički kod naredbi za početnike).

QBasic (skraćenica od Quick Basic) je programski jezik opće namjene za početnike. Ovaj programski jezik se može jednostavno naučiti i koristiti pri rješavanju različitih problema uz pomoć računara. Qbasic je jedna od verzija programskog jezika Basic-a.

Dakle Qbasic jeste:

- programski jezik iz familije Basic-a (kao prvi korak u učenju programiranja)
- prevoditelj jezika korisnika u jezik računara (cifara 0 i 1)
- jezik visokog nivoa (kao Pascal, Fortran, Logo)
- dio radnog sistema MS DOS koji upotrebu računara prilagođava našim željama

GLAVNA OBILJEŽJA BASIC-a

Prednosti:

- jednostavan
- prilagodljiv stepenu znanja
- pogodan za pisanje, ispravljanje i provjeru programa
- primjenjiv za rješavanje zadataka u različitim životnim područjima
- blizak sa naredbama radnog sistema
- rasprostranjen

Nedostaci:

- nije standardiziran postoji veliki broj dijalekata
- sporije izvođenje programa
- nešto neekonomičnije trošenje memorije

ELEMENTI

Kao i naš govorni jezik programski jezik Qbasic čine znakovi:

1. Velika slova A-Z i mala slova a-z (engleske abecede)