

Seminár z informatiky

Učebné osnovy predmetu

Názov predmetu:	Informatika
Časový rozsah výučby:	4. ročník – 2 hod. týždenne – spolu 60 vyuč. hodín
Kód a názov študijného odboru	79025 – gymnázium (od 2009/2010) 7902 J – gymnázium (od 2012/2013)
Stupeň vzdelania	ISCED 3A
Dĺžka štúdia	1 rok
Forma štúdia	denná
Vyučovací jazyk	Slovenský jazyk

Školský vzdelávací program predmetu pre 5. – 8. ročník osemročnej formy štúdia sa zhoduje so školským vzdelávacím programom 1. – 4. ročníka štvorročnej formy štúdia.

Učebné osnovy sú totožné so vzdelávacím štandardom ŠVP pre príslušný predmet.
V prvom ročníku je hodinová dotácia navýšená o jednu vyučovaciu hodinu.

VZDELÁVACIE OBLASTI PREDMETU

- **Reprezentácie a nástroje** – práca s grafikou, práca s textom, práca s prezentáciami, práca s multimédiami, práca s tabuľkami, informácie, štruktúry.
- **Komunikácia a spolupráca** – prezentovanie informácií prostredníctvom webovej stránky, vyhľadávanie na webe, práca s nástrojmi na spoluprácu a zdieľanie informácií, práca s nástrojmi na komunikáciu.
- **Algoritmické riešenie problémov** – analýza problému, jazyk na zápis riešenia, pomocou postupnosti príkazov, pomocou nástrojov na interakciu, pomocou premenných, pomocou cyklov, pomocou vetvenia, interpretácia zápisu riešenia, hľadanie a opravovanie chýb.
- **Softvér a hardvér** – práca so súbormi a priečinkami, práca v operačnom systéme, počítač a prídavné zariadenia, práca v počítačovej sieti a na internete, práca proti vírusom a špehovaniu.
- **Informačná spoločnosť** – bezpečnosť a riziká, digitálne technológie v spoločnosti, legálnosť používania.

V rámci predmetu sú aplikované nasledovné prierezové témy :

- **Mediálna výchova** – práca a spracovanie informácií, budovanie informatickej kultúry
- **Multikultúrna výchova** – medzinárodná komunikácia
- **Rozvoj komunikačných zručností** – učiť sa prezentovať svoju prácu
- **Rozvoj kooperácie a komunikácie v skupine**
- **Právo a etika**
- **Právne vedomie, osobnostný a sociálny rozvoj**

1. PROGRAMOVANIE

1.1. TYPY ÚDAJOV A PREMENNÉ

Obsah

Pojmy, vlastnosti, vzťahy:

- jednoduché typy – čísla, logické hodnoty, znaky, texty, konštanty – ich obmedzenia
- zložené typy – jednorozmerné a dvojrozmerné polia, záznam
- výrazy (operácie, premenné, zátvorky), pravidlá vyhodnocovania výrazov, priority operácií, funkcie, generátor náhodných čísel
- premenná (meno, hodnota, typ), príkaz priradenia (prepísanie pôvodného obsahu), vzťah premennej a pamäti
- prevody medzi typmi – číslo na text, text na číslo, prevod do/z číselnej sústavy

Požiadavky na vedomosti a zručnosti

Žiak vie:

- interpretovať výrazy a postupnosti priradovacích príkazov (číselné aj logické)
- identifikovať rozdiely pri práci s celými a reálnymi číslami
- pristupovať k prvkom jednorozmerných aj dvojrozmerných polí, k položkám záznamu
- manipulovať so znakovými reťazcami a ich podreťazcami

1.2. JAZYK ZÁPISU ALGORITMOV

Obsah

Pojmy, vlastnosti, vzťahy:

- algoritmus, programovací jazyk, vývojové prostredie, kompilácia
- program – spustenie, vykonanie, zastavenie
- pravidlá zápisu (syntax) – rezervované slová, identifikátory, konštanty, oddeľovače, programové konštrukcie
- deklarácie (deklaračná časť) a príkazy (príkazová časť)
- postupnosti príkazov (mechanizmus vyhodnocovania)
- opakovanie (typy cyklov, podmienky, počítadlo, telo cyklu, vnorené príkazy)
- vetvenie (typy podmienených príkazov, mechanizmus vyhodnocovania, vnorené príkazy)
- podprogramy, parametre (mechanizmus volania, lokálne premenné, parametre, návratová hodnota)
- opakujúce sa vzory

Požiadavky na vedomosti a zručnosti

Žiak vie:

- sformulovať formálny zápis algoritmu podľa kritérií rôznych vykonávateľov

- usporiadať príkazy do príkazových konštrukcií podľa pravidiel
- rozpoznať a odstrániť chyby vo formálnom zápise algoritmu
- vytvoriť riešenie pomocou pripravených podprogramov

1.3. TVORBA ALGORITMOV

Obsah

Pojmy, vlastnosti, vzťahy:

- problém, podproblém, zovšeobecňovanie, kritické situácie, obmedzenia
- textové a grafické výpisy, textové súbory (čítanie, zápis)
- prostriedky jazyka pre získanie vstupu, interakcia, interaktívne programy
- analýza riešenia problému – vstupy, výstupy, vzťahy medzi vstupom a výstupom
- riešenie problémov

Požiadavky na vedomosti a zručnosti

Žiak vie:

- rozpoznať zo zadania úlohy vstupné informácie, popísať očakávané výstupy, výsledky, výsledné akcie
- sformulovať, v čom spočíva informatický problém
- rozpoznať miesta, kde sa treba rozhodovať (vetvenie)
- identifikovať opakujúce sa vzory (opakovania)
- napláňovať riešenie úlohy ako postupnosť príkazov vetvenia a opakovania
- zovšeobecňovať (napríklad používaním premenných)
- rozdeliť problém na podproblémy (procedúry, funkcie)
- skladať riešenie problému z vyriešených podproblémov
- upraviť riešenie úlohy vzhľadom na rôzne požiadavky a obmedzenia (programu zmeniť vstupy, výstupy, spôsob spracovania, obmedzenie sady príkazov, premenných, typov)
- manipulovať s priebežne zapamätanými údajmi (premenná), manipulovať so štruktúrami
- zostaviť program, v ktorom sa vyhľadáva, modifikuje, vytvára, vypisuje, resp. vykresľuje

1.4. LADENIE, TESTOVANIE, ZDÔVODŇOVANIE

Obsah

Pojmy, vlastnosti, vzťahy:

- interpretovanie programu, krokovanie
- testovanie, správnosť a kvalita riešenia
- chyby v programe, logické chyby, chyby počas behu – hľadanie, identifikovanie, opravovanie

Požiadavky na vedomosti a zručnosti

Žiak vie:

- realizovať a interpretovať po krokoch aj formálny zápis postupu / algoritmu (krokovanie), vysvetliť proces vykonávania programu
- diskutovať o kritických situáciách a obmedzeniach, ktoré súvisia s riešením úlohy, uvádzať argumenty
- objaviť a vyjadriť princíp fungovania algoritmického riešenia (aj v rôznych iných typoch zápisov algoritmu, iný programovací jazyk, neformálny zápis)
- určiť vzťahy medzi vstupmi a výstupmi
- diskutovať o efektívnosti konkrétneho algoritmického riešenia vzhľadom na obmedzenia kontextu
- diskutovať o efektívnosti využívania IKT pri riešení konkrétneho problému
- overiť správnosť riešenia, algoritmu, zvoliť vhodnú testovaciu postupnosť údajov
- hľadať a opravovať chyby v algoritme, identifikovať miesta programu, v ktorých môže dôjsť k chybám počas behu programu a vedieť ich opraviť
- odhaľovať a riešiť kritické situácie
- odhadovať zložitosť konkrétnych algoritmov – počet vykonaných krokov, počet porovnaní, počet volaní, veľkosť očakávaného výstupu
- objaviť všeobecné vzťahy medzi informáciami
- popísať vzťahy medzi informáciami na abstraktnej úrovni
- generovať všeobecne platné pravidlá spracovania informácií
- porovnať techniky na analýzu väčších súborov údajov
- kriticky posúdiť klasické algoritmy a implementovať originálne algoritmy
- vyhodnotiť konkrétne algoritmy – ich efektivitu, správnosť a zrozumiteľnosť
- diskutovať o možnostiach riešenia úloh v rôznych programovacích jazykoch