

Obsah

9. ročník

- 20. Autorské práva ...40
- 21. Internet – ako funguje- opakovanie ...41
- 22. (16. Dvojková a desiatková sústava- opakovanie 8. roč.) ...43
- 23. Osmičková a šestnástková sústava –prevody ...43
- 24. WWW prehliadač ...46
- 25. Neinteraktívna a interaktívna komunikácia ...47
- 26. Programovanie, algoritmus ...48

9. ROČNÍK

20. Autorské práva

Každý program je podľa našich zákonov dielo. Autora programu chráni autorský zákon, ktorý určuje jeho práva a povinnosti. Podľa autorského zákona je právom autora rozhodovať o tom, ako sa bude jeho dielo používať alebo šíriť. Keď si od kamaráta kopírujeme napr. hudbu, video alebo počítačovú hru, aby sme ju mohli doma používať, porušujeme zákon. Rovnako ako nekradneme veci, nekradnime ani softvér.

Ak si kúpime počítačový program, jeho súčasťou je aj licenčná zmluva, ktorá stanovuje podmienky, za akých ho môžeme používať. Ak tieto podmienky dodržiavame, hovoríme, že máme legálny softvér. V prípade, že program chceme používať napr. v počítačovej učebni alebo vo viacerých kanceláriách firmy, musíme si zakúpiť väčší počet programov alebo multilicenciu. Počet počítačov, na ktorých sme legálne používať určitý program s multilicenciou, býva určený konkrétnym počtom počítačov alebo miestom, napr. v škole. Osobitná skupina sú školské a študentské licencie. Obyčajne sú oveľa lacnejšie, ale zakazujú používanie príslušného programu na komerčné účely (pomocou nich vytvoriť produkt, ktorý by sa predával).

Na Internete nájdeme množstvo shareware a freeware programov. Za používanie freeware programov sa neplatí, ak ich používame v súlade s licenčnou zmluvou. Shareware programy môžeme zvyčajne používať počas určenej doby, napr. 30 dní, bezplatne. Po tejto skúšobnej dobe sa musíme rozhodnúť, či program prestaneme používať, alebo zaň zaplatíme.

Používateľ, ktorý zodpovedne pristupuje k využívaniu počítačového programu, by mal vedieť, na čo je určený a aké informácie sa dajú pomocou neho získať a spracúvať. Špeciálna skupina (väčšinou mladých) používateľov sa snaží využiť nedostatky v práci programátorov a iných používateľov a svojou činnosťou často spôsobujú úmyselné škody. Typickým príkladom sú prieniky do počítačov a odcudzenie alebo zmena údajov, ktoré sú v počítačoch uložené. Programátori musia myslieť na to, že chyba v programe by nemala spôsobiť materiálne škody, zranenia, alebo dokonca smrť ľudí.

Aby sme sa vyhli zbytočným problémom, nemali vy sme ani vo svete počítačov zabúdať na slušné správanie. Základné pravidlo slušného správania je: „Nerob druhým to, čo nechceš, aby druhí robili tebe.“

Autorské práva

1. komerčný softvér je každý softvér ktorý jeho autor zaregistroval v súlade s uvedeným zákonom ako komerčný. Plne sa na neho vzťahujú autorské práva.

2. shareware je taký softvér ktorý, buď v jeho plnej verzii alebo iba v čiastočnej verzii, jeho autor umožňuje legálne používať každému používateľovi ktorý ho používa na nekomerčné účely. Niekedy je použitie sharewaru viazané iba na určitú počítačnú dobu po uplynutí ktorej si musí používateľ daný program buď zakúpiť alebo v prípade ak to

autor umožňuje sa u nich zaregistrovať a ďalej používať iba určitú neúplnú verziu tohto programu.

3. freeware je softvér ktorý jeho autor určil na voľné komerčné aj nekomerčné použitie všetkým užívateľom pričom sa vzdali nároku na finančné vyrovnanie. Autorské práva freewaru sú však ponechané jeho autorovi preto ho užívatelia nemôžu modifikovať ani vkladať do vlastných produktov

4. public domain je podobne ako freeware ponechaný verejnosti na voľné používanie ale oproti freewaru s tým rozdielom že sa jeho autori úplne vzdali autorských práv a dali každému užívateľovi právo.

21. Internet – ako funguje

Internet je celosvetový systém navzájom prepojených počítačových sietí. Internet je decentralizovaná sieť – nemá žiadny centrálny uzol, ani vlastníka, ani centrálného správcu.

Svet www pracuje na princípe klient – server.

- www server je program, bežiaci na počítači, na ktorom sú umiestnené www stránky.

- www klientom je prehliadač na našom lokálnom počítači. Klient sa obracia na server so žiadosťou o zaslanie stránky cez Internet. Každý počítač zapojený do počítačovej siete musí mať svoju IP adresu. Na nájdenie www stránky na Internete potrebuje prehliadač poznať jej adresu nazývanú URL – Uniform Resource Locator.

Pomocou URL sú všetky stránky na Internete jednoznačne určené.- príklad URL:

<http://www.spn.sk/knihkupectvo/knihy.html>.

Preložené do ľudskej reči – chceme, aby nám prehliadač zobrazil stránku knihy.html, ktorú nájde na počítači s adresou www.spn.sk v adresári knihkupectvo

<http://> označuje protokol umožňujúci prehliadaču komunikáciu s počítačom, na ktorom je hľadaná stránka.

www.spn.sk je doménová adresa počítača, na ktorom je stránka

[/knihkupectvo/](#) je názov adresára, v ktorom má počítač hľadať požadovanú stránku súboru so zdrojovým textom požadovanej stránky knihy.html je meno suboru stránky.

www-stránky :- **hypertext** – jednotlivé www-stránky sú vzájomne pospájané pomocou odkazov, vytvárajú tak hypertextový dokument

- jazyk **HTML** (hypertext markup language- „odkazovací značkovací jazyk“)- jazyk v ktorom sú písané www-stránky.

- štruktúra www-stránok, vedieť popísať **lineárnu, hierarchickú** štruktúru a štruktúru vo forme **pavučiny** – ich výhody a nevýhody + orientácia na stránkach, tzn. Poznať jednotlivé tlačidlá a ich funkcie v ovládacom paneli

FTP protokol :- prenos súborov medzi ľub. Počítačmi na Internete sa riadi protokolom FTP /File Transfer Protocol/ - na jednom počítači beží FTP klient (TotalCommander, WinCMD, WinSCP) a na druhom FTP server.

FTP klient sa spojí s FTP serverom a požiada ho o zaslanie príslušného súboru – download.

Upload – spätné zaslanie súboru na FTP server.

- anonymný FTP server – nemusíme mať na danom počítači, kde beží FTP server zriadené konto.

Fungovanie Internetu a bezpečnosť na Internete

Sieťové protokoly :- pakety (datagramy): informácie putujú po Internete vo forme menších blokov dát tzv. paketov, skladajú sa zo záhlavia kde je cieľová adresa a dátovej časti kde sú údaje. Aby sa pakety dostali na miesto určenia, musia zariadenia zapojené do siete dodržiavať pri komunikácii dohovorené pravidlá – sieťový protokol.

- protokol **TCP/IP**, TransmissionControlProtocol/InternetProtocol – je označením viacerých protokolov, ktoré sa používajú pri komunikácii prostredníctvom Internetu.

TCP/IP model sa skladá z týchto vrstiev – fyzická, linková, sieťová, transportná a aplikačná.

- **IP adresa** – jednoznačná a jedinečná adresa počítača, 4 čísla (každé v rozsahu od 0 po 255) oddelené bodkami. Na Internete neexistujú 2 PC s rovnakou IP adresou

- **Doménové adresy a DNS** :- doménové adresy – skladajú sa z niekoľkých reťazcov znakov vzájomne oddelených bodkou. - top level domain – doména, kt. je najvyššia v systéme domén.

Delenie – všeobecné tzv. **generické domény** (.edu, .com, .net, .gov, .arpa ...) a na dvojznakové domény podľa **štátov** (.sk, .cz, .uk, .de)

- DNS – **systém doménových adries**, ktorý umožňuje prekladanie doménových adries na IP adresy a naopak.

- menný (name) server – počítač v ktorom sa nachádza obrovská databáza, ktorá ku každej doménovej adrese prideluje príslušnú IP adresu.

- základná softvérová výbava pre pripojenie na Internet je IP adresa počítača, adresa aspoň jednej brány – **gateway** (tzn. smerovača alebo servera, ktorý spája počítač s Internetom) a adresa aspoň jedného DNS servera.

Ochrana poč.siete :- hacker- prihlasovanie sa do siete, tvorba „dobrého hesla“ – základné bezpečnostné pravidlá - počítačové vírusy (červy, trójske kone...)- hromadné zasielané e-poštové správy (hoax, spam, adware).

Ochrana listového tajomstva :- šifrovanie sa nazýva proces transformácie textovej správy do tajného kódu. Šifrovací program šifruje a dešifruje správy pomocou algoritmu alebo

matematického vzorca a pomocou šifrovacieho kľúča. - najrozšírenejšie a najbezpečnejšie sú v súčasnosti verejné šifrovacie systémy používajúce nesymetrické šifrovanie. Pri nich sa šifrovací kľúč skladá z 2 častí: verejného kľúča (pozná ho každý odosielateľ - je umiestnený na Internete - šifruje ním správu) a súkromného kľúča (pozná ho iba adresát). PGP - program, kt. funguje takýmto spôsobom.

- digitálny podpis

22. (16.Binárna a desiatková sústava- opakovanie 8. Roč.)

23. Osmičková a šestnástková sústava –prevody

1. Osmičková číselná sústava

Naučili sme sa prevádzať čísla z desiatkovej sústavy do dvojkovej a naopak. Teraz si vysvetlíme **prevod z desiatkovej sústavy do osmičkovej**.

Osmičková sústava obsahuje čísla 0, 1, 2, 3, 4, 5, 6, 7. Základom sústavy je číslo 8 a pozičnými hodnotami sú mocniny čísla 8.

$$8^0 = 1$$

$$8^1 = 8$$

$$8^2 = 64$$

$$8^3 = 512 \dots$$

1.1.1


1.1.2 Postup prevodu z desiatkovej sústavy do osmičkovej:

- pri prevode z vyššej sústavy (desiatkovej) do nižšej platí, že budeme používať delenie, takže číslo 15 v desiatkovej sústave premeníme do osmičkovej nasledujúcim postupom:

$$(15)_{10} \Rightarrow 15:8 = \textcircled{1} \Rightarrow (17)_8$$

zvyšok

$\textcircled{7}$



Úloha: Premente nasledujúce čísla z desiatkovej do osmičkovej sústavy:

9, 17, 44, 68 a 96.

Prevod z osmičkovej sústavy do desiatkovej:

1.1.3

$$(103)_8 \Rightarrow 3 * 8^0 + 0 * 8^1 + 1 * 8^2 = 3 + 0 + 64 = (67)_8$$

Úloha: Premeňte nasledujúce čísla z osmičkovej sústavy do desiatkovej:

17, 36, 61, 70 a 123.

Šestnástková číselná sústava patrí medzi jednu z najzložitejších, pretože okrem čísel obsahuje aj písmená. Musí v sebe zahŕňať 16 čísel, ale čísel je deväť – 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a číslo 10 je nahradené písmenom A, 11 = B, 12 = C, 13 = D, 14 = E a 15 = F.

Základom sústavy je číslo 16 a pozičnými hodnotami sú mocniny čísla 16.

$$16^0 = 1$$

$$16^1 = 16$$

$$16^2 = 256 \dots$$

1.1.4

1.1.5 Postup prevodu z desiatkovej sústavy do šestnástkovej:

- pri prevode budeme používať delenie, takže číslo 50 v desiatkovej sústave premeníme do osmičkovej nasledujúcim postupom:

$$(50)_{10} \Rightarrow 50:16 = \textcircled{3} \Rightarrow (32)_{16}$$

zvyšok

$\textcircled{2}$ 

Príklad s písmenom:

$$(31)_{10} \Rightarrow 31:16 = 1 \Rightarrow (115)_{16} \Rightarrow (1F)_{16}$$

zvyšok

$$15 = F$$

Úloha: Premeňte nasledujúce čísla z desiatkovej do šestnástkovej sústavy:

9, 17, 40, 68 a 96.

1.1.6

1.1.7

1.1.8 Prevod zo šestnástkovej sústavy do desiatkovej:

$$(60)_{16} \Rightarrow 0 \cdot 16^0 + 6 \cdot 16^1 = 0 + 96 = (96)_{10}$$

Príklad s písmenom:

$$\begin{aligned} (1A2)_{16} &\Rightarrow 2 \cdot 16^0 + A \cdot 16^1 + 1 \cdot 16^2 = \\ &= 2 \cdot 1 + 10 \cdot 16 + 1 \cdot 256 = \\ &= 2 + 160 + 256 = (418)_{10} \end{aligned}$$

Úloha: Premeňte nasledujúce čísla zo šestnástkovej sústavy do desiatkovej:

17, B, 61, CA a 4F .

Veľmi šikovným nástrojom na prevod čísel medzi dvojkovou (Bin – binárnou), osmičkovou (Oct – oktálovou), desiatkovou (Dec – dekadickou) a šestnástkovou (Hex – hexadecimálnou) sústavou je *vedecká kalkulačka z prostredia Microsoft Windows*. Zadáme v nej číslo, ktoré chceme previesť, myšou zvolíme cieľovú sústavu a prečítame výsledok.

24. WWW prehliadač

Webový prehliadač alebo **webový prezerač** alebo **web browser** (z [angl.](#))

je [aplikačný softvér](#) umožňujúci používateľovi zobrazenie a interakciu s [HTML](#) dokumentmi hostovanými na [webových serveroch](#) alebo na lokálnom [súborovom systéme](#). Medzi prehliadače dostupné pre [PC](#) patria napr.: [Internet Explorer](#), [Firefox](#), [Opera](#), [Google Chrome](#) a [Safari](#). Prehliadač je najčastejšie používaným druhom tzv. „*používateľského*

agenta“ (user agent), t. j. klienta určitého [internetového protokolu](#). Najrozsiahlejšia známa zbierka prepojených dokumentov je známa ako [World Wide Web](#).

Najznámejšie webové prehliadače:

Internet Explorer, skrátene **IE** či **MSIE**, je [proprietárny webový prehliadač](#) od spoločnosti [Microsoft](#), momentálne dostupný ako súčasť [Microsoft Windows](#). Internet Explorer je dnes najpoužívanejším webovým prehliadačom, hoci od roku [2004](#) stráca podiel na trhu. V apríli [2005](#) bol podiel IE okolo 85% .

Netscape Navigator, tiež známy ako **Netscape**, bol [webový prehliadač](#) extrémne rozšírený v 90. Rokoch 20. Storočia. Svojho času bol najrozšírenejším prehliadačom na trhu, no do roku [2002](#) prakticky zmizol, do istej miery aj vďaka zahrnutiu prehliadača [Internet Explorer](#) do [operačných systémov Windows](#) od [Microsoftu](#), a slabej inovácii koncom 90. Rokov.

Mozilla Firefox je grafický [multiplatformový webový prehliadač](#), ktorý vyvíja [Mozilla Foundation](#) a stovky dobrovoľníkov. *Firefox* je anglické meno pre pandu červenú (*Ailurus fulgens*). Prehliadač je šírený ako [slobodný softvér](#) (pod licenciami [MPL](#), [GPL](#) a [LGPL](#)). Medzi populárne vlastnosti Firefoxu patrí vstavané blokovanie vyskakovacích (pop-up) okien, prehliadanie v kartách (tabbed browsing) a systém rozširovania pomocou modulov s dopĺňujúcimi funkciami. Zámerom tvorcov bolo obsadenie 10 % trhu s prehliadačmi do konca roku 2005.

Safari je [webový prehliadač](#) vyvíjaný spoločnosťou [Apple Inc.](#) Pre [operačný systém Mac OS X](#) a [Windows](#). Po prvý raz bol uvedený ako hlavný prehliadač vo verzii Mac OS X 10.3 Panther a od verzie 10.4 Tiger je jediným prehliadačom dodávaným so systémom. Je tiež jediným prehliadačom v mobilnom telefóne [iPhone](#).

Google Chrome je [opensourcový webový prehliadač](#) vyvíjaný spoločnosťou [Google](#). Beta verzia bola uvoľnená [2. Septembra 2008](#).

Chromium je [webový prehliadač](#) typu [open source](#), projekt, z ktorého si [Google Chrome](#) prevzal svoj zdrojový kód. Napísaný bol v programovacích jazykoch [C++](#) a [jazyku symbolických inštrukcií](#). Predkompilované verzie je možné nainštalovať pre [Windows](#), [Linux](#) a [Mac](#).

Vyhľadávanie informácií na Internete :

- **katalógy** Internetu – predstavujú hierarchicky, podľa kategórií a podkategórií, usporiadané zoznamy hypertextových odkazov na rôzne www-stránky (zoznam.cz, atlas.sk, yahoo.com)

- **vyhľadavacie stroje** – používajú na vyhľadávanie informácií o www-stránkach špeciálny program, ktorý prechádza pavučinou stránok a info o nich si zaznamenáva do svojej databázy. (google, altavista)- kladenie požiadaviek (jednoduché, zložené).

Získavanie a vyhľadávanie informácií

Všetko o WorldWideWeb :Na prezeranie www stránok potrebujeme mať v našom počítači nainštalovaný prehliadač www stránok (www browser). Prehliadače slúžia na: komunikáciu so zdrojom www stránky – www serverom (starajú sa napríklad o zobrazenie www, presunutie www stránky z Internetu na používateľov počítač), prácu s www stránkou (uloženie, vytlačenie...). Najznámejšie prehliadače sú Netscape Navigator, Microsoft Internet Explorer, Opera, Mozilla.

25. NEINTERAKTÍVNA A INTERAKTÍVNA KOMUNIKÁCIA

Neinteraktívna komunikácia- pojmy: neinteraktívny, neinteraktívna komunikácia – e-mail tzn. elektronická pošta

- e-mail : - e-mailový klient a server- skladba e-správy (hlavička – To, Subject, Cc, adresa(@ - at) a telo správy)- adresár (distribučný zoznam, nickname)- čítanie a uchovávanie správ: priečinky (folder)- posielanie správ: preposlať (forward), odpovedať (reply), prílohy (attachments)- pravidlá pri posielaní e-správ.

Interaktívna komunikácia- pojmy: interaktívny – s možnosťou okamžitej reakcie

Textové telefóny :- talk – jednoduchý nástroj na interaktívnu textovú komunikáciu, - IRC (Internet relay chat) a ICQ (I seek you) – vysvetliť základný rozdiel medzi nimi (adresa IRC servera verzus ICQ-klient)- emotikony – sú to ikony s vyjadrením nálady píšuceho.

Videokonferencia :- videokonferencia je vzdialená interaktívna komunikácia medzi dvomi a viacerými účastníkmi, pričom dochádza medzi nimi k prenosu zvukovej a obrazovej informácie (účastníci sa vidia aj počujú).

- multimedialny počítač – počítač, ktorý vie pracovať s viacerými typmi dát (nielen textovými a grafickými, ale i zvukovými)

- videokonferenčná aplikácia (NetMeeting) – okrem hardwaru a rýchleho sieťového pripojenia je potrebné mať na počítači nainštalovaný software, ktorý umožní realizovať videokonferenciu.

25. PROGRAMOVANIE

Algoritmus je konečná postupnosť presne definovaných inštrukcií na splnenie určitej úlohy.

Algoritmy môžu byť zapísané (implementované) vo forme počítačových programov. Logická chyba v algoritme môže viesť k zlyhaniu výsledného programu.

Pojem algoritmu sa často ilustruje na príklade receptu, hoci algoritmy sú často oveľa zložitejšie. V algoritmoch sa často niekoľko krokov viacnásobne opakuje (iterácia), alebo ďalší postup závisí od aktuálneho stavu (vetvenie).

Na riešenie tej istej úlohy môže existovať niekoľko rôznych algoritmov s rôznymi postupnosťami inštrukcií. Rôzne algoritmy sa tiež môžu líšiť v množstve času a pamäte potrebných na splnenie úlohy.

VLASTNOSTI ALGORITMU

Konečnosť

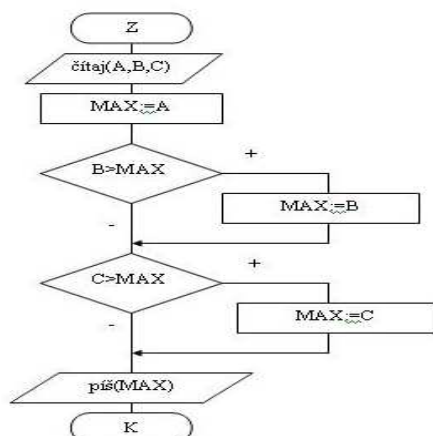
Každý algoritmus musí skončiť po vykonaní *konečného* počtu krokov. Tento počet krokov môže byť ľubovoľne veľký (podľa rozsahu a hodnôt vstupných údajov), ale pre každý jednotlivý vstup musí byť konečný. Postupy, ktoré túto podmienku nespĺňajú, sa môžu nazývať *výpočtové metódy*. Špeciálnym príkladom nekonečnej výpočtovej metódy je *reaktívny proces*, ktorý priebežne reaguje s okolitým prostredím.

Determinizmus

Každý krok algoritmu musí byť *jednoznačne a presne* definovaný; ako v každej situácii musí byť úplne zrejmé, čo a sa má vykonať, ako má vykonávanie algoritmu pokračovať. Pretože bežný jazyk zvyčajne neposkytuje úplnú presnosť a jednoznačnosť vyjadrovania, boli pre zápis v algoritmov navrhnuté [programovacie jazyky](#), ktorých má každý príkaz jasne definovaný význam. Vyjadrenie algoritmu v programovacom jazyku sa nazýva [program](#).

Vstup

Algoritmus zvyčajne pracuje s nejakými *vstupmi*, veličinami, ktoré sú mu odovzdané pred začatím jeho vykonávania, alebo v priebehu jeho činnosti. Vstupy majú definované [množiny](#) hodnôt, ktoré môžu nadobúdať.



Výstup

Algoritmus má aspoň jeden *výstup*, veličinu, ktorá je v požadovanom vzťahu k zadaným vstupom, a tým tvorí odpoveď na problém, ktorý algoritmus rieši.

Efektivita

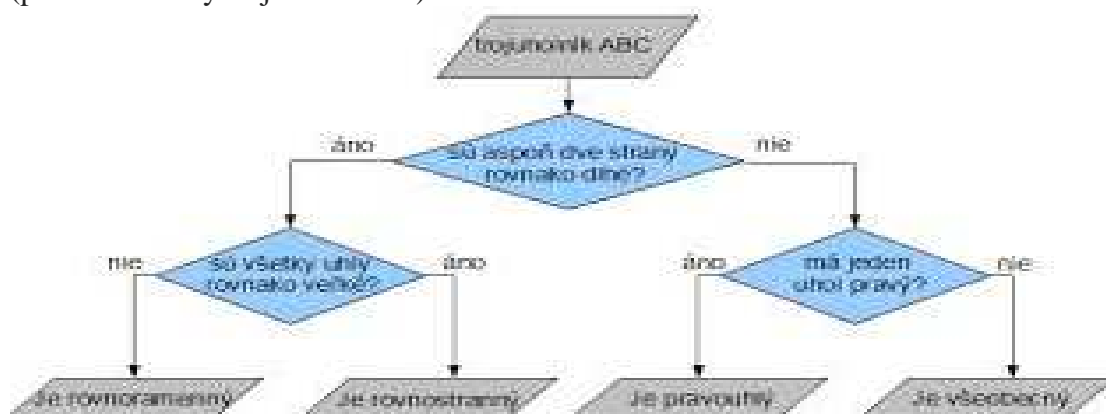
Všeobecne požadujeme, aby algoritmus bol *efektívny*, v tom zmysle, že požadujeme, aby každá operácia požadovaná algoritmom, bola dostatočne jednoduchá na to, aby mohla byť aspoň v princípe prevedená v konečnom čase iba s použitím ceruzky a papiera.

Všeobecnosť (hromadnosť)

Algoritmus nerieši jeden konkrétny problém (napr. „ako vypočítať 3×7 “), ale rieši všeobecnú triedu obdobných problémov (napr. „ako vypočítať súčin dvoch celých čísel“).

Vývojový diagram (pr. Porovnanie troch čísel- zistí, ktoré číslo má najvyššiu hodnotu)

(pr. – zistí o aký trojuholník ide)



Programovanie je zostavovanie a tvorba [programu](#).

V oblasti [výpočtovej techniky](#) ide o proces vytvárania algoritmov a [počítačových programov](#) vrátane ich testovania, ladenia a vypracovania dokumentácie na ich použitie. Programovanie je vytváranie a zapisovanie [návodov riešenia problémov](#) v [programovacích jazykoch](#) do tvaru programov na riadenie činnosti počítačov.

Pod pojmom *programovací jazyk* rozumieme prostriedok pre zápis [algoritmov](#) prevádzaných na počítači. Je komunikačným nástrojom medzi [programátorom](#), ktorý v programovacom jazyku formuluje postup riešenia daného problému, a [počítačom](#), ktorý program interpretuje technickými prostriedkami.

Rozdelenie programovacích jazykov

Programovanie je tvorivá činnosť ktorú vykonáva programátor teda človek , ktorý píše program (zdrojový kód) v niektorom z programovacích jazyku.

Existuje množstvo programovacích jazykov, z ktorých každý má svoju vlastnú syntax (*Pr. prog. jazykov: Jazyk C, C++, Pascal, HTML, PHP, Java,...*).

Program je algoritmus zapísaný v jazyku počítača tak, že ho daný procesor dokáže okamžite vykonávať.

Programovací jazyk – je to špeciálny jazyk v ktorom programátor dokáže pomerne ľahko vyjadrovať algoritmy, ktoré počítač bez problémov pochopí a vykoná.

Na začiatku existencie počítačov boli všetky programy písané v **strojovom jazyku (strojový kód)**. Pomocou tohto jazyka sa programy písali vždy pre konkrétny typ počítača (*programy neboli prenositeľné*). Programy boli ťažko čitateľné, bola to „reč čísel“.

Tento numerický zápis bol nepohodlný a tak sa časom vyvinul symbolický (alfanumerický) zápis – **assembler**.

Tieto programovacie jazyky sa označujú ako **jazyky nízkej úrovne**. Ich hlavnou výhodou je rýchlosť a nevýhodou strojová závislosť.

Ďalšiu triedu tvoria **vyššie programovacie jazyky**. Vznikli na základe potreby odstrániť strojovú závislosť a zvýšiť zrozumiteľnosť zdrojového kódu pre človeka. Programy napísané v niektorom vyššom programovacom jazyku je potrebné preložiť pomocou kompilátora – to sú **kompilované jazyky**. Kompilátor (prekladač) prekladá zdrojový kód jazyka (napísaný v programovacom jazyku) na strojový kód.

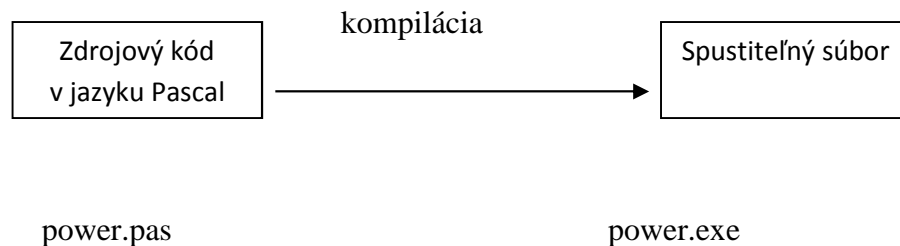
Niektoré zdrojové kódy, napísané v programovacom jazyku sa pred spustením programu neprekladajú (nekompilujú), ale sa riadok po riadku vykonávajú za pomoci **interpretera**. Teda interpreter číta a hneď vykonáva príkazy zapísané v zdrojovom kóde. Príkladom

takéhoto **interpretovaného jazyka** je jazyk HTML. Interpreterom príkazov jazyka HTML je internetový prehliadač (napr. Explorer, Netscape, ...).

Výhodou jazykov, ktoré sa kompilujú je fakt, že počas kompilácie zdrojového kódu sa vykonáva analýza (syntaktická, sémantická), t.j. kontrola zdrojového kódu a zobrazia sa všetky chyby. Aby bolo možné program skompilovať, musí byť napísaný bez chýb. Následne je možné daný program spustiť (skompilované programy sú obyčajné *.exe súbory).

Ak sa vyskytne chyba v zdrojovom kóde, ktorý sa nebude kompilovať, na chybu sa upozorní až pri jeho interpretovaní (program môže počas svojho behu uviaznúť).

Príklad kompilácie:



Neskôr sa jazyky začali deliť podľa typu a využitia. Poznáme **objektovo-orientované jazyky, funkcionálne, procedurálne jazyky a jazyky pre logické programovanie**.

Objektovo orientovaným programovaním sa bežne označuje prístup, keď programátor definuje nielen typ dátových štruktúr (atribúty), ale aj operácie ktoré sa dajú nad nimi vykonávať (metódy).

Procedurálne jazyky nemusia obsahovať objekty s vlastnosťami a udalosťami, sú programovacie jazyky, kde najvyššou zložkou býva procedúra alebo jednotka zahrňujúca jeden alebo viac procedúr.

Vo funkcionálnych jazykoch je množina funkcií komunikujúcich predávaním hodnôt. Často sa využívajú na zápis matematických algoritmov a na výučbu.

Jazyky pre logické programovanie využívajú matematický zápis.

Príklady programovacích jazykov a stručná charakteristika:

Jazyk C je univerzálny programovací jazyk, ktorého počiatky siahajú do konca sedemdesiatych rokov. Jazyk C pracuje priamo len so základnými dátovými typmi (číslo, znak). Vďaka štandardu ANSI C sú programy napísané v tomto jazyku prenositeľné na ľubovoľnú platformu s minimálnymi zmenami zdrojového kódu.

BASIC je pôvodne kompilovaný procedurálny jazyk, aj keď väčšina z používaných dialektov je interpretovaná. Basic je vhodný pre študentov a začínajúcich programátorov a preto sa využíva pri výučbe. S BASIC-om a jeho novšou verziou (**Visual BASIC**) sa stretnete najmä pod **OS Windows**. Visual BASIC podporuje už samozrejme prácu s objektmi a má niektoré ďalšie črty moderných programovacích jazykov.

Java je objektovo orientovaný programovací jazyk vytvorený firmou **Sun Microsystems**. Hlavnou výhodou Javy je jej platformová nezávislosť a bezpečnosť. Pozor, na rozdiel od JavaScriptu je Java kompilovaný jazyk.

Prolog je interpretovaný logický programovací jazyk. Základom programovania v Prologu je substitúcia premenných, rekurzia a spätný chod. Prolog má svoje využitie najmä v systémoch umelej inteligencie.

JavaScript je objektovo orientovaný programovací jazyk, ktorý patrí medzi najčastejšie využívané skriptovacie jazyky na strane klienta. Využíva sa najmä na kontrolu údajov z dotazníkov, či na zvýšenie interaktivity pavučinových stránok. (Je vykonávaný pomocou interpretera – internetový prehliadač)

PHP (Personal Home Page) PHP bolo inšpirované jazykmi podporujúcimi štruktúrované programovanie; najviac vlastností prebralo od jazyka C a jazyka Perl. V neskorších verziách bolo rozšírené o možnosť používať objekty. V súčasnosti je to výkonný objektovo orientovaný skriptovací jazyk používaný na strane servera. Programy jazyka PHP sa vpisujú priamo do HTML kódu. PHP sa využíva na generovanie dynamických HTML stránok. Jeho nespornou výhodou je jednoduchá syntax podobná programovaciemu jazyku C, ako aj podpora rôznych databázových serverov. (je interpretovaný, vykonáva sa na serveri nie v internetovom prehliadači). PHP dokáže spolupracovať s relačnými databázami, ako napríklad MySQL, Oracle, IBM DB2, Microsoft SQL Server.

HTML - **Hypertextový značkový jazyk (HyperText Markup Language)** – jednoduchý programovací jazyk určený pre tvorbu internetových stránok. Príkazy jazyka HTML sú interpretované internetovým prehliadačom. Špecifikáciu jazyka HTML udržiava [World Wide Web Consortium](#) (W3C). Aktuálna verzia je HTML 4.01, ale W3C plánuje nahradiť HTML jazykom XHTML, implementáciou jazyka XML, ktorá zjednoduší prácu prehliadačov aj tvorcov web stránok. Jazyk HTML je textový, umožňuje čítanie a upravovanie priamo v textovom editore.

XHTML - **Rozšíriteľný hypertextový značkový jazyk (Extensible Hypertext Markup Language)** - je značkový jazyk podobný HTML, ale s prísnejšou syntaxou.

Stále je objavujú nové možnosti prístupu k Internetu. Jazyk XHTML bol navrhnutý s ohľadom na kompatibilitu všetkých možných editorov a prehliadačov.

Jazyk XHTML je oproti jazyku HTML jednoduchší. Sú odstránené niektoré atribúty prvkov, ktoré upravovali vzhľad rozmery a pozíciu týchto prvkov. Tieto vlastnosti boli nahradené pravidlami jazyka CSS. **Kaskádové štýly** alebo **CSS** - skratka z angl. *Cascading Style Sheets*) je všeobecné rozšírenie HTML o možnosti opisu vzhľadu textu základnými parametrami bežného DTP (desktop publishing). Štýly umožnili oddeliť štruktúru HTML alebo XHTML od vzhľadu.

XML znamená **eXtensible Markup Language**, v preklade rozšíriteľný značkovací jazyk. Sila XML je najmä v jeho hierarchickej štruktúre a pomerne jednoduchom spôsobe kódovania. Umožňuje popisovať - označovať ľubovoľné dáta a prenášať ich medzi rôznymi aplikáciami a platformami. Hlavnou ideou XML je oddelenie obsahu a dizajnu dát.

C++ - objektovo orientovaný programovací jazyk, ktorý vznikol na základoch jazyka C – konkrétne šlo o *C with Classes* (C s triedami).

V porovnaní z jazykom C prináša nové vlastnosti, vrátane deklarácií, pretypovanie so syntaxou funkcií, `new/delete`, `bool`, referencie, `inline` funkcie, defaultné argumenty, preťaženie funkcií, priestory mien, triedy (zo všetkými ich črtami ako dedičnosť, členské funkcie, virtuálne funkcie, abstraktné triedy a konštruktory), preťaženie operátorov, šablóna, operátor `::`, ošetrovanie výnimiek, RTTI.

Treba poznamenať, každý programovací jazyk sa časom vyvíja, vznikajú/pridávajú sa nové príkazy, funkcie, teda vznikajú novšie a novšie verzie.

Niekedy sa môžete stretnúť aj s pojmmami:

Preprocesor – je súčasťou prekladača, upravuje zdrojový súbor tak, aby mal prekladač ľahšiu prácu, napr. vynecháva komentáre (sú určené len pre užívateľov, aby sa ľahšie orientovali v spleti príkazov programu), zaisťuje správne vloženie hlavičkových súborov, a pod.

Linker – alebo aj zostavovací program, pridelí relatívnemu kódu absolútne adresy. Výsledkom je spustiteľný exe súbor.

Debugger – ladiaci program,, používa sa pre ladenie – odhaľovanie chýb počas behu programu.