



evropský  
sociální  
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,  
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání  
pro konkurenceschopnost



**Masarykova střední škola zemědělská  
a Vyšší odborná škola, Opava,  
příspěvková organizace**

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

<b>Číslo projektu</b>	<b>CZ.1.07/1.5.00/34.0565</b>
Číslo materiálu	VY_32_INOVACE_284_Programovací_jazyky
Autor	Mgr. Pavel Vojkůvka
Průřezové téma	Informační a komunikační technologie
Předmět	Informatika
Ročník	3. / 4.
Datum tvorby	5. 10. 2012
Datum ověření	14. 11. 2012
Druh učebního materiálu	Prezentace
Anotace	Prezentace seznamuje studenty s historií programovacích jazyků, principy jejich fungování.
Klíčová slova	Program, Programovací jazyk, Syntax, Pascal
Metodický pokyn	Určeno k výkladu do hodiny a k praktickému procvičení
Pokud není uvedeno jinak, použitý materiál je z vlastních zdrojů autora	

# Programovací jazyky

Mgr. Pavel Vojkůvka



# Počítačový program

- ▶ Určen k zápisu instrukcí z vytvořeného algoritmu, kterému počítač rozumí a umí z něho vytvořit strojový kód

## Výhody:

- forma zápisu, které rozumí člověk i počítač
- nedá se ničím nahradit ani obejít

## Nevýhody:

- zápisu rozumí pouze programátor, ovládající jazyk
- Nízká forma názornosti ; může být nepřehledný

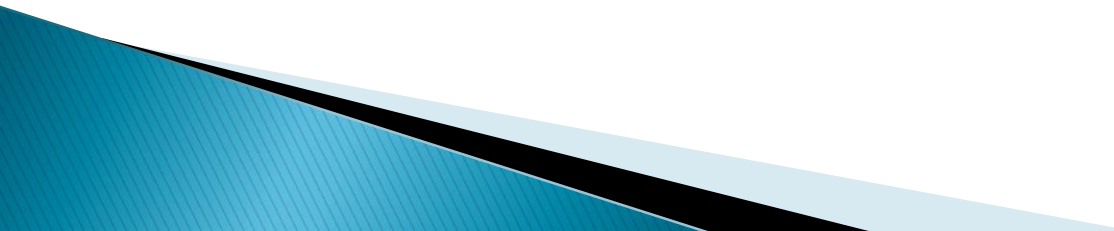
# Programovací jazyk

- ▶ Popis **výpočtů**, obvykle ve tvaru, jenž umožňuje provedení elektronickým počítačem (*program*)
- ▶ Standardizovaný nástroj pro komunikaci s počítačem
  - S jakými daty má počítač pracovat
  - Jak se tato data budou ukládat a přenášet
  - Které akce a kdy se mají provést

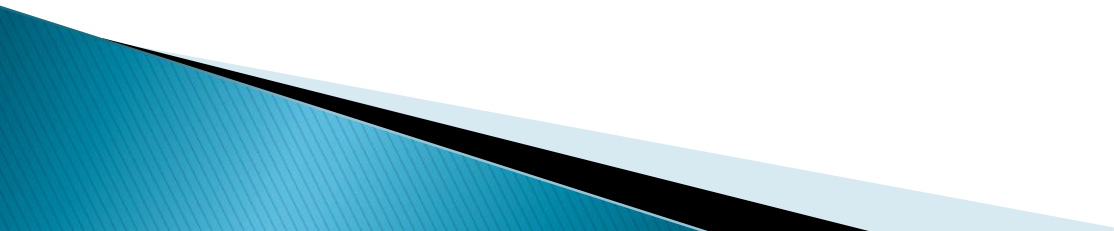
# Chyby v programech

- ▶ Syntaktické – špatně zapsaný nějaký příkaz. Překladač programu se zastaví a zobrazí hlášení o chybě.
- ▶ Logické – program je funkční, ale nedělá co má. Může skončit v nekonečném cyklu.
  - Tzv. zamrznutí

# Rozdělení programovacích jazyků

- ▶ Obecné – (C, C++, C#, Objekt Pascal, Visual Basic, .NET Framework, Java)
  - ▶ Databázové (MS Fox Pro, Oracle)
  - ▶ Pro webové klienty (Java, JavaScript, ActiveX)
  - ▶ Specializované (makrojazyky, simulační jazyky)
- 

# Nástroje pro tvorbu programů

- ▶ Editor
  - ▶ Překladač / interpreter x zpětný překladač
  - ▶ Spojovací program (linker)
  - ▶ Správa verzí
  - ▶ Ladicí program (debugger)
  - ▶ Nástroje pro ladění výkonu (profiler)
  - ▶ Testovací nástroje, generátory testů
  - ▶ Generátor instalačních balíčků
  - ▶ Nástroje pro internacionalizaci
- 

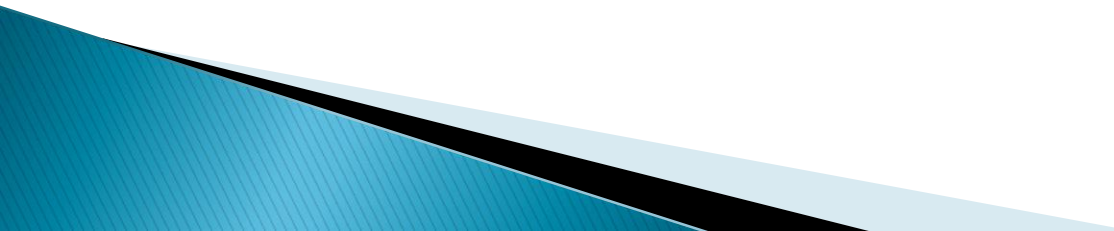
# Překladač a interpret

- ▶ Překladač neboli kompilátor je program, který ze zdrojového kódu vytvoří spustitelný soubor. Vytvoří instrukce pro procesor.
- ▶ Interpret načítá program v textovém zápisu a okamžitě jednotlivé příkazy vykonává. Program tedy není předem přeložený. Používá se například u skriptů ve webových stránkách

# Vývojové prostředí

- ▶ Je to komplexní nástroj na tvorbu programů.

## Obsahuje:

- překladač pro určený programovací jazyk
  - nástroje na vizuální tvorbu programu
  - hotové komponenty programů
  - nástroje pro ladění a odstraňování chyb
- 

# Důležité pojmy

- ▶ Proměnná – je to místo v paměti počítače, ve kterém se nachází určitá hodnota
- ▶ Identifikátor – je název proměnné
- ▶ Datový typ – je to určitý typ proměnné
  - char – znak
  - integer – celé číslo
  - real – reálné číslo
  - boolean – logická proměnná

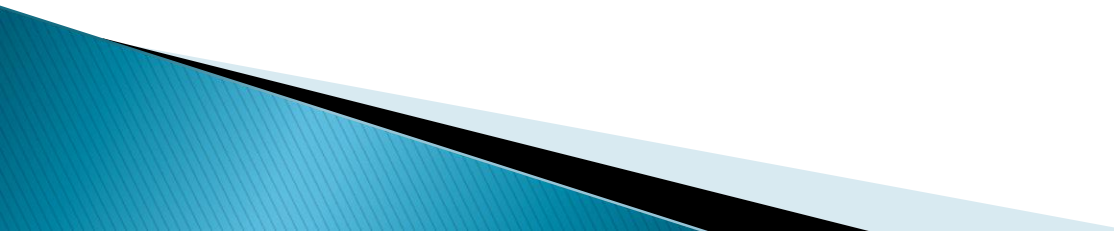
# Důležité pojmy

- ▶ Deklarace – stanovení typu proměnných. Provádí se vždy na začátku programu.
- ▶ Operátory a standardní funkce
  - Matematické
  - Logické
- ▶ Syntaxe –určení zápisu jednotlivých příkazů a částí programu

# Strukturované programování

- ▶ Soubor doporučení, jak by měl dobře napsaný program vypadat.
  - Tato doporučení vytvářejí jakési „mantinely“ pro programátory.
  
- ▶ Strukturované programování používá:
  - Proměnné
  - Procedury
  - Funkce

# Objektové programování

- ▶ Soubor doporučení, jak by měl dobře napsaný program vypadat.
  - ▶ Objektové programování používá:
    - Proměnné
    - Procedury
    - Funkce
    - Objekty
- 

# Vizuální programování

- ▶ Vizuální programování je dnes většinou objektové a současně strukturované.
- ▶ Spočívá v tom, že velké množství standardně používaných objektů (tlačítka, políčka, práci se soubory) je již připraveno a jen je myší přetahujeme do vytvářeného programu.

# Typy programovacích jazyků

## ▶ Imperativní jazyky

- Posloupnost příkazů měnících stav výpočtu
- Možnost opakovaného přiřazení do proměnné

## ▶ Deklarativní jazyky

- Posloupnost podmínek
- Počítač musí najít postup, jak je dodržet
  - Funkcionální jazyky – ML, Haskell, LISP
  - Logické jazyky – PROLOG, CLP, Gödel, Mercury
  - Dotazovací jazyky – SQL

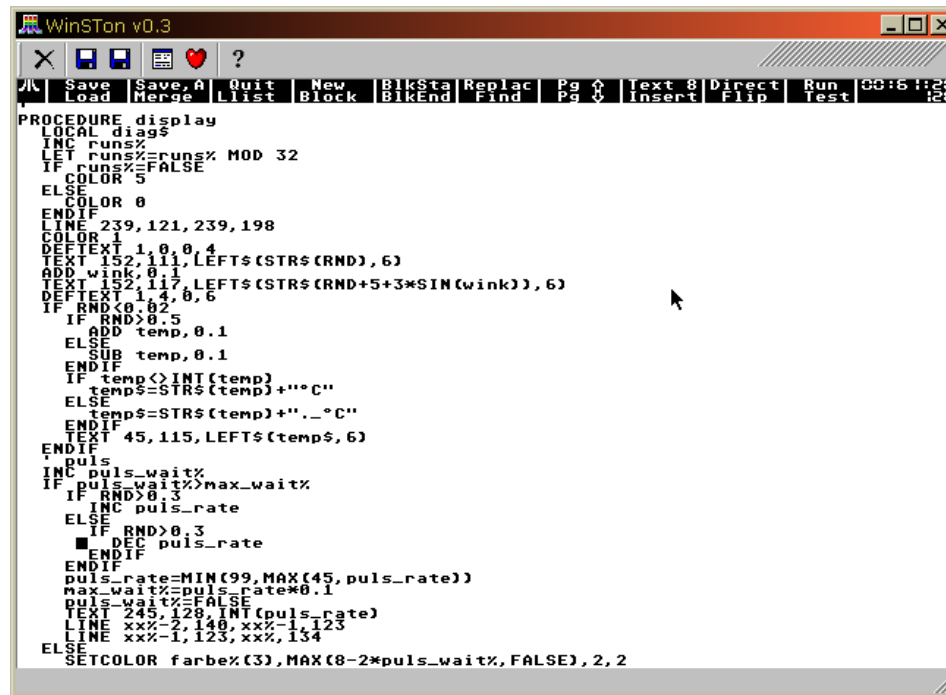
# Vývoj programovacích jazyků

- ▶ **Strojové jazyky**
  - Posloupnost instrukcí v číselném tvaru
- ▶ **Jazyky symbolických instrukcí / adres**
  - Pojmenované instrukce
  - Pojmenované adresy – proměnné, návěští skoků
  - Makrodefinice
- ▶ **Autokódy**
  - Složitější příkazy
  - Příklad: MAT (Minsk autokód – CTI, PIS, JDI NA, ...)
- ▶ **Vyšší programovací jazyky**

# BASIC (1964)

## Beginners All-Purpose Symbolic Instruction Code

- ▶ Vývojáři: John G. Kemeny, Thomas E. Kurz, Dartmouth University

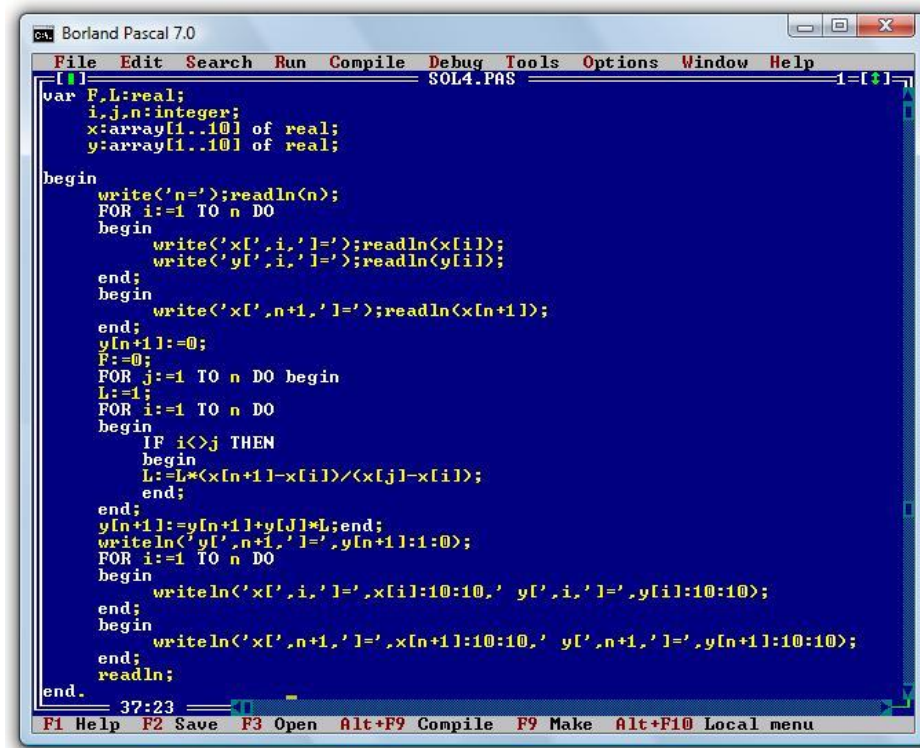


```
WinSTon v0.3
Save Load Save, A Merge Llist Quit New BlkSta Replac Pg Pg Text 8 Direct Run 00:16:20
Load Merge Llist Find Pg Insert Flip Test 128
PROCEDURE display
LOCAL diag$
INC runs%
LET runs%=runs% MOD 32
IF runs%=FALSE
COLOR 5
ELSE
COLOR 0
ENDIF
LINE 239,121,239,198
COLOR 1
DEFTEXT 1,0,0,4
TEXT 152,111,LEFT$(STR$(RND),6)
ADD wink,0.1
DEFTEXT 1,4,0,6
IF RND<0.02
IF RND>0.5
ADD temp,0.1
ELSE
SUB temp,0.1
ENDIF
IF temp<>INT(temp)
temp$=STR$(temp)+"°C"
ELSE
temp$=STR$(temp)+"°C"
ENDIF
TEXT 45,115,LEFT$(temp$,6)
ENDIF
puls
INC puls_wait%
IF puls_wait%>max_wait%
IF RND>0.3
INC puls_rate
ELSE
IF RND>0.3
DEC puls_rate
ENDIF
puls_rate=MIN(99,MAX(45,puls_rate))
max_wait%=puls_rate*0.1
puls_wait%=FALSE
TEXT 245,128,INT(puls_rate)
LINE xx%-2,140,xx%-1,123
LINE xx%-1,123,xx%,134
ELSE
SETCOLOR farbe%(3),MAX(8-2*puls_wait%,FALSE),2,2
```

Zdroj: [http://commons.wikimedia.org/wiki/File:GFA\\_BASIC\\_editor.png](http://commons.wikimedia.org/wiki/File:GFA_BASIC_editor.png)

# Pascal (1968–71)

- ▶ Niklaus Wirth, ETH Zurich
- ▶ jednoduchý → určen pro výuku programování
- ▶ Borland Turbo Pascal (Anders Hejlsberg)



```
var F,L:real;
    i,j,n:integer;
    x:array[1..10] of real;
    y:array[1..10] of real;

begin
  write('n=');readln(n);
  FOR i:=1 TO n DO
  begin
    write('x[',i,']=');readln(x[i]);
    write('y[',i,']=');readln(y[i]);
  end;
  begin
    write('x[',n+1,']=');readln(x[n+1]);
  end;
  y[n+1]:=0;
  F:=0;
  FOR j:=1 TO n DO begin
    L:=1;
    FOR i:=1 TO n DO
    begin
      IF i<>j THEN
      begin
        L:=L*(x[n+1]-x[i])/<math>x[j]-x[i]</math>;
      end;
    end;
    y[n+1]:=y[n+1]+y[j]*L;end;
  writeln('y[',n+1,']=',y[n+1]:1:0);
  FOR i:=1 TO n DO
  begin
    writeln('x[',i,']=',x[i]:10:10,' y[',i,']=',y[i]:10:10);
  end;
  begin
    writeln('x[',n+1,']=',x[n+1]:10:10,' y[',n+1,']=',y[n+1]:10:10);
  end;
  readln;
end.
```

Zdroj: <http://commons.wikimedia.org/wiki/File:LagrangePAS.jpg>

# Skriptovací jazyky

- ▶ Slouží obvykle k rozšíření nějaké aplikace
  - Operační systém – sh, command.com
  - WWW server – Perl, PHP
  - WWW prohlížeč – JavaScript, VBScript

# Další jazyky

- ▶ C / C++ / C#
- ▶ Java
- ▶ Python
  
- ▶ [http://cs.wikipedia.org/wiki/Kategorie:Programovací\\_jazyky](http://cs.wikipedia.org/wiki/Kategorie:Programovací_jazyky)

# Zdroje

- ▶ *CLEVINGER. Stránky všeobecně o programování.* [online]. [cit. 2012-10-05]. Dostupný z WWW: <http://k-prog.wz.cz/index.php>
- ▶ *Wikipedie: Otevřená encyklopedie: Programovací jazyk* [online]. [cit. 2012-10-05]. Dostupný z WWW: [http://cs.wikipedia.org/wiki/Programovací\\_jazyk](http://cs.wikipedia.org/wiki/Programovací_jazyk)