ITDevCon
European Delphi Conference

Introduction to CODEsign
Do IT twice!

Max Kleiner

maXbox
max@kleiner.com
softwareschule.ch

Check of Simplification

```
function StripTags2(const S: string):
string;
var
  Len: Integer;
  i, APos: Integer;
begin
  Len:= Length(S);
  i:= 0;
  Result:= '';
  while (i <= Len) do begin
    Inc(i);
    APos:= ReadUntil(i, len, '<', s);
    Result:= Result + Copy(S, i, APos-i);
    i:= ReadUntil(APos+1,len, '>',s);
  end;
end;
```

• 1

- Design your functions twice to simplify (expert systems, planning engines or classes) with the same result

• 2

```
Writeln(ReplaceRegExpr ('<[^>]*>',
                 '<p>This is text.<br/> This is line 2</p>','', True))
```

Check Performance

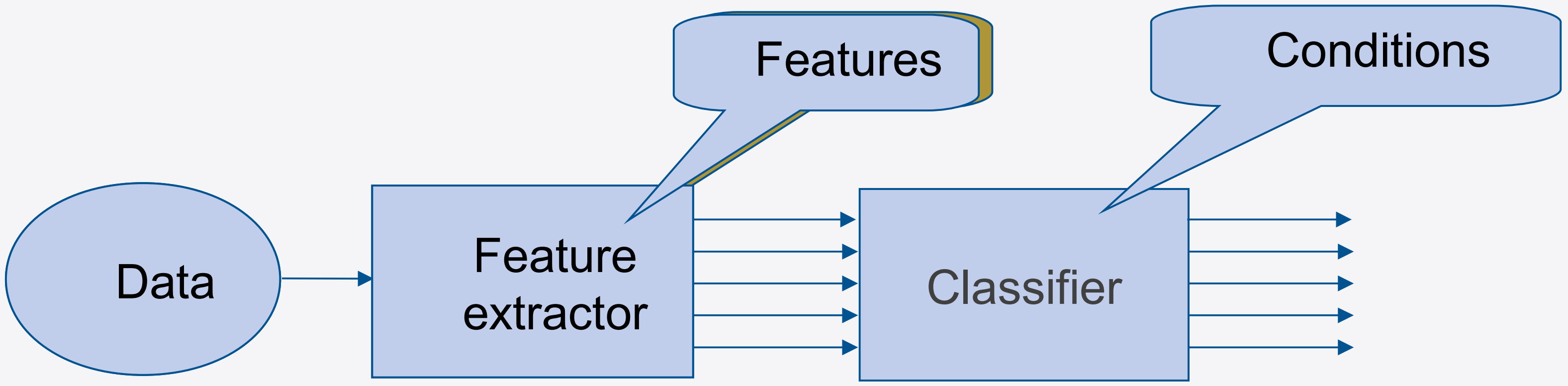
- 1

```
function IsPrime(N: Integer): Boolean;
var I: Integer;
begin
 for I:= 2 to N - 1 do
  if (N mod I) = 0 then
   exit;
 result:= true;
end;
```

- 2

```
function IsPrime2(acti: integer):
boolean;
var   j: Integer;
        isprim: boolean;
begin
 isprim:= true;
 if acti=1 then isprim:= false;
 for j:= 2 to round(sqrt(acti)) do
  if ((acti mod j) = 0) then begin
   isprim:= false;
   break
  end;
 result:= isprim;
end;
```

- function1 is slower than function2
- A profiler detects conditions based on features



2 Types of Validations

```
function GetLinesCount(sFileName : String):
Integer;
var
oSL : TStringlist;
begin
  oSL:= TStringlist.Create;
  oSL.LoadFromFile(sFileName);
  result:= oSL.Count;
  oSL.Free;
end; //[/mX4]
```

• 1

```
function getLinesCount2(sfilename:string):
double;
var    hFile : TextFile;
       sLine : String;
       iLinescount: Double;
begin
  result:=0;
  if not FileExists(sfilename) then exit;
  AssignFile(hFile, sFileName);
  Reset(hFile);
  closefile(hfile);
  iLinescount:=0;
  while NOT EOF(hFile) do begin
    ReadLn(hFile, sLine);
    iLinescount:=iLinescount+1;
  end;
 result:=iLinescount;
end;
```

• 2

- Check a function twice to compare the result (plausible)
- By the way function1 is faster than function2

Proof it twice

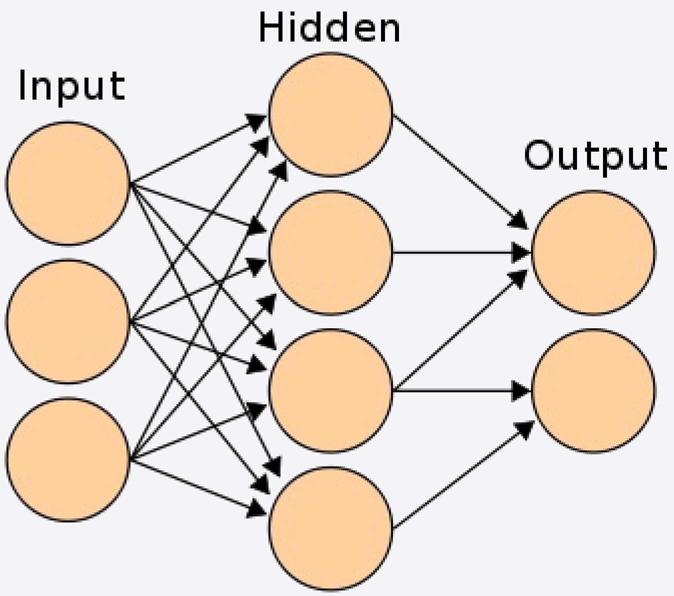
```
writeln('CRC:'+itoa(ComputeFileCRC32(exepath+'\maXbox3.exe')));
//writeln(itoa(CRC32(exepath+'\maxbox3.exe')));
   writeln('CRC:'+itoa(Crc32OfFile(exepath+'\maXbox3.exe')));
   writeln(intToHex(-1808407689,2));
```

- **1**

```
   writeln(IntToHex(CRC32OfFile(exepath+'\maXbox3.exe'),4));
```

- **2**

```
   writeln(intToHex(ComputeFileCRC32(exepath+'\maXbox3.exe'),2));
```



- Compare the reference

Run & Test with Redundancy

ITDevCon European Delphi Conference

```
function DownloadFile(SourceFile, DestFile: string):
Boolean;
begin
  try
    Result:= UrlDownloadToFile(Nil, Pchar(SourceFile),
                            PChar(DestFile),0,Nil) = 0;
  except
    Result:= False;
  end;
end;
```

• 1

```
wGet2('http://max.kleiner.com/images/texturemap.jpg','texturemap7.jpg');
DownloadFile('http://max.kleiner.com/images/texturemap.jpg','texturemap77.jpg')
```

• 2

```
//Test also Result
Result:= UrlDownloadToFile(Nil, Pchar(SourceFile),  PChar(DestFile),0,Nil) = 0;
```

- Supervised test Exception: “Socket Error # 11004”
- Non supervised redundancy of code at run-time
- Two functions as fallback for mission critical availability

Test twice

- Simple probabilistic testing theorem – Function Test, Unit Test
- Assumes that the functions are independent from each other
- Very efficient when trained in supervised environment
- Requires relatively small amount of test data and conditions to produce good results with high quality
- Check your function & test in unit integration

- **1**
- **2**

```
printF('addition theorem %.18f ',[maXcalc('sin(2.5/2)')])
printF('addition theorem %.18f ',[maXcalc('sqrt(1/2*(1-cos(2.5)))')])
printF('addition theorem2 %22.18f ',[maXcalc('cos(2.5/2)')])
printF('addition theorem2 %22.18f ',[maXcalc('sqrt(1/2*(1+cos(2.5)))')])
```

Windows crashed again. I am the Blue Screen of Death. No one hears your screams.

Document & save twice

- Make a Model or UML Diagram
- Write a Description or Comment with an example
- Save/Sync project in GIT & Bitbucket!

```
function StripTags2(const S: string): string;
```

• 1

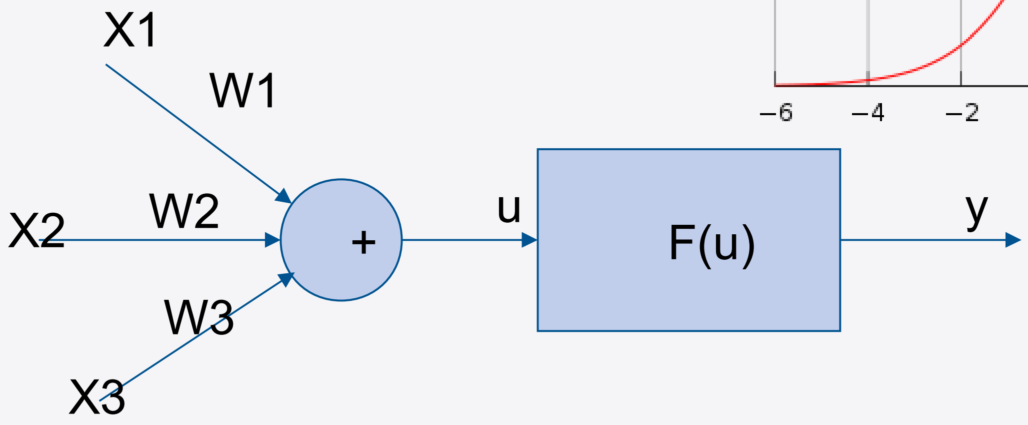
```
function StripTags2(const S: string): string;
// Strips tags from a HTML file: <p>This is text.<br/> ---> This is text.
// Make sure valid TStrings has been passed in
```

• 2

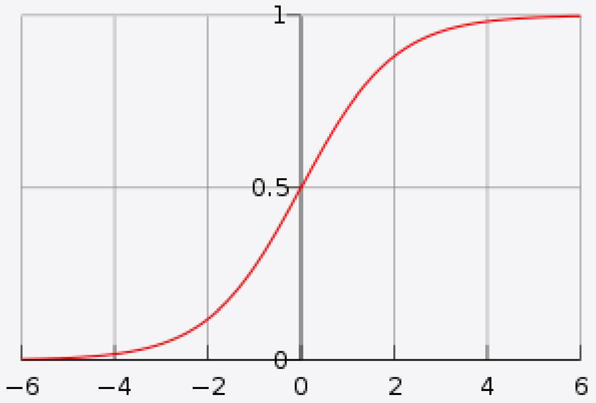
- They don't need to be complex; they just need to be clever in that they will allow for an easy/inelegant solution, and a difficult/elegant solution. In time, the students will learn to prefer the difficult/elegant solution, and that is how their brains will "domesticate" themselves into thinking algorithmically.

Visualize twice

- 1



- 2



Teach it twice

- **1**

```
TSortThread = class(TThread)
  strict private
    FBox: TPaintBox;
    //FSortArray: PThreadSortArray;
    FSortArray: TSortArray;
    FSize: Integer;
    FA, FB, FI, FJ: Integer;
    Fbolthslowmotion: boolean;
    procedure DoVisualSwap;
    procedure SetbolthSlowmotion(const Value:
                                  boolean);
```

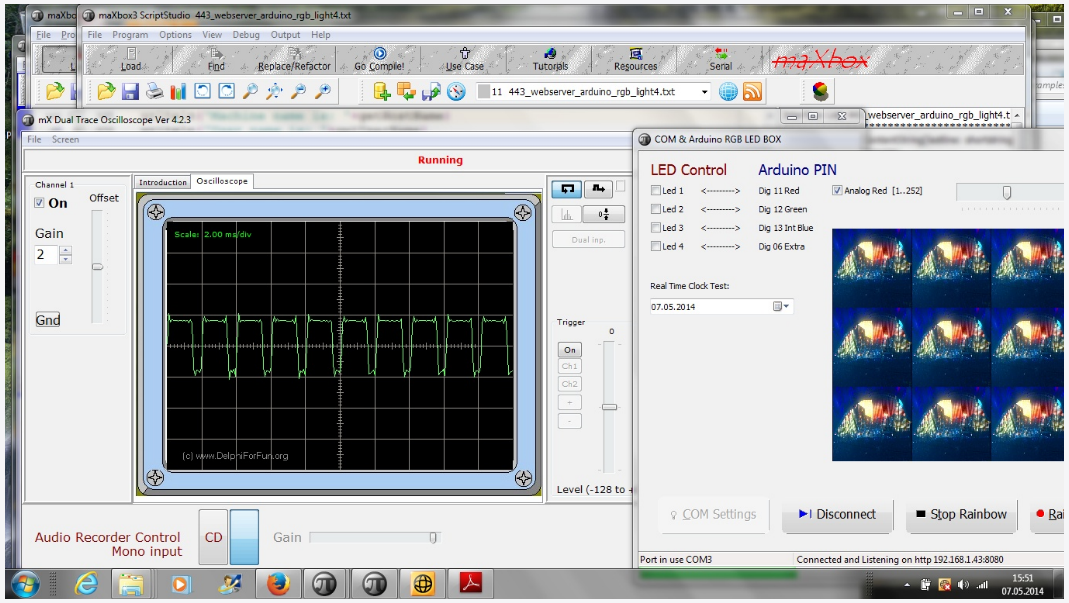
- **2**

```
  TThread = class
  private
{$IFDEF MSWINDOWS}
    FHandle: THandle;
    FThreadID: THandle;
{$ENDIF}
{$IFDEF LINUX}
    // ** FThreadID is not THandle in Linux
**
    FThreadID: Cardinal;
    FCreateSuspendedSem: TSemaphore;
```

- Train with data and functions
- Use validation data to optimize the teaching
- Test the teacher – Train the trainee!

CODEsign for

- SPAM filtering
- Computer vision
- OCR and Pattern Recognition
- Speech recognition
- Diagnostic utilities
- Industrial control
- Investment & Science
- Code formatting
- Gesture recognition
- Robotics, IOT
- Games
- Function approximation



About the Speaker

ITDevCon European Delphi Conference

Max Kleiner – softwareschule.ch
- max@kleiner.com

maXbox Software - www.softwareschule.ch

Products:
- maXbox – Scripter Studio
- DWS – Delphi Web Start, Delphi Web Security
- CryptoBox – Crypto processing library
- PEP – Pascal Education Program
-
- https://github.com/maxkleiner/maXbox3/releases
-