Software Development Process

N4/5 Computing Science

The Software Development Process

The SDP consists of seven main stages that are followed to develop software.

They are:

1. Analysis
2. Design
3. Implementation
4. Testing
5. Documentation
6. Evaluation
7. Maintenance

The iterative nature of the SDP

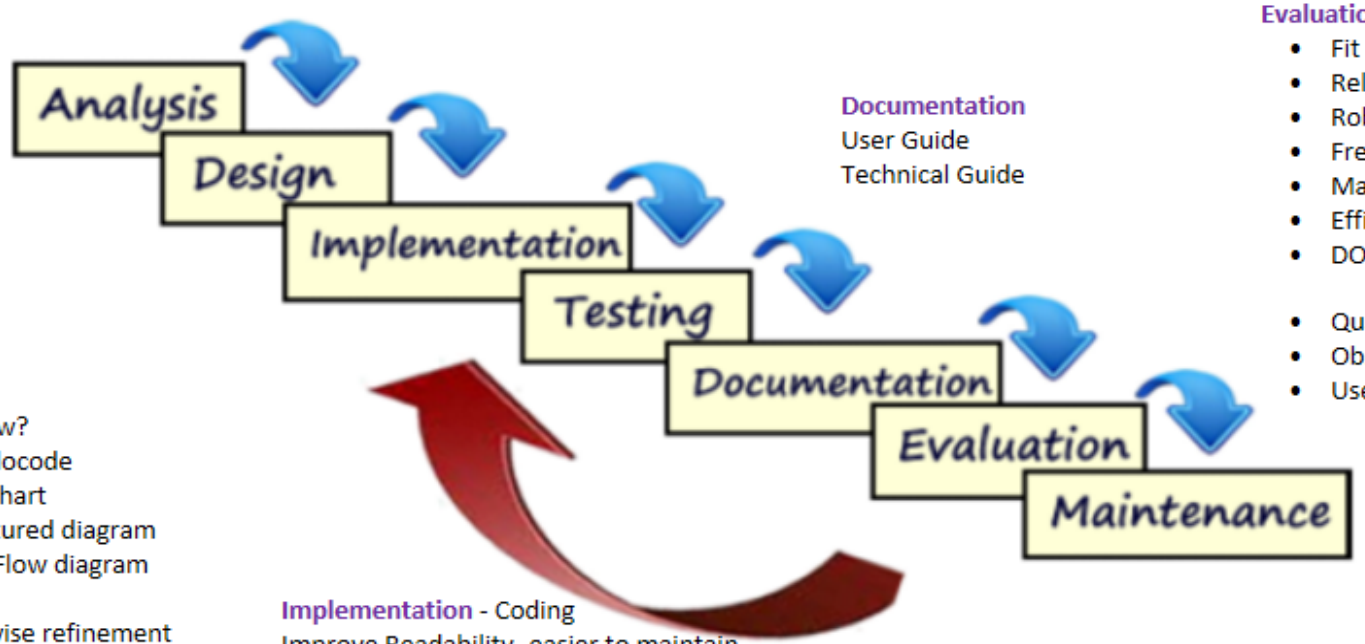
The **SDP** is **iterative** as any of the stages may be revisited throughout the project.

Example:

If an error is found in the maintenance stage (after release) then the implementation stage will be revisited to fix the code.

N5

ITERATIVE CYCLE - revisiting stages throughout the development.



Analysis
Fact finding exercise
What? Who? Where? When?
Limitations - Budget, Time
Systems Analyst would:
- Extract information
- Interview the client
- Observe the current working practices

Legally binding contract - software specification

Design - How?
- Pseudocode
- Flowchart
- Structured diagram
- Data Flow diagram

Wireframe
- Stepwise refinement
- Top Down Design

Implementation - Coding
Improve Readability- easier to maintain
- Indentation
- Annotation - internal commentary
- Meaningful variable names
- Passing parameters
- Modularity

Programming constructs
- Selection
- Iteration
- Variables
- Arrays
- Modules
- Standard algorithms
- File handling
- Predefined functions
- User defined functions

Testing
- Normal
- Extreme
- Exceptional

Systematic and robust
- Alpha
- Beta

Types of Error
- Syntax
- Logic
- Execution

Documentation
User Guide
Technical Guide

Evaluation
- Fit for purpose
- Reliable
- Robust
- Free from errors
- Maintainable
- Efficiency Nested Ifs > CASE
- DO While ... Do Loop Until

- Quantitive Data - Surveys etc.
- Observations & feedback
- User Interface

Maintenance
Adaptive
Corrective
Perfective

The Software Specification & Personnel

The **Software Specification** is a document that defines the scope and boundaries of the project and becomes **legally binding** between the client and the developers.

- **Client** - commissions the software project as a result of an identified 'need'.
- **Systems Analyst –** Investigates the client's current system and designs the new or upgraded system.
- **Project Manager –** Project leader who manages the project and the development team. Ensures deadlines being met and resources are in place.
- **Programmer –** Implements the designs (pseudocode etc.) into program code.
- **Independent Test Group –** tests the software to identify any errors by testing subroutines, modules and the completed program.

Analysis

The **Systems Analyst** meets with the client, **interviews** and **observes** users of the current system and identifies what the program has to do, in a process known as **Requirements Elicitation**.

The result is the production of the **Software Specification.**

Design

The design process is methodical and involves designing and representing **algorithms** as **pseudocode and/or structure diagrams.**

Design

Algorithm

An algorithm is a sequence of actions that, when performed in the correct order, will solve a problem or task in a finite number of steps.

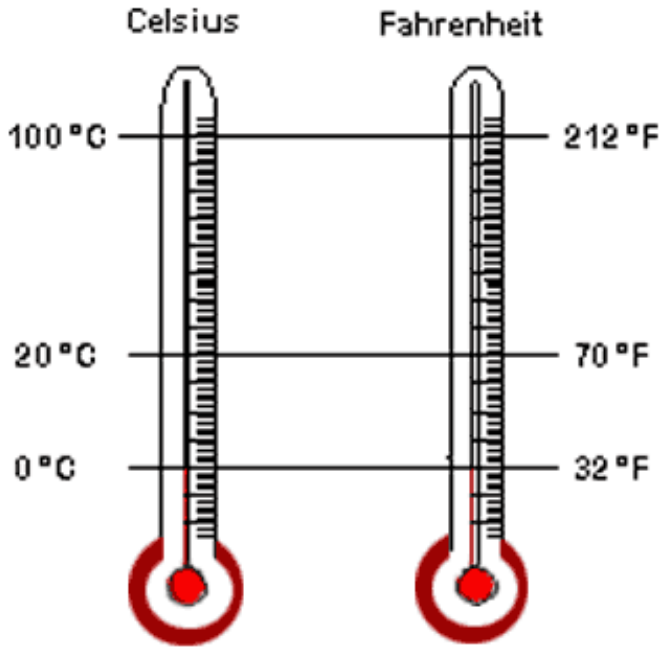
Example:

Here is a simple algorithm to convert a Fahrenheit temperature to Celsius:

1 Subtract 32 from Fahrenheit temperature
2 Multiply by 5/9ths to get Celsius temperature

So,

98.6°F - 32 = 66.6
66.6 * 5/9ths = 37°C

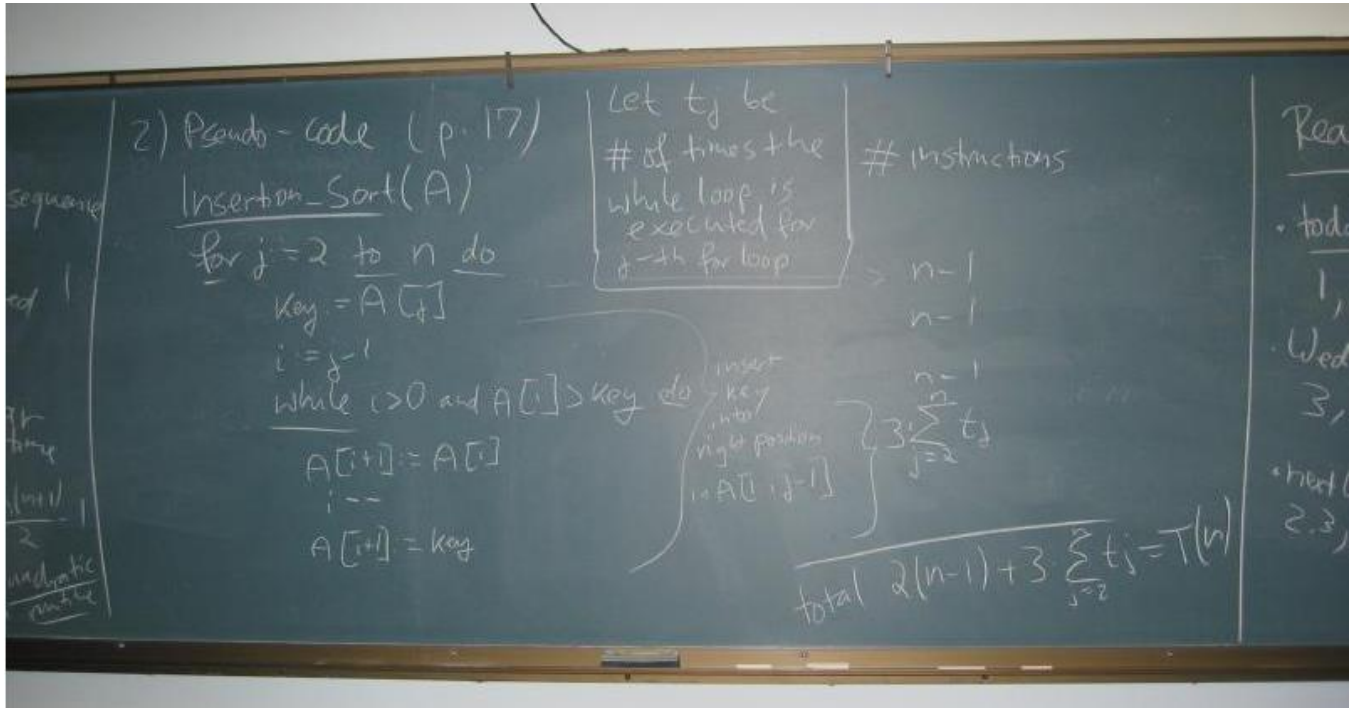


What happens if you swap steps 1 and 2 - do you get the same answer?

Design

Pseudocode

Pseudocode is a design notation that combines English language and Maths with programming language constructs.



Design

Pseudocode

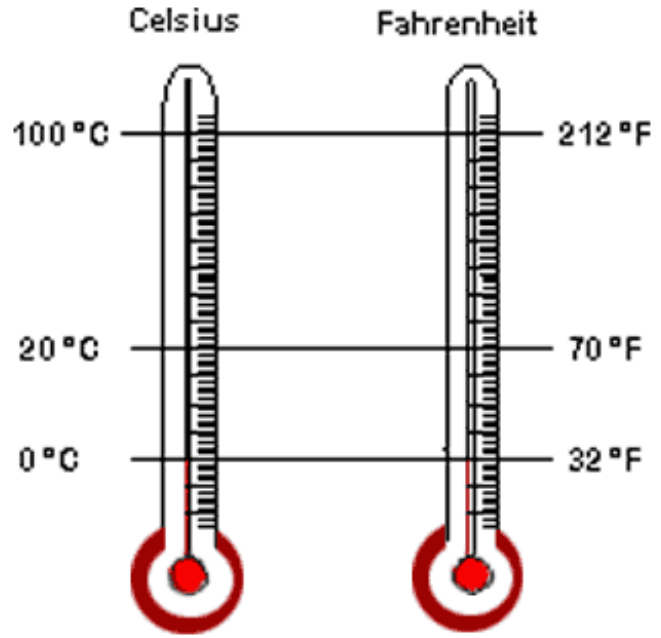
Here is the Fahrenheit conversion algorithm represented as pseudocode to be implemented as program code:

```
1.1 Get temp(°F)

1.2 Set temp(°F) = temp(°F)-32

1.3 Set temp(°C) = temp(°F)*0.556

1.4 Display temp(°C)
```



Note:

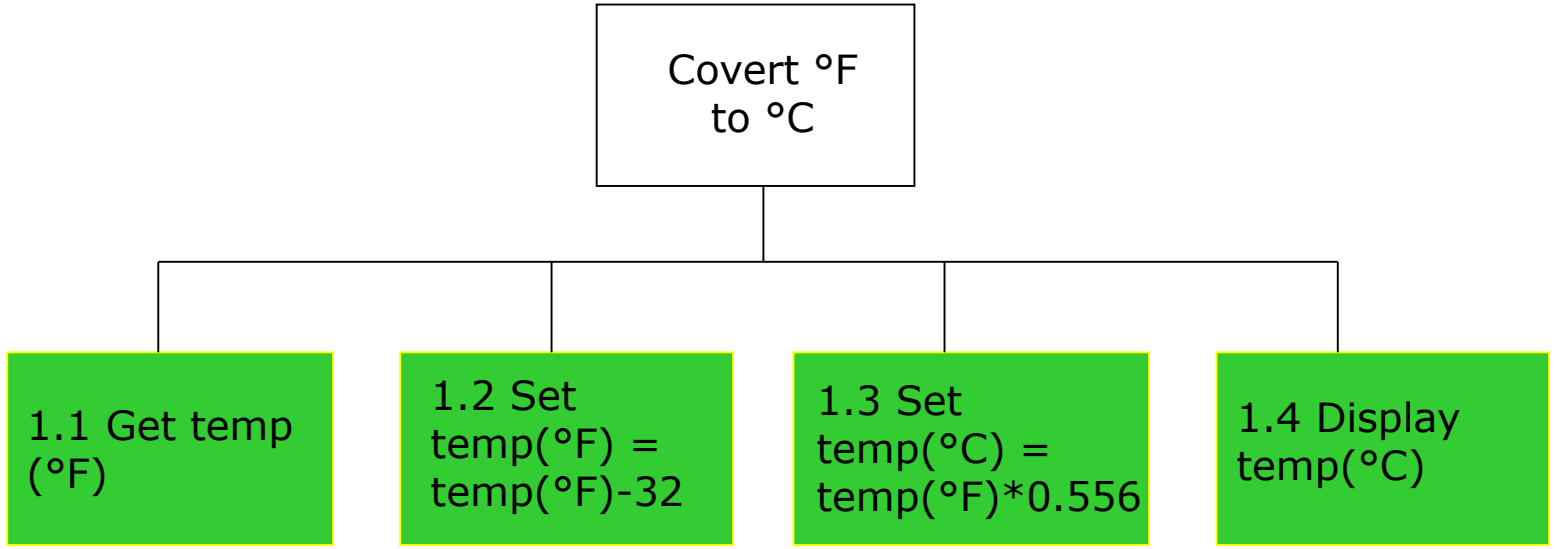
Pseudocode is **not** specific to any programming language. It is a design notation that can be implemented by programmers in **any** programming language

Design

Structure Diagram

A Structure Diagram is another design notation that represents algorithms in a chart or graphical form

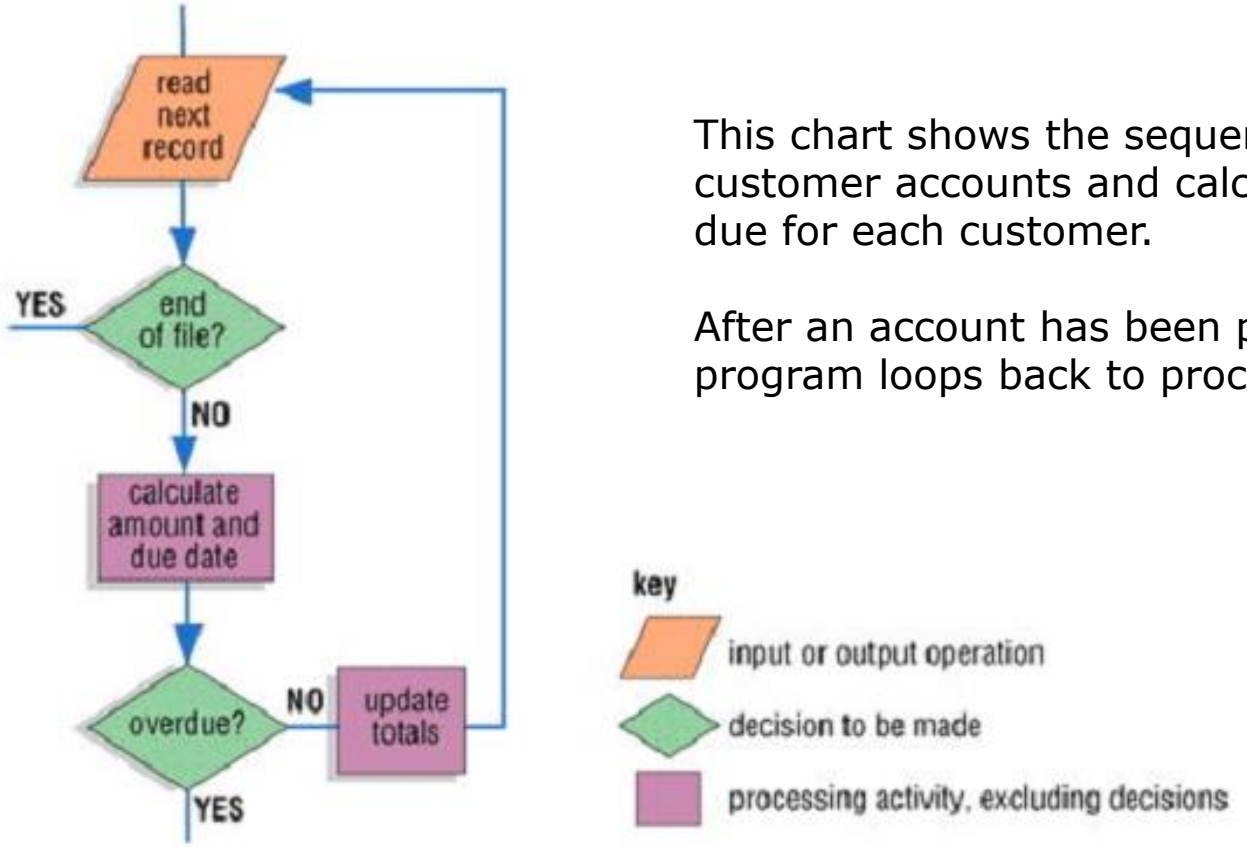
Here is the Fahrenheit conversion algorithm represented as a structure diagram to be implemented as program code:



Design

Flow chart

A **flow chart** is another design notation that shows the sequence of operations required to complete a task.



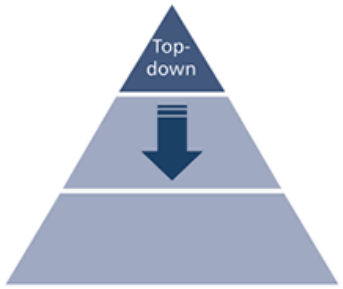
This chart shows the sequence of reading customer accounts and calculating the amount due for each customer.

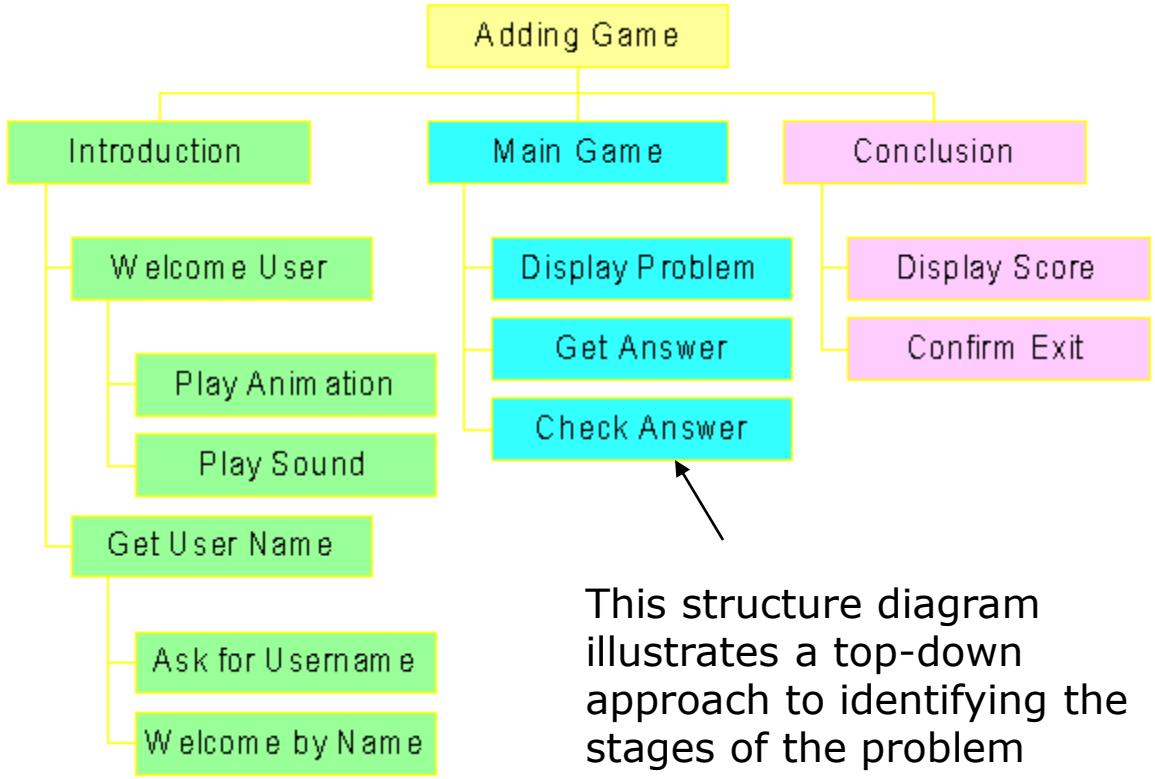
After an account has been processed, the program loops back to process the next one.

Design

Top-down design

Is the process of breaking a problem down in to sub-problems.



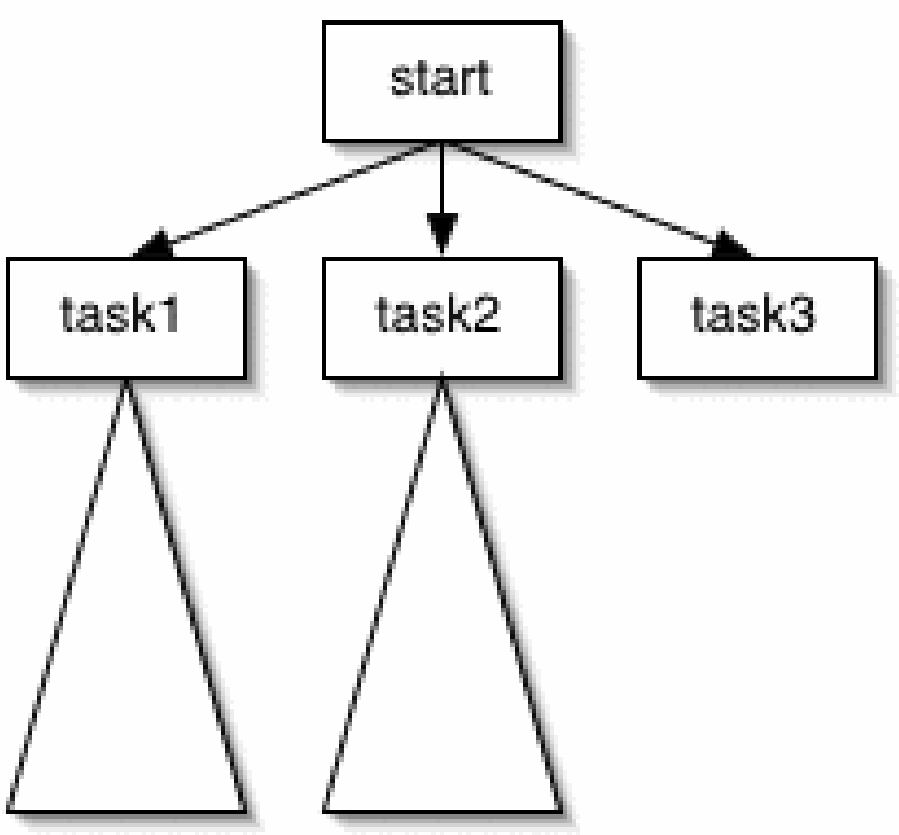


This structure diagram illustrates a top-down approach to identifying the stages of the problem solution.

Design

Stepwise Refinement

Stepwise-refinement is the process of refining stages down into steps. Each step can then be coded in a progr



Implementation

This is where the programmer(s), overseen by the project manager, convert the pseudocode (algorithms) into programming code e.g. Visual Basic, C++, Java etc.

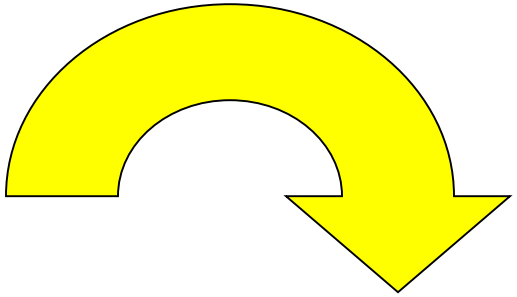
Pseudocode

1.1 Get temp(°F)

1.2 Set temp(°F) = temp(°F)-32

1.3 Set temp(°C) = temp(°F)*0.556

1.4 Display temp(°C)



A formatted printout of the program code is known as a **structured listing.** It includes indentation, white space and internal commentary.

```
'Program Code
tempF = txtTempF.Text
tempF = tempF-32
tempC = tempF*0.556
lblTempC = temp(°C)
```

N5

Testing

Testing should be both **systematic** and **comprehensive** i.e. methodical with **test reports** of predicted and actual results kept, and tested under all operational situations with a full range of test data.

The three types of testing are **normal, extreme** and **exceptional.**

Example

Program should only accept whole values in the range 0 to 100:

Normal data: 2, 34, 66 etc.

Extreme data: 0,100

Exceptional: -1, 156, abc, 2.9 etc.

Usability (beta) testing is when **independent test groups** and/or the client try out the software and report back any bugs to the development team prior to final release.

Evaluation

An **evaluation report** is prepared and supplied to the client with the release of the software.