Programing Methodology in C

Lecture -1 Course Overview

By Elubu Joseph MSc. IS

Tel: +256 773 086 497

Email: josebulinda@gmail.com

University: Kumi University

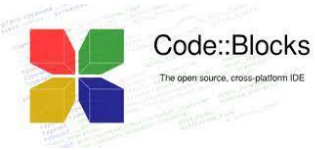
Website: www.kumiuniversity.ac.ug

Agenda

- Course Overview
- A glance on Generally Important terms in programing
- Introduction to programming languages

Course Description

This course is designed to introduce the learner(s) to the various programming concepts of the C language using the modern integrated development environment IDE.



The course introduces learners to the use of console application features, to appreciate and demonstrate the use of variables, constants, control flow, arithmetic, functions, input/output, and dynamic memory allocation.

Course Description +

It also introduces the various library functions available for use within the language compiler.

A high level programming language such as C is used to impart a discipline of structured/procedural programming into learner(s) so as to equip them with needed knowledge for the feature advance programming approaches such as object oriented programing done using various languages. The students cover both theoretical principles and hands on practical skills.

The objectives of the course are:

- To explain the fundamental properties of the C language.
- To introduce the learners to the basic concepts of algorithms, program structure and data structures.
- To discuss planning and organization processes in programming projects.
- To create strong practical basis in programming

Learning outcomes:

By the end of this course the student(s) should be able to;

- Explain the fundamental properties of the C language
- Understand the concepts of algorithms, program structure and data structures.
- Follow the planning and organization processes while programming projects
- Develop strong practical basis in programming

Audience & Prerequisites

Audience

This course is designed for anyone who wants to learn programming methodologies and how to use them to design solutions to a given problem.

Prerequisites

There are no mandatory prerequisites for this course except a desire to learn how to write good programs. However, having Elementary Mathematics knowledge may be an added advantage to the learner(s)

This course is designed to cater for both total beginners and those with some pogroming knowledge.

Course Requirements

1. Computer or Smartphone or Tablet
2. Connection to Internet
3. An Integrated Development Environment IDE, Code Blocks recommended

Assessment Criteria

- Course work/Tests 30%
- Final Examination 70%
- Total 100%

- Pass Mark is 50%

Textbooks

21st Century C, Ben, K; 2nd Edition, O'Reilly Media publishers, (2014)

Introduction to C Programming, Sass C. J, Volume 1 Wm. C. Brown Publishers, (1994);

Practical C Programming, Steve; 3rd Edition, Oualline, (1997),

C Programming: A Modern Approach, 2nd Edition, (1978). ISBN-13: 978-0393979503 ISBN-10: 0393979504

References

- O'Reilly Media, Inc, USA. (2014). *21St century C.*
- Oualline, S. (1997). Practical C Programming (3rd Edition, Ser. ISBN: 1-56592-306-5). O'Reilly Media, Inc.
- Wikimedia Foundation. (2021, June 21). *C17 (c standard revision)*. Wikipedia. Retrieved September 21, 2021, from https://en.wikipedia.org/wiki/C17_(C_standard_revision).
- Programming methodologies tutorial. (n.d.). Retrieved September 21, 2021, from https://www.tutorialspoint.com/programming_methodologies/index.htm.
- *Higher level and lower level languages*. Higher level and lower level languages - Computer Science Wiki. (n.d.). Retrieved September 21, 2021, from https://computersciencewiki.org/index.php/Higher_level_and_lower_level_languages.

References+

- *Difference between high level and low level languages*. GeeksforGeeks. (2020, May 13). Retrieved September 21, 2021, from https://www.geeksforgeeks.org/difference-between-high-level-and-low-level-languages/.
- Admin. (2021, June 15). *Types and differences between programming languages*. WatElectronics.com. Retrieved September 21, 2021, from https://www.watelectronics.com/types-of-programming-languages-with-differences/.
- AmitDiwan. (2021, March 2). Difference between high-level language and low-level language. Retrieved September 25, 2021, from https://www.tutorialspoint.com/difference-between-high-level-language-and-low-level-language.

Intro. to Programing Methodology (C)

is the approach used to analyze complex problems by planning the software development and controlling the development process (Programming methodologies tutorial. n.d.).

This helps when programs are developed to solve real-life problems like inventory management, payroll processing, student admissions, examination result processing, etc., they tend to be huge and complex. Imagine accessing your money from the bank while you are outside the building.

In this course, we shall use C programing language technics to come up with lines of codes that solve some basic problems.

Generally Important terms in programing

1. Programing - is the process of designing and building an executable computer program to accomplish a specific computing result

2. Programing language is a set of grammatical rules for instructing a computer or computing device to perform specific tasks. The term programming language usually refers to high-level languages, such as BASIC, C, C++, COBOL, Java, FORTRAN, Ada, and Pascal, python, C# etc.

3. A program/Software is a collection of instructions that can be executed by a computer to perform a specific task

4. Machine language is a computer programming language consisting of binary or hexadecimal instructions which a computer can respond to directly.

Generally Important terms programing +

5. Assembly language - is a type of low-level programming language that is intended to communicate directly with a computer's hardware. A language consisting mostly of symbolic equivalents of a particular computer's machine language.

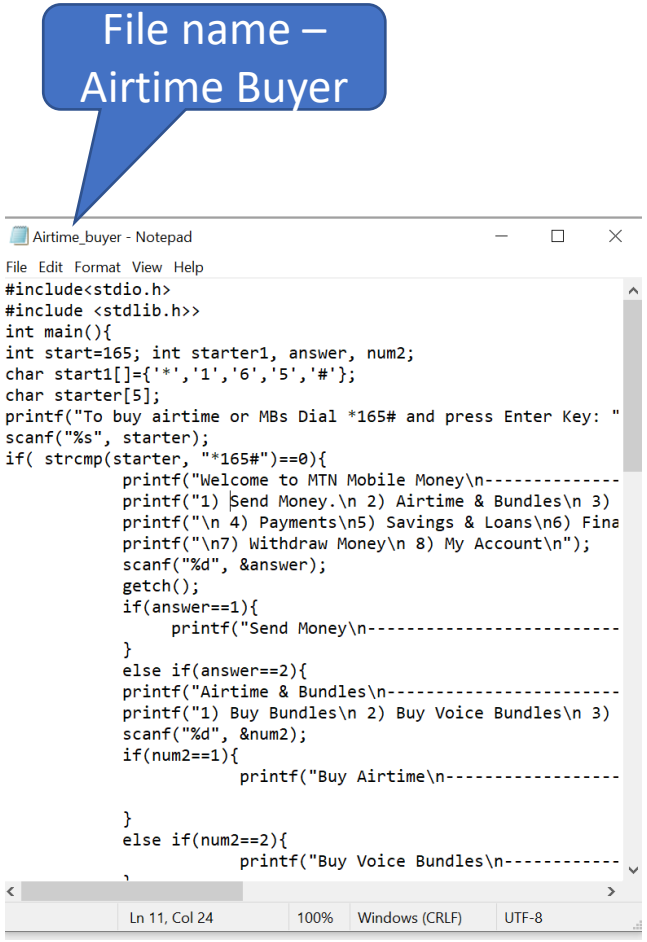
6. Source Code- a text listing of commands to be compiled or assembled into an executable computer program.

E.g.

```
#include<stdio.h>
#include <stdlib.h>>
int main(){
int start=165; int starter1, answer, num2;
char start1[]={'*','1','6','5','#'};
char starter[5];
printf("To buy airtime or MBs Dial *165# and press Enter Key: ");
scanf("%s", starter);
if( strcmp(starter, "*165#")==0){
            printf("Welcome to MTN Mobile Money\n-----------------------------------
            printf("1) Send Money.\n 2) Airtime & Bundles\n 3) MomoPay");
            printf("\n 4) Payments\n5) Savings & Loans\n6) Financial Services");
            printf("\n7) Withdraw Money\n 8) My Account\n");
            scanf("%d", &answer);
```

Generally Important terms programing ++

7. Source code file/ source file - is **a text file on disk**. It contains instructions for the computer that are written in the C programming language



8. **A source-code editor** is a text editor program designed specifically for editing source code of computer programs. It may be a standalone application or it may be built into an integrated development environment (IDE) or web browser.

Generally Important terms programing +++

9. Compiler - a program that converts instructions/Source code into a machine-code or lower-level form so that they can be read and executed by a computer.

10. Integrated development environment (**IDE**) is a software application that provides comprehensive facilities to computer programmers for software development. An IDE normally consists of at least a source code editor, build automation tools and a debugger. Some IDEs, such as Code Blocks, NetBeans and Eclipse, contain the necessary compiler, interpreter, or both; others, such as SharpDevelop and Lazarus, do not.

Generally Important terms programing +++

11. Debugger or **debugging tool** is a program used to test and debug other programs (the "target" program).

The main use of a debugger is to run the target program under controlled conditions that permit the programmer to track its operations in progress and monitor changes in computer resources (most often memory areas used by the target program or the computer's operating system) that may indicate malfunctioning code.

Need More??? Visit https://hackr.io/blog/programming-terms-definitions-for-beginners

Categories/Types of programing Languages

(Admin. 2021,).

These languages fall under different categories

Categories/Types of programing Languages +

1. Procedural Programming Languages
2. Functional Programming Languages
3. Object-oriented Programming Languages
4. Scripting Programming Languages
5. Logic Programming Language (Admin. 2021).

1. Procedural Programming Languages

The procedural programming languages are used to execute a sequence of statements which lead to a result.

Typically, these type of programming language uses multiple variables, heavy loops and other elements, which separates them from functional programming languages. Functions of procedural language may control variables, other than function's value returns. For example, printing out information (Admin. 2021, June 15).

E.g. C programing language

2. Functional Programming Languages

These typically uses stored data, frequently avoiding loops in favor of recursive functions.

The functional programing's primary focus is on the return values of functions, and side effects and different suggests that storing state are powerfully discouraged. For example, in an exceedingly pure useful language, if a function is termed, it's expected that the function not modify.

E.g. **Lisp, Python, Erlang, Haskell, Clojure, etc**.

3 Object-oriented Programming Languages

These programming languages views the world as a group of objects that have internal data and external accessing parts of that data.

One of the main principle of object oriented programming language is *encapsulation* that everything an object will need must be inside of the object.

These languages also emphasizes reusability through *inheritance* and the capacity to spread current implementations without having to change a great deal of code by using *polymorphism*.

E.g. Java, Python, C++, Lisp, and Perl.

4.Scripting Programming Languages

These programming languages are often procedural and may comprise object-oriented language elements, but they fall into their own category as they are normally not full-fledged programming languages with support for development of large systems.

For example, they may not have compile-time type checking. Usually, these languages require tiny syntax to get started.

E.g. Php, Java script etc.

5. Logic Programming Languages

These types of languages let programmers make declarative statements and then allow the machine to reason about the consequences of those statements.

In a sense, this language doesn't tell the computer how to do something, but employs restrictions on what it must consider doing.

5. Logic Programming Languages +

To call these groups ” types of language ” is really a bit confusing. It’s easy to program in an object-oriented style in C language. In truth, most of the languages include ideas and features from various domains, which only helps to increase the usefulness of these types of languages. Nevertheless, most of the programming languages do not best in all styles of programming (Admin. 2021, June 15).

E.g. Prolog, Answer Set Programming (ASP) and Datalog.

High level Programing Languages

Are programing languages that Programmers can easily understand or interpret or compile.

Many types of high-level language exist and are in common use today, including:

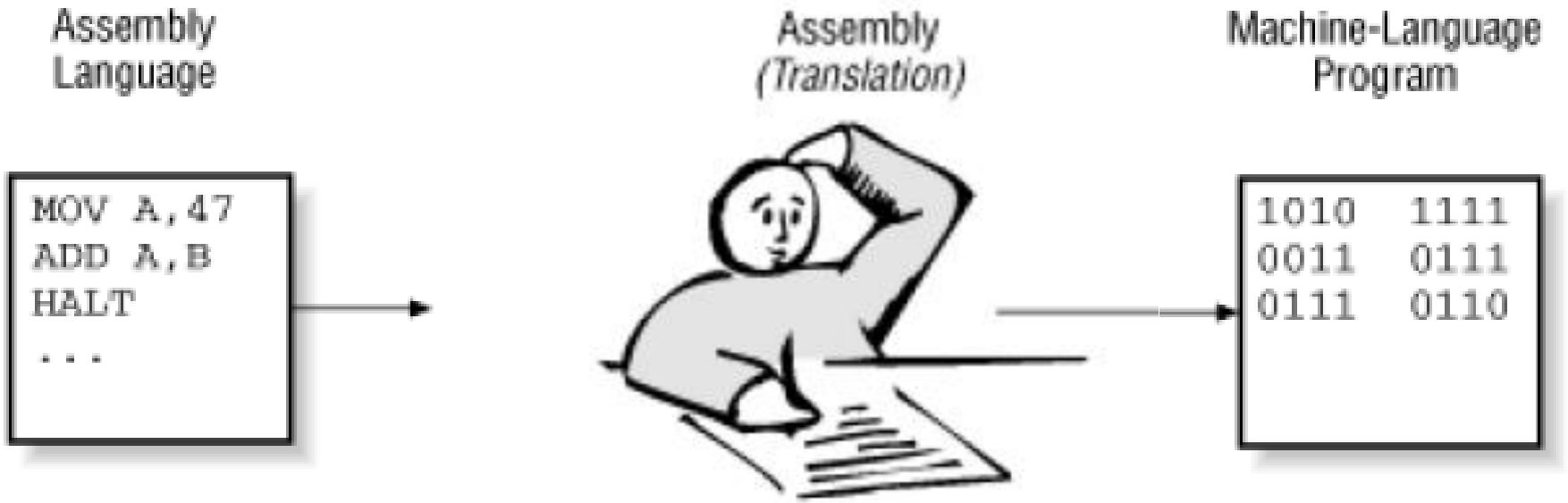
E.g. Python, Java., C++, C#, Visual Basic., JavaScript. And PHP etc.

Low Level Programing Languages

Also called machine Language are programing languages that are easily understood by machine. Sometimes referred to as Binary language.

Two examples of low-level languages are **assembly and machine code.**

Assembling a program to Machine Language



(Steve O, 1997)

Difference between high level and low level languages

S/N	High Level Language	Low Level Language
1.	It is programmer friendly language.	It is a machine friendly language.
2.	High level language is less memory efficient.	Low level language is high memory efficient.
3.	It is easy to understand.	It is tough to understand.
4.	It is simple to debug.	It is complex to debug comparatively.
5.	It is simple to maintain.	It is complex to maintain comparatively.

Difference between high level and low level languages +

S/N	High Level Language	Low Level Language
6	It is portable.	It is non-portable.
7.	It can run on any platform.	It is machine-dependent.
8.	It needs compiler or interpreter for translation.	It needs assembler for translation.
9.	It is used widely for programming.	It is not commonly used now-a-days in programming (AmitDiwan 2021).

Summary

In summary therefore, we looked at and learnt about:

1. Course Overview

i. Course Description

ii. Course Objectives

iii. Learning Outcomes

iv. Audience & Prerequisites

v. Course Requirements

vi. Assessment Criteria

vii. Textbooks

viii. References

Summary +

2. A glance on Generally Important terms in programing

i. Programing
ii. Programing language
iii. A program/Software
iv. Machine language
v. Assembly language
vi. Source Code
vii. Source code file/ source file
viii. A source-code editor
ix. Compiler
x. Integrated development environment (IDE)
xi. Debugger or debugging tool

Summary ++

3. Introduction to programming languages

i. Categories/Types of programing Languages

ii. High level Programing Languages

iii. Low Level Programing Languages

Revision Question

1. Define a programing language?

2. Differentiate between Program source code and Code editor.

3. With examples, explain the difference between procedural programing language and Object Oriented Programing languages.

4. What is programing as used in Computer Science?

Lecture References

- O'Reilly Media, Inc, USA. (2014). *21St century C.*
- Oualline, S. (1997). Practical C Programming (3rd Edition, Ser. ISBN: 1-56592-306-5). O'Reilly Media, Inc.
- Wikimedia Foundation. (2021, June 21). *C17 (c standard revision)*. Wikipedia. Retrieved September 21, 2021, from https://en.wikipedia.org/wiki/C17_(C_standard_revision).
- Programming methodologies tutorial. (n.d.). Retrieved September 21, 2021, from https://www.tutorialspoint.com/programming_methodologies/index.htm.
- *Higher level and lower level languages*. Higher level and lower level languages - Computer Science Wiki. (n.d.). Retrieved September 21, 2021, from https://computersciencewiki.org/index.php/Higher_level_and_lower_level_languages.

Lecture References+

- *Difference between high level and low level languages*. GeeksforGeeks. (2020, May 13). Retrieved September 21, 2021, from https://www.geeksforgeeks.org/difference-between-high-level-and-low-level-languages/.
- Admin. (2021, June 15). *Types and differences between programming languages*. WatElectronics.com. Retrieved September 21, 2021, from https://www.watelectronics.com/types-of-programming-languages-with-differences/.
- AmitDiwan. (2021, March 2). Difference between high-level language and low-level language. Retrieved September 25, 2021, from https://www.tutorialspoint.com/difference-between-high-level-language-and-low-level-language.

Next Lecture

- Overview of C programing language
- Features of C programing language
- Installation of IDE- Code Blocks
- My First C program
- Compile and Run C program

Thank you for you attention