

► **Activity 3:** Programming languages have evolved tremendously since the early 1950s and this evolution has resulted in hundreds of different languages being developed and used. The history of programming languages can be better understood if we classify them into different groups. One classification is into five 'generations', each of which is more programmer friendly, powerful, and at a higher level of abstraction of the computer hardware details than the previous one. This 'generational' classification is more or less abandoned now, as it is not considered particularly precise. Another way of grouping is into programming *paradigms*. The term *paradigm* refers to a set of design principles, according to which a given programming language is organized. Some languages fit in just one main paradigm, whereas others can be in more than one paradigm.

A) Use any source available (print or online) and draw a tree diagram of the two main programming paradigms (Imperative & Declarative) and their subgroups. Include subgroups such as: *Procedural, Logic(al), Mathematical, Object-Oriented, Functional*, etc.

B) The following table provides an overview of the history of programming languages. Match the languages with their descriptions:

a) The complexities of machine language led to the development of the next language. This language assigned a mnemonic code to each instruction, which enabled programmers to remember more easily how to write instructions to the computer. This language was translated into machine code by a separate program called <i>assembler</i>.	1. Machine Language
b) These languages are closer to human language as they have simple, English-like syntax rules. They are called so because they instruct the computer what to do rather than how to perform a task.	2. Assembly Language
c) This was a language that was very difficult to read and write, and could differ from computer to computer. It was binary, which means that the instructions and data that the processor was given to work on were written using 0s and 1s.	3. High-Level Procedural Languages
d) Assembly language required deep knowledge of computer architecture and programming skills. This led to the development of the next generation of languages, such as FORTRAN, BASIC, Pascal, and C. These languages described step by step the procedure to be followed to solve a specific problem. Programs written in these languages are converted into machine code with the use of compilers.	4. Non-Procedural Languages

► **Activity 4:** Which language is used in A., B., and C. below to say 'Hello, World'? Choose from: Python, C, and Basic.

A. _____
10 PRINT "Hello, World!"
20 END

B. _____
#include <stdio.h>
int main(void)
{
printf("hello, world\n");
}

C. _____
print("Hello
World")