ENCS 6020 Advanced Computing

Assignment 01

Name: Ibna Kowsar
Affiliation: MSCS
Submission Date: 02/13/2024

Q1.

Compiled programming language's implementation are translated into machine language, offering faster execution and optimization (i.e., C, C++). Interpreted languages run programs that are interpreted by another program known as an interpreter (PHP, Javascript). However, interpreted languages might run slower than compiled languages because the interpretation process adds overhead at runtime. Hybrid languages (i.e., Java or Python) compile source code to an intermediate form, such as bytecode, which is then interpreted or compiled just-in-time (JIT) into machine code by a virtual machine (JVM for Java, Python Interpreter for Python).

As we know compiled programming languages are closer to the computer hardware. Therefore, I would use compiled language over interpreted language when I need fast response for my system (more efficient).

Q2.

A programming language must be reliable to ensure that each step of the written program is correct and compatible. This reliability is also one of the main evaluation criteria for a programming language.

Type checking feature can be helpful to improves the reliability of a programming language. For instance, if a programming language automatically converts a variable to an integer even though the user is passing float type data, this could create issues in different application scenarios, such as tax calculation or weight calculation. Another example is if a user provides a string data type instead of an integer or float data type, the program should be able to detect the problem and throw an exception error. Therefore, having a type checking feature can help make a programming language more reliable.

The example below will throw an exception error in python programming language stating, "TypeError: unsupported operand type(s) for +: 'float' and 'str'". This example shows that it has a type checking which doest not convert '4.20' to 4.20 and instead does a type checking before each operation.

```
def doSum(prices):
    return sum(prices)

total_price = doSum([10.99, 15.49, 3.75, '4.20'])
print("Total Price:", total_price)
```

Q3.

```
# In C-programming language

struct Car {
char model[50];
int year;
float engineNumber;
}


int main (){
struct Car C1;
}
```

In this example Car is an abstraction, used for different instances of Car when needed without writing the same lines of code repetitively. A lot of computation can be done using a very small amount of coding which helps to improve the writability of a programming language.

Q4.

My preferred programming language is Python and the data types implementations are stated below,

a. Example for integer type,
    i. x = 6020
b. Example for string type,
    i. myString = “ENCS6020”
c. Example for character type,
    i. myChar = ‘C’
d. Example for floating type,
    i. y = 60.20
e. Example for boolean type,
    i. myBool = True
f. Example for arrays,
    i. One dimensional
        1. myArray1D = np.array([2, 3, 5])
    ii. Two dimensional
        1. myArray2D = np.array([[2,3,5],[7,11,13]])
g. Example of list as heterogenous array,
    i. myList = [“ENCS”,“6020”, “L”, “0.1”]
h. Example of enumeration type,
    i. tempEnum = enumerate(“TSU”, 5)
i. Example of associative array,
    i. myDict = {“Score” : {“TeamA” : 2}, {“TeamB” : 3}, {“TeamC” : 0}}

Q5.

In Java there is no built-in enumerate function like python. However, we can achiev similar result by using a loop and manually handling the index.

```
public class Task05 {
    public static void main(String[] args) {
        String[] names = {"John", "Bob", "Dave"};

        for (int k = 0; k < names.length; k++) {
            System.out.println(k + " " + names[k]);
        }
    }
}
```