North South University

Name: Md. Zahid Hasan Juel

ID: 1811767042

CSE425 sec: 09

Faculty: EZM

Assignment 3: “Design a new programming language”.

My Programming Language Name: "Toffy"

Type

"Toffy programming language will be Object oriented".

Will Contains 4 features,

1. Encapsulation
2. Inheritance
3. Polymorphism
4. Abstraction

Encapsulation:

Encapsulation refers to the creation of self-contained modules that bind processing functions to the data. These user-defined data types are called "classes," and one instance of a class is an "object." For example, in a payroll system, a class could be Manager, and Pat and Jan could be two instances (two objects) of the Manager class. Encapsulation ensures good code modularity, which keeps routines separate and less prone to conflict with each other.

Inheritance:

Classes are created in hierarchies, and inheritance allows the structure and methods in one class to be passed down the hierarchy. That means less programming is required when adding functions to complex systems. If a step is added at the bottom of a hierarchy, only the processing and data associated with that unique step needs to be added. Everything else is inherited. The ability to reuse existing objects is considered a major advantage of object technology.

Polymorphism:

Object-oriented programming allows procedures about objects to be created whose exact type is not known until runtime. For example, a screen cursor may change its shape from an arrow to a line depending on the program mode. The routine to move the cursor on screen in response to mouse movement would be written for "cursor," and polymorphism allows that cursor to take on whatever shape is required at runtime. It also allows new shapes to be easily integrated.

Abstraction: Abstraction is the concept of object-oriented programming that "shows" only essential attributes and "hides" unnecessary information. The main purpose of abstraction is hiding the unnecessary details from the users. Abstraction is selecting data from a larger pool to show only relevant details of the object to the user. It helps in reducing programming complexity and efforts. It is one of the most important concepts of OOPs.

Application domain of My language

Toffy programming language will be systems programming applications.

Systems programming application:
Systems software is used almost continuously and so a language
for systems programming must have fast execution.
a language used for systems programming must have low-level
features to access external devices.

Design Issue

Considering my programming language as an object-oriented programming language there are some design issues,

The Exclusivity of Objects:

Everything is an object

Advantage - elegance and purity.

Disadvantage - slow operations on simple objects.

Add objects to a complete typing system

Advantage - fast operations on simple objects

Disadvantage - results in a confusing type system (two kinds of entities).

Include an imperative-style typing system for primitives but make everything else objects.

Advantage - fast operations on simple objects and a relatively small typing system.

Disadvantage - still some confusion because of the two type systems.

Subclasses Subtypes:

If a derived class is-a parent class, then objects of the derived class must behave the same as the parent class object.

Subclass can only add variables and methods and override inherited methods in “compatible” ways.

Single and Multiple Inheritance:

- **Disadvantages of multiple inheritance:**

Language and implementation complexity (in part due to name collisions).

Potential inefficiency - dynamic binding costs more with multiple inheritance (but not much).

Allocation and De Allocation of Objects:

If they behave line the ADTs, they can be allocated from anywhere

Allocated from the run-time stack

Explicitly create on the heap (via new)

If they are all heap-dynamic, references can be uniform thru a pointer or reference variable

Simplifies assignment - dereferencing can be implicit

If objects are stack dynamic, there is a problem with regard to subtypes – *object slicing*

Dynamic and Static Binding:

If none are, you lose the advantages of dynamic binding

If all are, it is inefficient

Nested Classes:

In some cases, the new class is nested inside a subprogram rather than directly in another class

Initialization of Objects: Implicit or explicit initialization.

Criteria of my programming Language Toffy

Readability

The ease with which programs can be read and understood

Over Simplicity

A manageable set of features and constructs.

Minimal feature multiplicity.

Minimal operator overloading.

Orthogonality

A relatively small set of primitive constructs can be combined in a relatively small number of ways.

Every possible combination is legal.

Data types

Adequate predefined data types

Syntax considerations

Identifier forms: flexible composition

Special words and methods of forming compound statements

Form and meaning: self-descriptive constructs, meaningful keywords

“In that case Toffy programming language will be very easy to read and execute”.

Writability

Simplicity and orthogonality

Few constructs, a small number of primitives, a small set of rules for combining them.

Support for abstraction

The ability to define and use complex structures or operations in ways that allow details to be ignored.

Expressivity

A set of relatively convenient ways of specifying operations.

Strength and number of operators and predefined functions.

“In that case Toffy programming language will be very easy to write and execute”.

Reliability

Type checking

Testing for type errors.

Exception handling

Intercept run-time errors and take corrective measures.

Aliasing

Presence of two or more distinct referencing methods for the same memory Location.

“In that case Toffy programming language will be reliable”.

Cost

Training programmers to use the language will be within budget.

Writing programs will very easy.

Compiling programs: less difficulty

Executing programs: less difficulty

Language implementation system: availability of free compilers.

“So, the cost for Toffy programming language will be very cheap.”

The End