CME 3202 - Concepts of Programming Languages

Laboratory Worksheet#4

Laboratory Aim

In this laboratory section, you are expected to exercise examples of the fifth chapter in your lecture book - ***"Names, Bindings, and Scopes"***.

Dynamic type binding

Some programming languages apply dynamic type binding, in which the type of a variable is not specified by a declaration statement, nor can it be determined by the spelling of its name. Instead, the variable is bound to a type when it is assigned. ***"JavaScript and PHP"*** are examples of programming languages with dynamic type binding.

Exercise 1

"JavaScript" is a programming language with dynamic type binding. Apply the following steps, and implement a sample illustrating dynamic binding feature.

Step 1

Open the website - ***http://ideone.com/***. Select ***"JavaScript (spidermonkey)"*** programming language. Click on the text editor, and add the following lines. Then, click ***"Run"*** button.

```
list = [0.78, 2];
print(list[0]);
print(list[1]);

list= "Hello World!";
print(list);
```

Task 1

Paste the output of the code in the following.

```
0.78

2

Hello World!
```

Question 1

Can we write an equivalent code in ***"C#"*** programming language? What is the difference between ***"JavaScript"*** and ***"C#"*** considering type binding?

```
using System; //C# code

class Program {
   static void Main() {
      object list = new object[] { 0.78, 2 };
      Console.WriteLine(((object[])list)[0]);
      Console.WriteLine(((object[])list)[1]);

      list = "Hello World!";
      Console.WriteLine(list);
   }
}
```

Question 2

What are the advantages and disadvantages of dynamic type binding?

Advantages:
Allows changing variable types at runtime. No need to declare types explicitly. Works well in interpreted languages for rapid development.
Disadvantages:
Errors due to unexpected type changes. Type checking occurs at runtime. Harder to debug due to implicit type changes

Static scoping

In most of the programming languages including C, C++, and Java apply static (lexical) scoping, in which the scope of a variable can be determined statically by program text before execution and is independent of the run-time function call stack.

Exercise 2

"Ada" is a programming language with static scoping and nested subprograms. Apply the following steps, and implement a sample illustrating static scoping feature.

Step 1

Select ***"Ada"*** programming language. Click on the text editor, and add the following lines. Then, click ***"Run"*** button.

```
with Ada.Integer_Text_Io, Ada.Text_Io;
use Ada.Integer_Text_Io, Ada.Text_Io;

procedure ParentMethod is
   i: Integer :=1;
   j: Integer :=2;
   k: Integer :=3;
   procedure SubMethod1 is
      i: Integer :=4;
      j: Integer :=5;
      begin
         Put("i=");           1
         Put(i);
         New_Line;
      end;
   procedure SubMethod2 is
      k: Integer :=6;
      begin
         Put("i=");           2
         Put(i);
         New_Line;
      end;


   begin
      Put("ParentMethod");    3
      New_Line;
      Put("i=");
      Put(i);
      New_Line;
      Put("SubMethod1");
      New_Line;
      SubMethod1;
      Put("SubMethod2");
      New_Line;
      SubMethod2;
   end;
```

Task 1

Paste the output of the code in the following.

```
ParentMethod

i=1

SubMethod1

i=4

SubMethod2

i=1
```

Question 1

The referencing environment of a statement is the collection of all variables which are visible in the statement. Write the referencing environments of the specified code blocks into the given text boxes. **(Also, write the hidden values, but mark them as hidden.)**

Named constants

A named constant is a variable which is bound to a value only once mainly for achieving better readability and reliability. Some programming languages use static binding for these named constant, whereas some use dynamic binding.

Exercise 3

"C++" is a programming language which makes dynamic binding of named constants. Apply the following steps, and implement a sample illustrating named constants with dynamic binding.

Step 1

Select ***"C++"*** programming language. Click on the text editor, and add the following lines. Then, click ***"Run"***.

```
#include <iostream>
using namespace std;

int main() {
        int number=1;
        const int result = 5 * number + 5;

        cout << result << "\n";

        return 0;
 }
```

Task 1

Paste the output of the code in the following.

```
10
```

Task 2

Select ***"C#"*** programming language. Click on the text editor, and add the following lines. Paste the output of the code in the following.

```
using System;
public class Test
{
     public static void Main()
     {
          int number= 5;
          const int result = 5 * number + 5;
          Console.WriteLine(result);
     }
}
```

```
Compilation Error (CS0133: The expression being assigned to 'result' must be constant)
```

Question 1

Explain the differences between named constants of ***"C++"*** and ***"C#"*** as you saw in the examples above.

 C++ const:
Can be assigned values computed at runtime.More flexible in allowing variable expressions.
 C# const:
Must be assigned a compile-time constant. Prevents runtime computation in constant expressions.