VELAMMAL MAT. HR. SEC. SCHOOL, SURAPET, CHENNAI- 66
XII - COMPUTER SCIENCE - CHAPTER 14
IMPORTING C++ PROGRAMS IN PYTHON

PART-II

1. Theoretical Difference between Scripting and Programming Languages :

Scripting Language	Programming Language
A scripting language requires an interpreter	Programming language requires a compiler
A Scripting language need not be compiled	C++ program needs to be compiled before running
Example: JavaScript, VBScript, PHP, Perl, Python, Ruby, ASP and Tcl.	Example: C, C++, Java, C# etc.

2. Differentiate Compiler and Interpreter:

Compiler	Interpreter
Compiler generates an Intermediate Code.	Interpreter generates Machine Code.
Compiler reads entire program for compilation.	Interpreter reads single statement at a time for interpretation.
Error deduction is difficult	Error deduction is easy
Comparatively faster	Slower
Example:C, C++	**Example:** Python, Basic, Java

3. Write the expansion of (i) SWIG (ii) MinGW

- SWIG - Simplified Wrapper Interface Generator
- MinGW - *Minimalist GNU for Windows*

4. What is the use of modules? (OR) Define Module. Uses of Module:

- Modular programming is a software design technique to split your code into separate parts. These parts are called modules.
- Minimization of dependencies is the goal.
- Modules to break down large programs into small manageable and organized files.
- Modules provide reusability of code.
- We can define our most used functions in a module and import it, instead of copying their definitions into different programs.

5. What is the Use of cd command. Give an Example:

- Syntax for cd command: cd <absolute path>
- cd command refers to change directory and absolute path refers to the complete path where python is installed.
- For Example: “cd C:\Program Files\OpenOffiice 4\Program”.

PART - III

1. Differentiate Python and C++:

Python	C++
Python is typically an interpreted language	C++ is typically a compiled language
Python is a dynamic – typed language	C++ is compiled statically typed language
Data type is not required while declaring variable	Data type is required while declaring variable
It can act both as scripting and general purpose language	It is a general purpose language

2. What are the Applications of Scripting Languages?

- To automate certain tasks in a program
- Extracting information from a data set
- Less code intensive as compared to traditional programming language
- Can bring new functions to applications and glue complex systems together

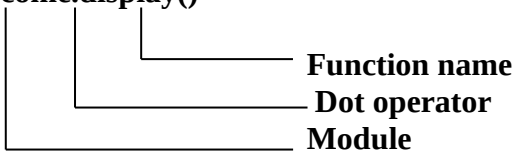
3. What is MinGW? What is its use?

MinGW - *Minimalist GNU for Windows*:

- MinGW refers to a set of runtime header files, used in compiling and linking the code of C, C++ and FORTRAN to be run on Windows Operating System.
- MinGw-W64 (version of MinGW) is the best compiler for C++ on Windows.
- To compile and execute the C++ program, you need 'g++' for Windows.
- MinGW allows to compile and execute C++ program dynamically through Python program using g++.

4. Identify the module, operator, definition name for the following

welcome.display()



5. What is sys.argv? What does it contain?

- **sys.argv**
- sys.argv is the list of command-line arguments passed to the Python program.
- argv contains all the items that come via the command-line input, it's basically a list holding the command-line arguments of the program.
- To use sys.argv, import sys should be used.
- The first argument, sys.argv[0] contains the name of the python program (example pali.py) and sys.argv[1] is the next argument passed to the program (here it is the C++ file).
- Which will be the argument passed through main ().
- For Example main(sys.argv[1])

PART - IV

1. Write any 5 Features of Python.

- Python uses Automatic Garbage Collection whereas C++ does not.
- C++ is a statically typed language, while Python is a dynamically typed language.
- Python runs through an interpreter, while C++ is pre-compiled.
- Python code tends to be 5 to 10 times shorter than that written in C++.
- In Python, there is no need to declare types explicitly where as it should be done in C++
- In Python, a function may accept an argument of any type, and return multiple values without any kind of declaration beforehand. Whereas in C++ return statement can return only one value.

2. Explain each word of the following command.

Python <filename.py> -i <C++ filename without cpp extension>

Python	**Keyword to execute the Python program from command line**
Filename.py	**Name of the Python program to executed**
-i	**Input mode**
C++ filename without cpp extension	**Name of C++ file to be compiled and executed**

3. What is the purpose of sys, os, getopt module in Python. Explain.

sys Module:

- This module provides access to builtin variables used by the interpreter. One among the variable in sys module is argv
- **sys.argv:**
- **sys.argv** is the list of command-line arguments passed to the Python program.
- **argv** contains all the items that come via the command-line input, it's basically a list holding the command-line arguments of the program.
- To use **sys.argv**, **import sys** should be used.
- The first argument, **sys.argv[0]** contains the name of the python program (example pali.py) and **sys.argv[1]** is the next argument passed to the program (here it is the C++ file). Which will be the argument passed through main (). For Example **main(sys.argv[1])**
- The input file (C++ file) is send along with its path as a list(array) using **argv[1]**.
- **argv[0]** contains the Python program which is need not to be passed because by default **__main__** contains source code reference

os Module:

- The functions that the ***OS*** module allows you to interface with the Windows operating system where Python is running on.
- **os.system():** Execute the C++ compiling command (a string contains Unix, C command which also supports C++ command) in the shell (Here it is Command Window).
- For Example to compile C++ program **g++** ***compiler*** should be invoked.
- To do so the following command is used.
- ***os.system ('g++' + <varaiable_name1> + '-<mode>' + <variable_name2>)***

os.system :-	function system() defined in **os** module to interact with the operating system
g++ :-	General compiler to compile C++ program under Windows Operating system.
variable_name1:-	Name of the C++ file along with its path and without extension .cpp in string format
mode :-	To specify input or output mode. Here it is o prefixed with hyphen.
variable_name2 :-	Name of the executable file without extension .exe in string format

For example the command to compile and execute C++ program is given below

os.system('g++ ' + cpp_file + ' -o ' + exe_file)	g++ compiler compiles the file cpp_file and –o (output) send to exe_file

'+' in os.system() indicates that all strings are concatenated as a single string Therefore give a space after each word for the above argument.
For example: 'g++ ' + cpp_file + ' -o ' + exe_file

getopt Module:

- The **getopt** module of Python helps you to parse (split) command-line options and arguments. This module provides two functions to enable command-line argument parsing.
- **getopt.getopt method**
- This method parses command-line options and parameter list.
- **Following is the syntax for this method −**
- ***<opts>,<args>=getopt.getopt(argv, options, [long_options])***
- **argv** − This is the argument list of values to be parsed (splited). In our program the complete command will be passed as a list.
- **For example** ***c:\pyprg\pali.py -i c:\pyprg\pali_cpp***
- **options** − This is string of option letters that the Python program recognize as, for input or for output, with options (like 'i' or 'o') that followed by a colon (:). Here colon is used to denote the mode.
- **long_options** −This contains a list of strings. Argument of Long options should be followed by an equal sign ('=').
- In our program the C++ file name along with its path will be passed as string and 'i' i will be also passed to indicate it as the input file.
- **getopt()** method returns value consisting of two elements.
- Each of these values are stored separately in two different list (arrays) **opts and args** .
- **opts** contains list of splitted strings like mode and path.
- **args** contains error string, if at all the comment is given with wrong path or mode.
- **args** will be an empty list if there is no error.
- **opts, args = getopt.getopt (argv, "i:",['ifile='])**

where opts contains	[('-i', 'c:\\pyprg\\p4')]
-i :-	**option - mode should be followed by : (colon)**
'c:\\pyprg\\p4'	**value - absolute path of C++ file.**

4. Write the syntax for getopt() and explain its arguments and return values.

- **syntax for getopt() method −**
- ***<opts>,<args>=getopt.getopt(argv, options, [long_options])***
- **argv** − This is the argument list of values to be parsed (splited). In our program the complete command will be passed as a list.
- **For example** ***c:\pyprg\pali.py -i c:\pyprg\pali_cpp***
- **options** − This is string of option letters that the Python program recognize as, for input or for output, with options (like 'i' or 'o') that followed by a colon (:). Here colon is used to denote the mode.
- **long_options** −This contains a list of strings. Argument of Long options should be followed by an equal sign ('=').
- In our program the C++ file name along with its path will be passed as string and 'i' i will be also passed to indicate it as the input file.
- **getopt()** method returns value consisting of two elements.
- Each of these values are stored separately in two different list (arrays) **opts and args** .
- **opts** contains list of splitted strings like mode and path.
- **args** contains error string, if at all the comment is given with wrong path or mode.
- **args** will be an empty list if there is no error.

5. Write a Python program to execute the following c++ coding

```
#include <iostream>
using namespace std;
int main()
{ cout<<"WELCOME";
```

```
        return(0);
        }
```

The above C++ program is saved in a file welcome.cpp

- *#Now select File→New in Notepad and type the Python program*
- *# Save the File as wel.py . Program that compiles and executes a .cpp file*
- *# Python c:\pyprg\wel.py -i c:\pyprg\welcome*

```
import sys, os, getopt
def main(argv):
        opts, args = getopt.getopt(argv, "i:")
        for o, a in opts:
                if o in "-i":
                        run(a)
def run(a):
        inp_file=a+'.cpp'
        exe_file=a+'.exe'
        os.system('g++ ' + inp_file + ' -o ' + exe_file)
        os.system(exe_file)
#program starts executing from here
if __name__=='__main__':
        main(sys.argv[1:])
```

OUTPUT:

```
WELCOME
```

EXTRA QUESTION ANSWER:

Scripting Language:

- A scripting language is a programming language designed for integrating and communicating with other programming languages.
- Some of the most widely used scripting languages are JavaScript, VBScript, PHP, Perl, Python, Ruby, ASP and Tcl.

Garbage Collection:

- Python deletes unwanted objects (built-in types or class instances) automatically to free the memory space.
- The process by which Python periodically frees and reclaims blocks of memory that no longer are in use is called Garbage Collection.

Wrapping:

Importing C++ program in a Python program is called wrapping up of C++ in Python.

Wrapping or creating Python interfaces for C++ programs. The commonly used interfaces are:

- Python-C-API (API-**Application Programming Interface** for interfacing with C programs)
- Ctypes (for interfacing with c programs)
- SWIG (Simplified Wrapper Interface Generator- Both C and C++)
- Cython (Cython is both a Python-like language for writing C-extensions)
- Boost. Python (a framework for interfacing Python and C++)
- MinGW (*Minimalist GNU for Windows*)

Executing C++ Program through Python:

1. Double click the run terminal of MinGW
2. Go to the folder where the Python software is located (Python.exe) is located.
3. In this example "Python folder" is located in **C:\Program Files\OpenOffiice 4\Python.**
4. The syntax to execute the Python program is

How to import modules in Python?

- We can import the definitions inside a module to another module. We use the **import** keyword to do this.
- To import our previously defined module **factorial** we type the following in the Python prompt.

  >>> **import factorial**

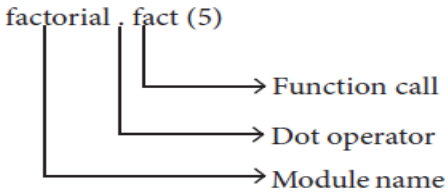
How to access function from the module:

- The dot (.) operator is used to access the functions from the module.

- The syntax for accessing the functions from the module is

  <module name> . <function name>

  For Example:



__name__ (A Special variable) in Python :

- **__name__ is one such special variable which by default stores the name of the file.** If the source file is executed as the main program, the interpreter sets the __name__ variable to have a value as "__main__".
- **__name__ is a built-in variable which evaluates to the name of the current module.** Thus it can be used to check whether the current script is being run on its own.
- For example consider the following

```
if __name__ == '__main__':
    main (sys.argv[1:])
```

Python program Executing C++ Program using control statement:

- The steps for executing the C++ program to check a given number is palindrome or not.
- Step1: Type the C++ program to check whether the input number is palindrome or not in notepad and save it as **"pali_cpp.cpp".**
- Step2: Type the Python program and save it as pali.py
- Step3: Click the Run Terminal and open the command window
- Step4: Go to the folder of Python using cd command.
- Step5: Type the command Python pali.py -i pali_cpp

Example:- 14.7.1 - Write a C++ program to enter any number and check whether the number is palindrome or not using while loop.

```
/*. To check whether the number is palindrome or not using while loop.*/
//Now select File->New in Notepad and type the C++ program
#include <iostream>
using namespace std;
int main()
{
    int n, num, digit, rev = 0;
    cout<< "Enter a positive number: ";
    cin>>num;
    n = num;
    while(num)
    {
        digit = num % 10;
        rev = (rev * 10) + digit;
        num = num / 10;
    }
cout<< " The reverse of the number is: " << rev <<endl;
if (n == rev)
    cout<< " The number is a palindrome";
else
    cout<< " The number is not a palindrome";
return 0;
}
// Save this file as pali_cpp.cpp

#Now select File→New in Notepad and type the Python program
# Save the File as pali.py . Program that compiles and executes a .cpp file
# Python c:\pyprg\pali.py -i c:\pyprg\pali_cpp
import sys, os, getopt
def main(argv):
        opts, args = getopt.getopt(argv, "i:")
        for o, a in opts:
                if o in "-i":
                        run(a)
def run(a):
        inp_file=a+'.cpp'
        exe_file=a+'.exe'
        os.system('g++ ' + inp_file + ' -o ' + exe_file)
        os.system(exe_file)
#program starts executing from here
if __name__=='__main__':
        main(sys.argv[1:])
```

Output of the above program:

```
Enter a positive number: 56765
The reverse of the number is: 56765
The number is a palindrome
```

Output of the above program:

Enter a positive number: 56756
The reverse of the number is: 65765
The number is not a palindrome

Choose the best answer (1 Marks)

1. Which of the following is not a scripting language?
(A) JavaScript
(B) PHP
(C) Perl
(D) HTML
2. Importing C++ program in a Python program is called
(A) wrapping
(B) Downloading
(C) Interconnecting
(D) Parsing
3. The expansion of API is
(A) Application Programming Interpreter
(B) Application Programming Interface
(C) Application Performing Interface
(D) Application Programming Interlink
4. A framework for interfacing Python and C++ is
(A) Ctypes
(B) SWIG
(C) Cython
(D) Boost
5. Which of the following is a software design technique to split your code into separate parts?
(A) Object oriented Programming
(B) Modular programming
(C) Low Level Programming
(D) Procedure oriented Programming
6. The module which allows you to interface with the Windows operating system is
(A) OS module
(B) sys module
(C) csv module
(D) getopt module
7. getopt() will return an empty array if there is no error in splitting strings to
(A) argv variable
(B) opt variable
(C)args variable
(D) ifile variable
8. Identify the function call statement in the following snippet.
if __name__ =='__main__':
main(sys.argv[1:])
(A) main(sys.argv[1:])
(B) __name__
(C) __main__
(D) argv
9. Which of the following can be used for processing text, numbers, images, and scientific data?
(A) HTML
(B) C
(C) C++
(D) PYTHON

10. What does __name__ contains ?
(A) c++ filename
(B) main() name
(C) python filename
(D) os module name
11. **Python and C++** are general-purpose programming language.
12. Python is mostly used as a **scripting or "glue"**, language.
13. The top level program mostly calls routines written in **C or C++**.
14. API - **Application Programming Interface**
15. SWIG - Simplified Wrapper Interface Generator
16. MinGW - Minimalist GNU for Windows
17. **MinGw-W64** (version of MinGW) is the best compiler for C++ on Windows.
18. To compile and execute the C++ program, you need **'g++'** for Windows.
19. **MinGW** allows to compile and execute C++ program dynamically through Python program using g++.
20. **g++** is a program that calls **GCC (GNU C Compiler)** and automatically links the required C++ library files to the object code.
21. In the execution command, the **input file** doesn't require its extension.
22. To clear the screen in command window use cls command.
23. **Modular programming** is a software design technique to split your code into separate parts.
24. **Modules** refer to a file containing Python statements and definitions.
25. The **dot (.)** operator is used to access the functions.
26. **'+' in os.system()** indicates that all strings are concatenated as a single string and send that as a List.
27. There is no **main() function** in Python.
28. **__main__** contains the **name of that Python program** and the Python special variable **__name__** also contain the **Python program name**.
29. **C++** is a compiler based language while **Python** is an interpreter based language.
30. **C++** is compiled statically whereas **Python** is interpreted dynamically
31. **sys module** provides access to some variables used by the interpreter and to functions that interact with the interpreter
32. **OS module** in Python provides a way of using operating system dependent functionality
33. The **getopt module** of Python helps you to parse (split) command-line options and arguments

CHOOSE THE CORRECT ANSWER:
1. Which of the following are general purpose programming languages?
a. Python
b. C++
c. Java
d. All of these
2. Which of the following is not general purpose language?
a. Python
b. C++
c. Java
d. Perl
3. Which of the following is not a compiled statically typed language?
a. Python
b. C++
c. Java

d. All of these
4. Which of the following language used automatic garbage collection?
a. C++
b. C
c. Python
d. Java
5. Which of the following interface used for interfacing with C programs?
a. MinGW
b. Boost
c. Ctypes
d. Cython
6. SWIG expansion is
a. Simplified Wrapper Interface Generator
b. Software Wrapper Information Generator
c. Single Wrapper Interface Generator
d. System Wrapper Interface Generator
7. Which of the following python interface used for both C and C++
a. MinGW
b. Ctypes
c. Cython
d. SWIG
8. MinGW expansion is
a. Minimalist Graphics for windows
b. Minimum GNU for windows
c. Minimalist GNU for windows
d. Motion Graphics for windows
9. Which of the following is needed to run a C++ program on windows
a. m++
b. g++
c. ghre++
d. f++
10. The command to change to the folder where python is located is
a. Change
b. cd
c. Dir
d. CDir
11. The syntax to execute the python program is
a. python –i <filename.py> <C++ filename>
b. python <filename –py> <C++ filename> -i
c. python <C++ filename> -i <filename.py>
d. python <filename.py> -i <C++ filename>
12. In the command python <filename.py> -i <C++ filename> where i denotes
a. Information
b. Input mode
c. ios
d. Interpreter
13. Which of the following is not a python module?
a. os
b. sys
c. getopt
d. tcl

14. Which of the following is an array holding the command line arguments of the program?
a. g++
b. argv
c. opts
d. getopt
15. getopt() method returns values are started in
a. opts
b. args
c. sys
d. both a and b
16. Which symbol in os.system() indicates that all strings are concatenated and send that as a list
a. +
b. -
c. ()
d. *
17. Which one of the following is a special variable by default stores the name of the file
a. __name__
b. __os__
c. __sys__
d. __argv__
18. Which method of os module executed the exe file to get the desired output
a. main()
b. name()
c. run()
d. system()
19. Choose the incorrect statement from the following.
i. C++ program needs to be compiled before running
ii. Python need to be compiled
iii. perl, ruby, asp are the scripting language
iv. Python is not high level general purpose programming language
a. i and ii
b. ii and iii
c. ii and iv
d. i, ii and iv
20. Which allows to compile and execute C++ program dynamically through python program using g++
a. MinGW
b. Ctypes
c. Boost
d. SWIG

***** Best of Luck *****