Programming Languages

Lesson Outcomes

At the end of this lesson, you should be able to:

- Distinguish between the low-level machine-language and high-level programming language
- Describe the two ways of translating high-level language program code to machine-language instructions
- State the main features of high-level programming languages: C, C++, Java, Python

Programming Language

Programming is the process of

- implementing a representation of the solution for the computer to execute
- taking an algorithm and encoding it using certain programming language

Programming language

- a medium through which programmer may give instructions to a computer

A programming language

- must support certain control constructs and data types needed to implement algorithms

Examples

- Sequential processing
- Decision-making selection
- Iteration for repetitive execution

- Numbers such as integer and real (floating point)
- Characters and Strings

Types of Programming Language

The computer has its own language.

- machine-language that is specific to the type of computer processor
- consists of a set of machine instructions and data objects that are encoded in ‘0’ and ‘1’ binary format
- error-prone for human to code a program directly using the computer’s machine codes

Types of Programming Language (Cont'd)

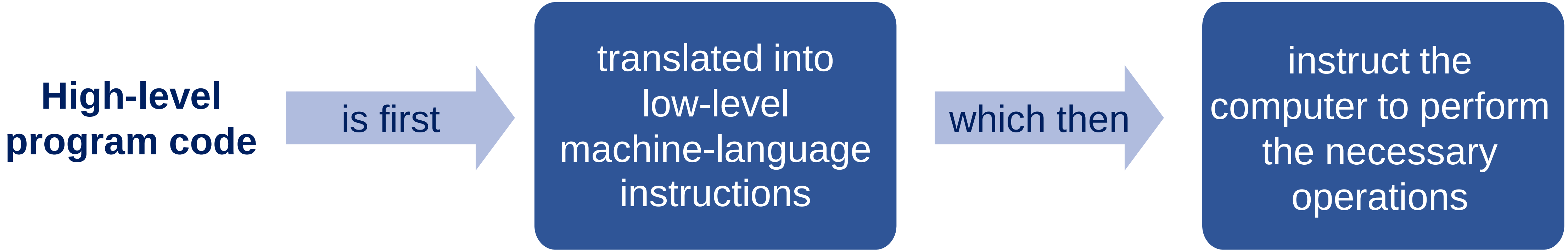
High-level programming languages that are more user-friendly are hence invented.

- provide abstraction from the internal operating detail of the computer
- enable the programmers to focus on solving the problem
- make the process of developing the algorithm simpler

Most programs used in modern computers

- coded in high-level programming languages, such as C, Java, Python, etc.

Program Translations



High-level program code

is first

translated into low-level machine-language instructions

which then

instruct the computer to perform the necessary operations

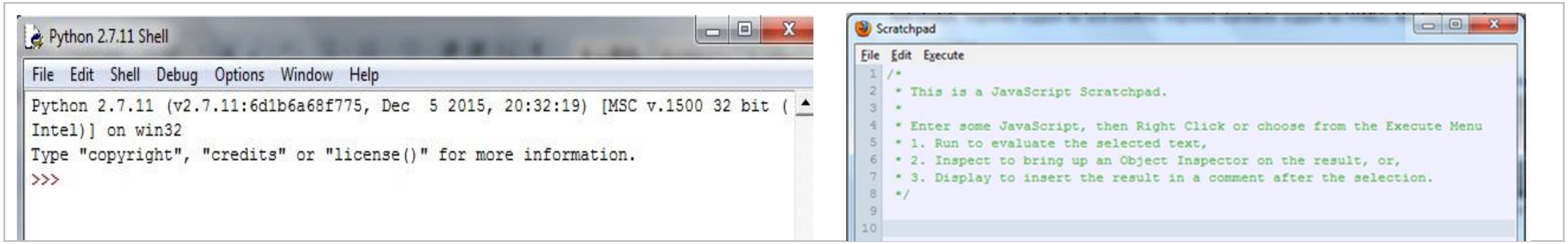
Two approaches for program translation:

- interpretation
- compilation

Program Translations: Interpretation Approach

Interpretation approach

- uses a program known as **interpreter**
- reads one high-level code statement at a time
  - immediately translates and executes the statement before processing the next one
- examples: Python, R, and JavaScript



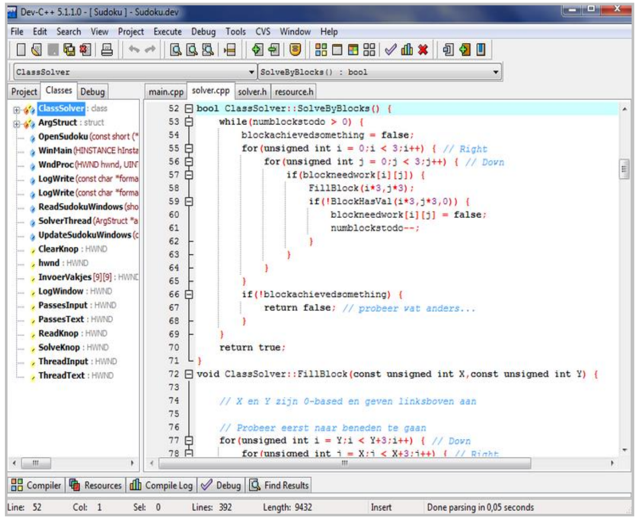
```
Python 2.7.11 (v2.7.11:6d1b6a68f775, Dec  5 2015, 20:32:19) [MSC v.1500 32 bit (
Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
```

```
/*
 * This is a JavaScript Scratchpad.
 *
 * Enter some JavaScript, then Right Click or choose from the Execute Menu
 * 1. Run to evaluate the selected text,
 * 2. Inspect to bring up an Object Inspector on the result, or,
 * 3. Display to insert the result in a comment after the selection.
 */
```

Program Translation: Compilation Approach

Compilation approach

- uses a program called **compiler**
- reads and translates the entire high-level language program (source) code into its equivalent machine-language instructions in an executable file
- the resulting machine-language instructions can then be executed directly on the computer when the program is launched
- examples: C and C++



```
bool ClassSolver::SolveByBlocks() {
    while(numblockstodo > 0) {
        blockachievedsomething = false;
        for(unsigned int i = 0;i < 3;i++) { // Right
            for(unsigned int j = 0;j < 3;j++) { // Down
                if(blockneedwork[i][j]) {
                    FillBlock(i*3,j*3);
                    if(!BlockHasVal(i*3,j*3,0)) {
                        blockneedwork[i][j] = false;
                        numblockstodo--;
                    }
                }
            }
        }
        if(!blockachievedsomething) {
            return false; // probeer wat anders...
        }
    }
    return true;
}
void ClassSolver::FillBlock(const unsigned int X,const unsigned int Y) {

    // X en Y zijn 0-based en geven linksboven aan

    // Probeer eerst naar beneden te gaan
    for(unsigned int i = Y;i < Y+3;i++) { // Down
        for(unsigned int j = X;j < X+3;j++) { // Right
```

Line: 52 Col: 1 Sel: 0 Lines: 392 Length: 9432 Insert Done parsing in 0,05 seconds

It is also possible to use the **combination of both** translation techniques

- example: Java

Interpreter vs. Compiler

Interpreter

- very portable across different computing platforms
- produces results almost immediately
- easy to debug
- program executes more slowly
- useful for implementing dynamic, interactive features, such as those used on web pages

Compiler

- program runs very fast AFTER compilation
- smaller in code size after compilation
- must compile the entire program before execution
- needs to be re-compiled if to be used on different computing platforms
- used in large and sophisticated software applications when speed is of the utmost importance

Programming Language: C

C language

- originally developed to help implement the Unix operating system
- allow for direct access of, and manipulation of, the underlying computer’s hardware

It is a compiled language

- the program code (i.e. source code) needs to be compiled first, before it can be executed

Most suitable for applications

- direct access to the computer’s hardware
- fast real-time response

Examples of applications

- embedded systems development
- operating systems implementations
- device drivers that control the hardware inside the computer

Programming Language: C++

C++ language

- an extension to the C language
- with additional support for Object-Oriented Programming (OOP)

In OOP

- programs are designed based on objects that contain attributes and behaviors
- programming is then focused on how the objects interact with one another
  - without the need to know the internal code detail of the object

C++ is a compiled language

- used for applications that need high performance and are based on object oriented software design principles

Examples of Applications

- Video games
- Graphic User Interface (GUI) based applications
- Latest operating systems design and implementation

Programming Language: Python

Python language

- a relatively easy-to-learn programming language
- targeted for general purpose programming for applications that do not need direct access to the computer's hardware

It is an interpreted language

- program code is hence portable across different computing platforms

Python interpreter converts and executes the Python program statement one-by-one to the corresponding machine instructions

- provides instant feedback to the user at runtime
- very helpful for learning to program

python™

Examples of Applications

- Data science
- Web developments

Programming Language: Java

Java language

- highly portable, general purpose language with Object-Oriented (OO) methodological support

Designed to run as an interpreted language on a Java Virtual Machine (JVM)

- portable across different types of machines and devices
- no direct access to the computer’s hardware

Combination of compiler and interpreter, as well as Just-in-time (JIT) compiler

- the source code (with file extension .java) is first compiled to Java bytecode (with file extension .class)
- Java bytecode instruction is then interpreted by the JVM interpreter during execution
- JVM may call the JIT compiler to compile some of the bytecode instructions at runtime before they are executed, achieving better performance

Examples of Applications

- Mobile Apps for Android phones
- Game development

Programming Language: Examples of Source Code

Source Code of a C Program

```
#include <stdio.h>
int main()
{
    printf("Hello, World!");
    return 0; //success
}
```

Source Code of a Java Program

```
public class HelloWorld {
    public static void main( String[] args ) {
        System.out.println("Hello, World!");
        System.exit( 0 ); //success
    }
}
```

Source Code of a C++ Program

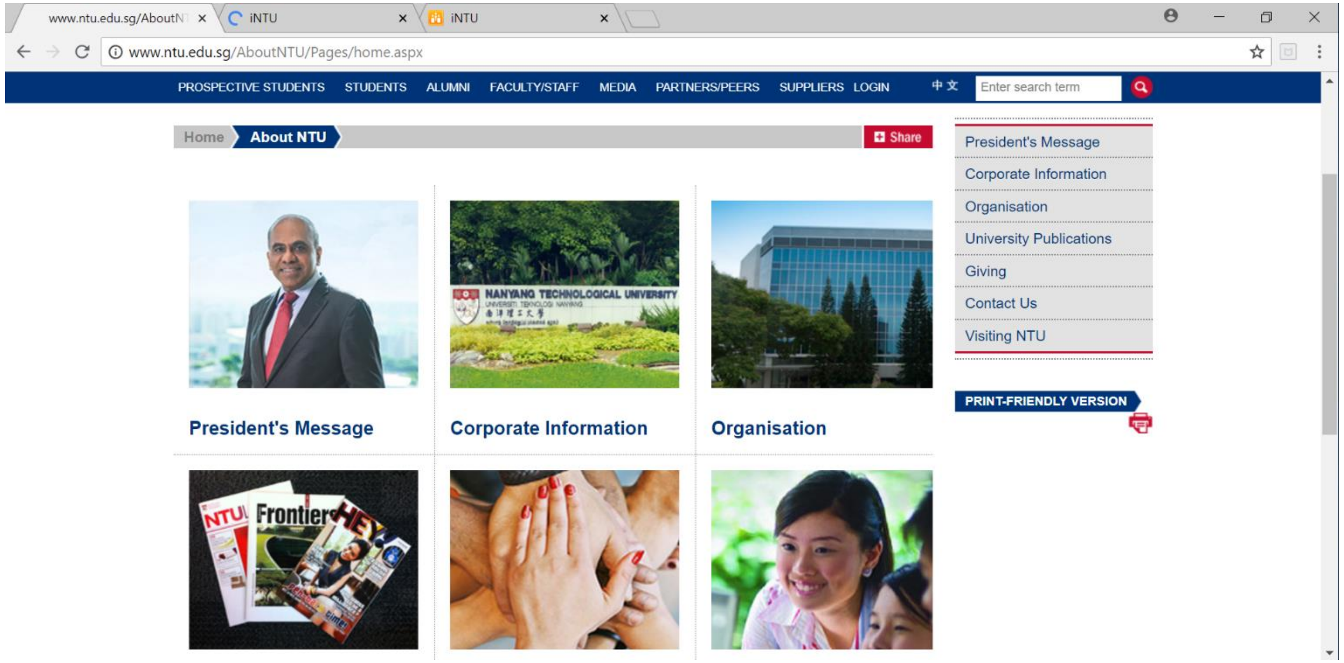
```
#include <iostream>
int main()
{
    std::count << "Hello, World!";
    return 0;
}
```

Source Code of a Python Program

```
print('Hello, World!')
```

Programming Language for Internet

Webpage: a text document transferred over the internet and be displayed in a web browser, such as Chrome, Safari, Firefox, and Internet Explorer (IE)



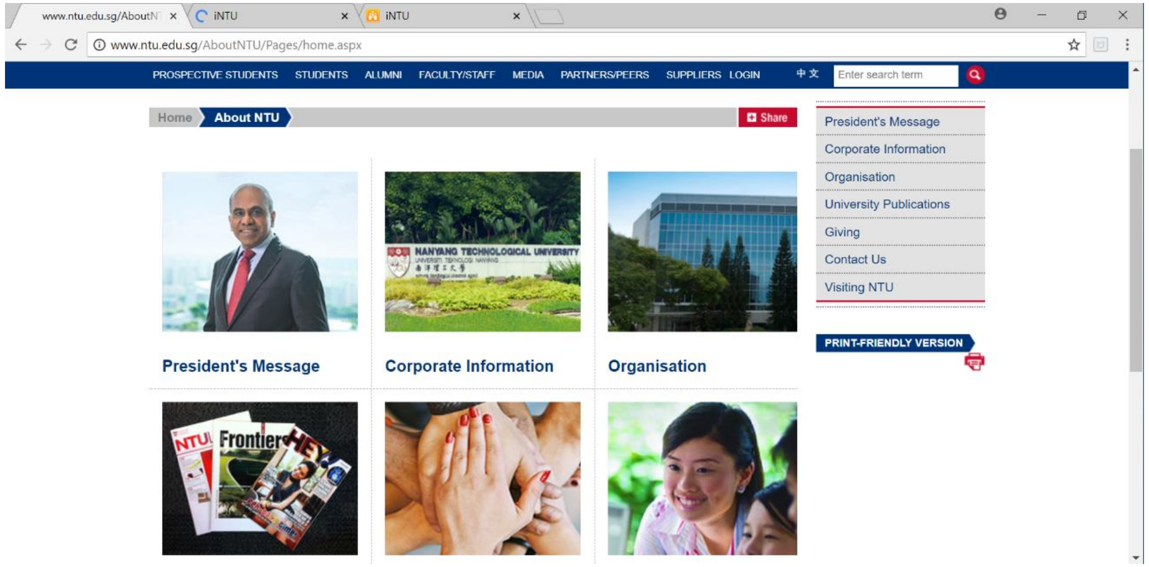
www.ntu.edu.sg/AboutNTU/Pages/home.aspx

A web browser reads the webpage and composes it into visible or audible page(s).

Programming Language for Internet (Cont’d)

The content of a webpage is commonly described by using the language **HTML** (Hypertext Markup Language)

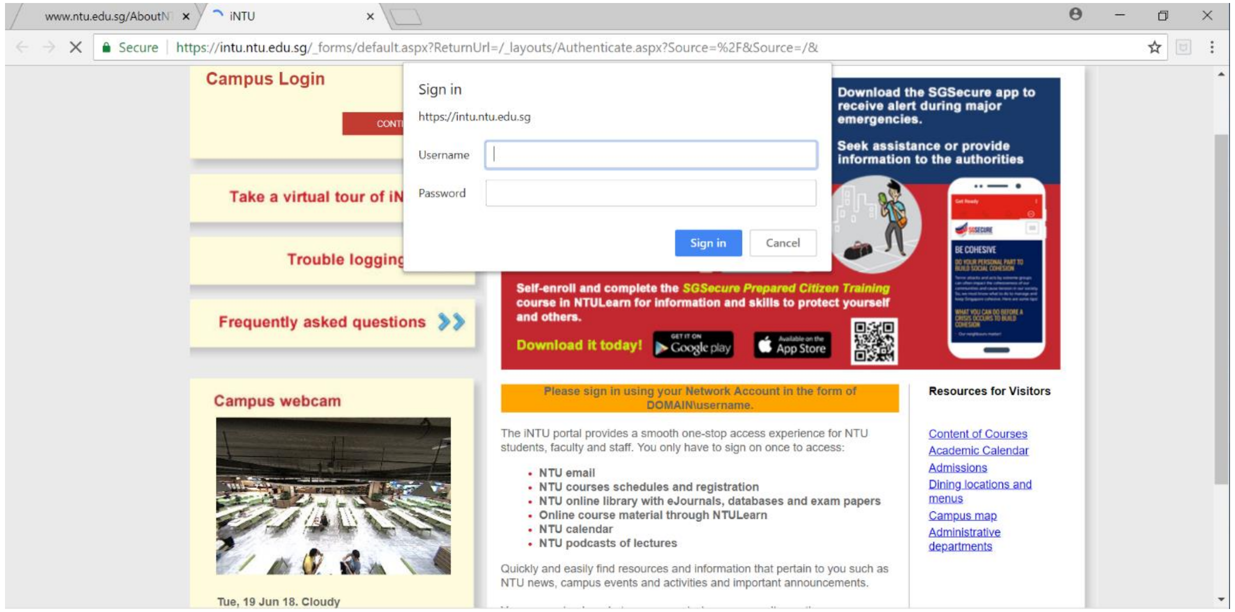
- a content-markup language uses symbols and phrases
- instructs the web browser to structure information for display or process


www.ntu.edu.sg/AboutNTU/Pages/home.aspx

Programming Language for Internet (Cont’d)

A webpage can be made to appear dynamic and interactive by adding scripts in language, such as **JavaScript**.

Example: to prompt for user’s name in the webpage



www.ntu.edu.sg/AboutN
iNTU
Secure
https://intu.ntu.edu.sg/_forms/default.aspx?ReturnUrl=/_layouts/Authenticate.aspx?Source=%2F&Source=/&

Sign in

https://intu.ntu.edu.sg

Username

Password

Sign in
Cancel
Campus Login
CONTI
Take a virtual tour of iN
Trouble logging
Frequently asked questions
Campus webcam

Tue, 19 Jun 18. Cloudy

Download the SGSecure app to receive alert during major emergencies.

Seek assistance or provide information to the authorities

Self-enroll and complete the *SGSecure Prepared Citizen Training* course in NTULearn for information and skills to protect yourself and others.

Download it today!
GET IT ON Google play
Available on the App Store
BE COHESIVE

Please sign in using your Network Account in the form of DOMAIN\username.

The iNTU portal provides a smooth one-stop access experience for NTU students, faculty and staff. You only have to sign on once to access:

- NTU email
- NTU courses schedules and registration
- NTU online library with eJournals, databases and exam papers
- Online course material through NTULearn
- NTU calendar
- NTU podcasts of lectures

Quickly and easily find resources and information that pertain to you such as NTU news, campus events and activities and important announcements.

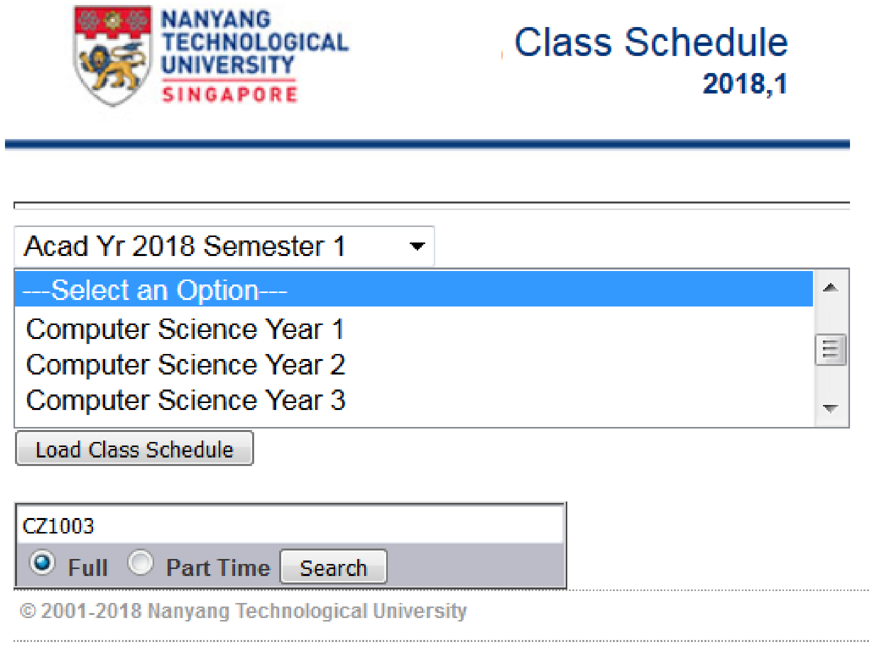
Resources for Visitors

Content of Courses
Academic Calendar
Admissions
Dining locations and menus
Campus map
Administrative departments

Programming Language for Internet (Cont'd)

PHP (PHP: Hypertext Pre-processor) is another script language

- dynamically generates html webpage before it is sent to the browser
- Example: contents that need to be composed based on user's request



Class Schedule
2018,1

Acad Yr 2018 Semester 1

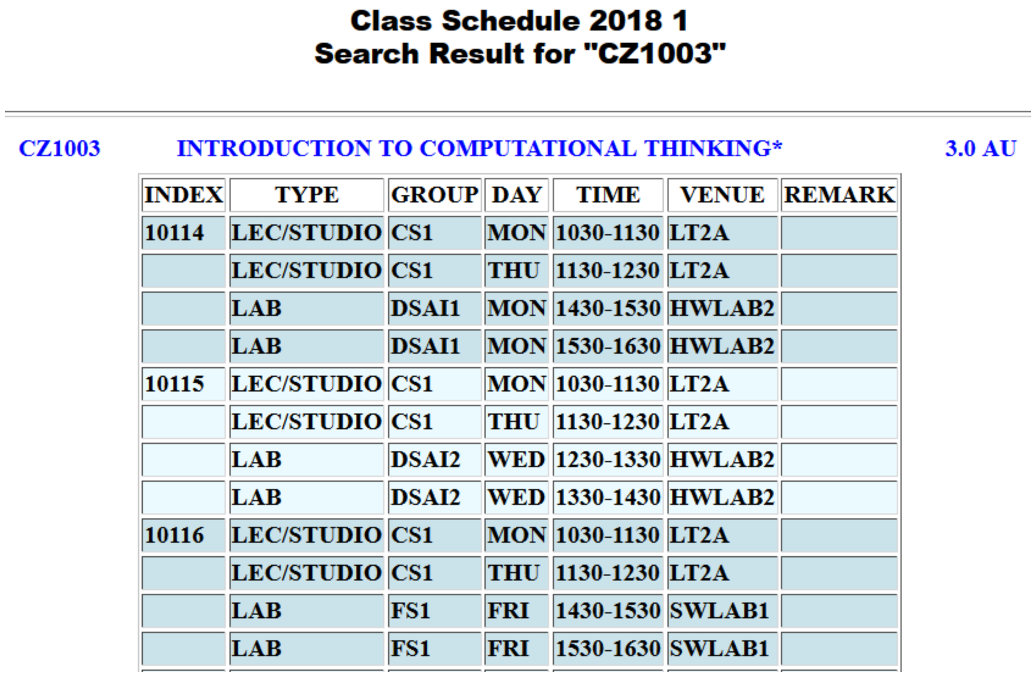
---Select an Option---
Computer Science Year 1
Computer Science Year 2
Computer Science Year 3

Load Class Schedule

CZ1003

● Full ○ Part Time Search

© 2001-2018 Nanyang Technological University



Class Schedule 2018 1
Search Result for "CZ1003"

CZ1003 INTRODUCTION TO COMPUTATIONAL THINKING* 3.0 AU

INDEX	TYPE	GROUP	DAY	TIME	VENUE	REMARK
10114	LEC/STUDIO	CS1	MON	1030-1130	LT2A	
	LEC/STUDIO	CS1	THU	1130-1230	LT2A	
	LAB	DSAI1	MON	1430-1530	HWLAB2	
	LAB	DSAI1	MON	1530-1630	HWLAB2	
10115	LEC/STUDIO	CS1	MON	1030-1130	LT2A	
	LEC/STUDIO	CS1	THU	1130-1230	LT2A	
	LAB	DSAI2	WED	1230-1330	HWLAB2	
	LAB	DSAI2	WED	1330-1430	HWLAB2	
10116	LEC/STUDIO	CS1	MON	1030-1130	LT2A	
	LEC/STUDIO	CS1	THU	1130-1230	LT2A	
	LAB	FS1	FRI	1430-1530	SWLAB1	
	LAB	FS1	FRI	1530-1630	SWLAB1	

Programming Language: Others

R language

- an interpreted language
- used for statistical computing, such as data analysis
- provides extensive graphic plotting functions
  - useful for displaying data trends

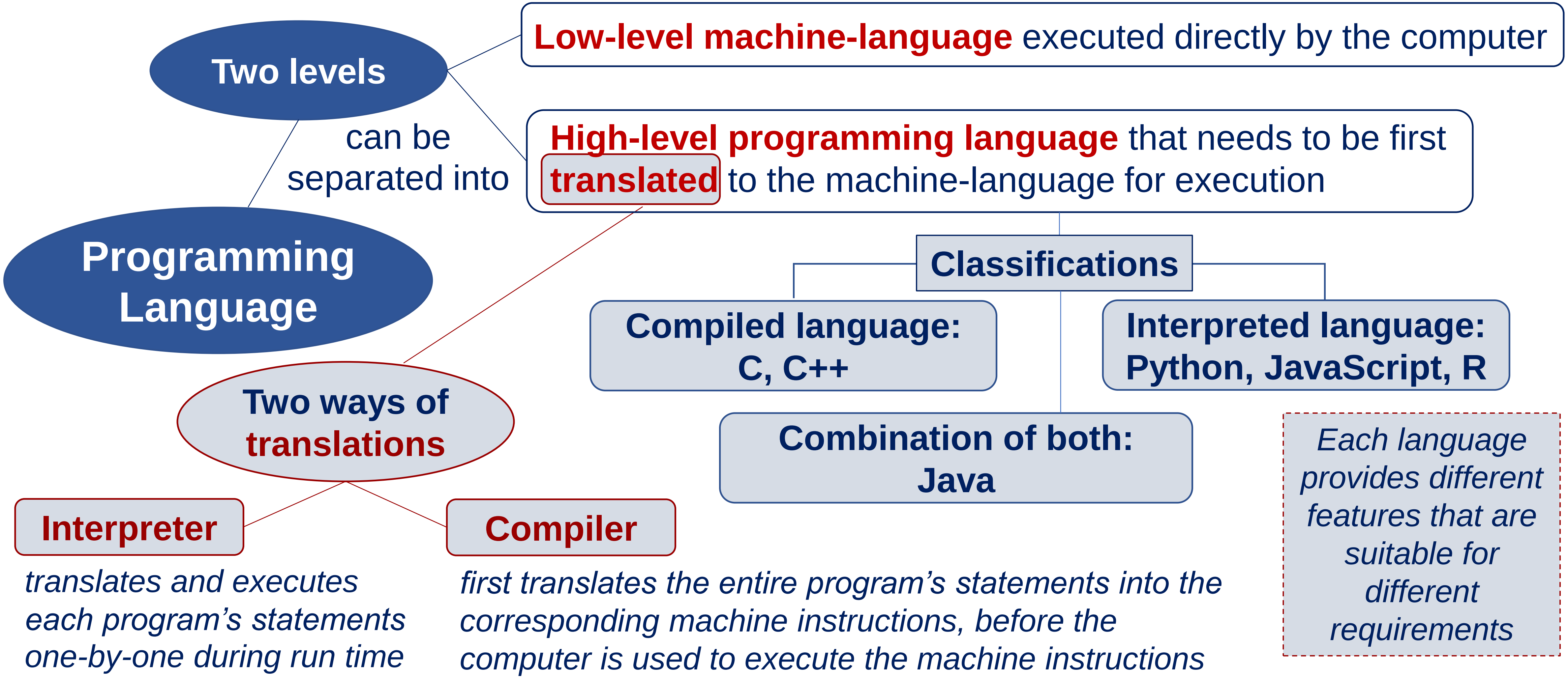
MATLAB language

- commonly used for developing science and engineering computation applications
- an interpreted language used primarily for numerical computing
- supports many functions and libraries, such as matrix manipulation, plotting, and visualization of data

Examples of types of computation:

- audio, video, and image related signal processing
- control systems

Summary



Programming Language

can be separated into

Two levels

Low-level machine-language executed directly by the computer

High-level programming language that needs to be first **translated** to the machine-language for execution

Classifications

Compiled language: C, C++

Interpreted language: Python, JavaScript, R

Combination of both: Java

Each language provides different features that are suitable for different requirements

Two ways of **translations**

Interpreter

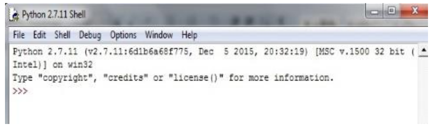
translates and executes each program's statements one-by-one during run time

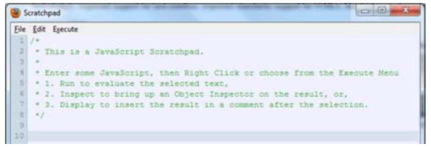
Compiler

first translates the entire program's statements into the corresponding machine instructions, before the computer is used to execute the machine instructions

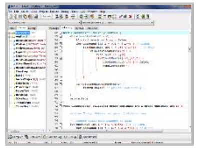
References for Images

No.	Slide No.	Image	Reference
1	3		Monitor screen [Online Image]. Retrieved June 20, 2018 from https://pixabay.com/en/monitor-screen-computer-speaker-2455524/.
2	6, 13, 15		Python logo [Online Image]. Retrieved June 20, 2018 from https://www.python.org/static/community_logos/python-logo-master-v3-TM.png.
3	6, 14, 15		Java log [Online Image]. Retrieved June 20, 2018 from http://www.oracle.com/us/technologies/java/java-licensing-logo-guidelines-1908204.pdf.

4	8	 Python 2.7.11 Shell
		File Edit Shell Debug Options Window Help
		`Python 2.7.11 (v2.7.11:6d1b6a68f775, Dec  5 2015, 20:32:19) [MSC v.1500 32 bit (`
		`Intel)] on win32`
		`Type "copyright", "credits" or "license()" for more information.`
		`>>>`
			By Annakoppad - Own work, CC BY-SA 4.0,retrieved June 20, 2018 from https://commons.wikimedia.org/w/index.php?curid=48721101.

5	8	
		`/*`
		` * This is a JavaScript Scratchpad.`
		` *`
		` * Enter some JavaScript, then Right Click or choose from the Execute Menu`
		` * 1. Run to evaluate the selected text,`
		` * 2. Inspect to bring up an Object Inspector on the result, or,`
		` * 3. Display to insert the result in a comment after the selection.`
		` */`
			By Trewyy - Own work, CC BY-SA 3.0,retrieved June 20, 2018 from https://commons.wikimedia.org/w/index.php?curid=18694958.

References for Images

No.	Slide No.	Image	Reference
6	9		Devcpp5110 [Online Image]. Retrieved June 20, 2018 from https://pixabay.com/en/amd-cpu-processor-microprocessor-1310766/.
7	10, 15		By Jeremy Kratz - https://github.com/isocpp/logos, Copyrighted free use, retrieved June 20, 2018 from https://commons.wikimedia.org/w/index.php?curid=62851110.
8	17		By W3C, CC BY 3.0, retrieved June 20, 2018 from https://commons.wikimedia.org/w/index.php?curid=12736763.
9	19		Php Logo [Online Image]. Retrieved June 20, 2018 from http://php.net/images/logos/new-php-logo.png.