



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)



دانشکده مهندسی کامپیوتر
و فناوری اطلاعات

برنامه نویسی پیشرفته

مقدمه و آشنایی با جاوا

نیم سال دوم ۹۹-۹۸

Who am I? 😊

- www.linkedin.com/in/ehsan-edalat
- <https://ceit.aut.ac.ir/formalsecurity/>
- Research Interests
 - Privacy
 - Language-based Security
 - Programming Languages
 - Runtime Verification of Software Systems
 - Formal Methods in Security and Software Engineering
 - Software Vulnerability Analysis and Detection



Technical Team

- Lecturers

- Dr. Kalbasi, Dr. Zeinali, and ME!

- Workshop Instructors

- Reshad, Tannaz, Hamideh, Sina, Roya, and Reza

- TAs

- MohammadHossein, Mohammad(3), Omid,
MohammadEbrahim, Erfan, Hadi, Mehdi,
MohammadReza, MohammadAli, Faezeh,
Mahshid, Pariya, Mana, AliReza, Rozhin, and
Mahvash

Contact Me

- Moodle (courses.aut.ac.ir)
 - Keep yourself updated!
- Office hours: (@Formal Security Lab. – 2nd Floor)
 - Wednesdays 16:30-18:00, Or by appointment
- The fastest, most reliable, most appropriate, and believe me (!) **THE BEST** way to contact me
[mailto: ehsan.e.71@gmail.com](mailto:ehsan.e.71@gmail.com)

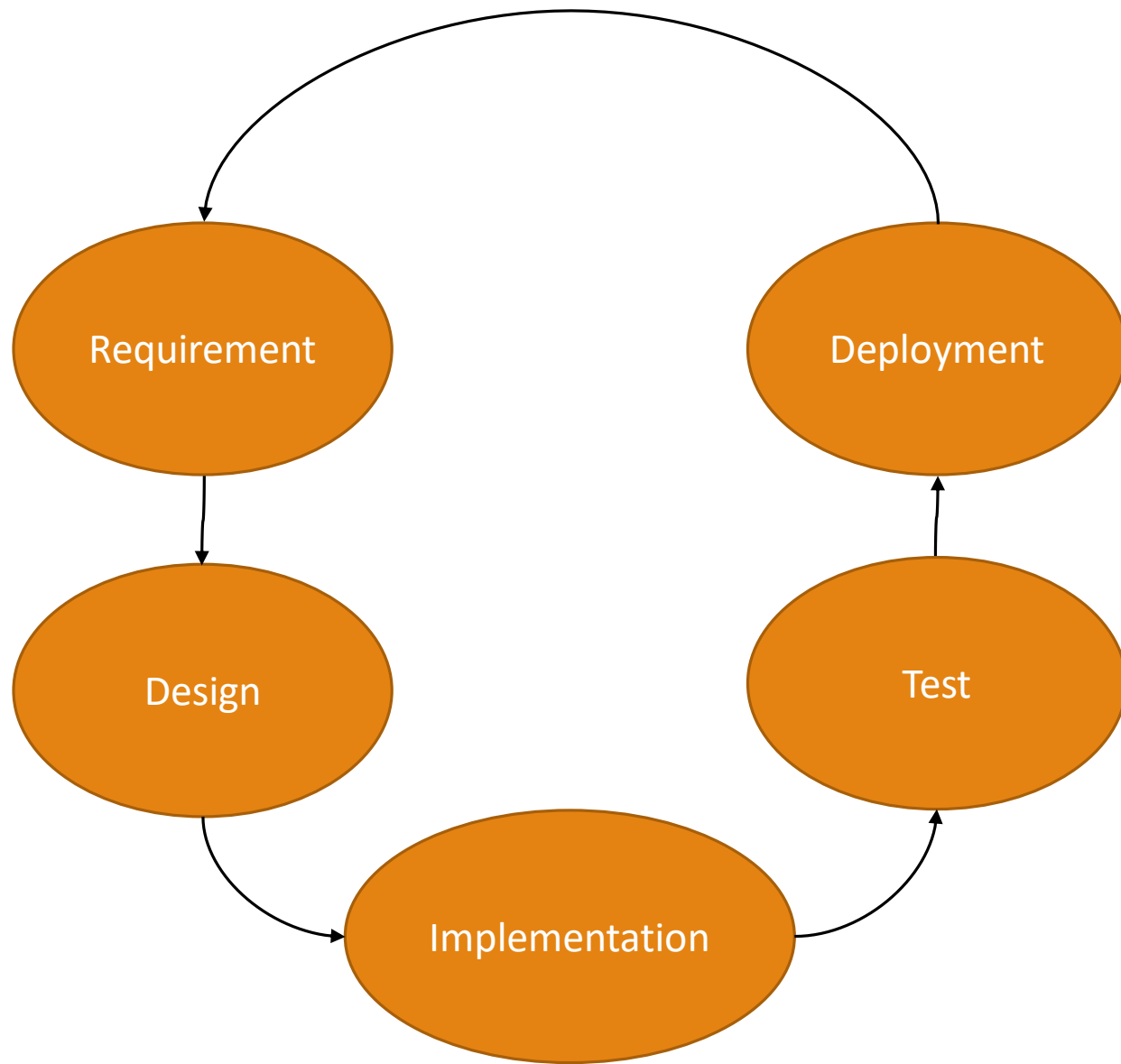
طرح درس و قوانین کلاس

□ فایل طرح درس!

برنامه‌نویسی پیشرفته – مباحث درس

- مهندسی نرم افزار
- مهندسی فرآیند تولید نرم افزار
- تولید نرم افزار با کیفیت

- نرم افزار با کیفیت
- قابلیت استفاده مجدد
- قابل نگهداری
- کارایی
- قابل راستی آزمایی
- ...



برنامه‌نویسی پیشرفته – مباحث درس

- ❑ آموزش برنامه‌نویسی شیء‌گرا (Object-Oriented)
- ❑ آموزش برنامه‌نویسی با یک زبان مدرن و توانمند
- ❑ مرور و آموزش بیشتر قواعد برنامه‌نویسی خوب

یادگیری برنامه نویسی

❑ برنامه نویسی یک هنر و در عین حال یک فن است

❑ نشستن سر کلاس و گوش کردن به صحبت های حتی بهترین اساتید برنامه نویسی راه آموختن برنامه نویسی نیست!

- ❑ Get taught a little,
- ❑ read other programmer's codes and books,
- ❑ and **start programming a lot!!**

Java

- ❑ Almost pure object-oriented language
- ❑ Portable & platform independent
- ❑ Well supported by Oracle corporation
- ❑ Broad range of applications:
 - Java ME, Java SE, Java EE, Web, Android, ...
- ❑ Highly popular



Java's Popularity

- ❑ TIOBE ranking 2020 Java = #1
 - <https://www.tiobe.com/tiobe-index/>

- ❑ RedMonk ranking 2019 Java = #2
 - <https://redmonk.com/sograzy/2019/07/18/language-rankings-6-19/>

- ❑ IEEE Spectrum ranking 2018 Java = #3
 - <https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2018>

- ❑ PYPL ranking 2020 Java = #2
 - <http://pypl.github.io/PYPL.html>

Java's Popularity



Jan 2020	Jan 2019	Change	Programming Language	Ratings	Change
1	1		Java	16.896%	-0.01%
2	2		C	15.773%	+2.44%
3	3		Python	9.704%	+1.41%
4	4		C++	5.574%	-2.58%
5	7	⬆	C#	5.349%	+2.07%
6	5	⬇	Visual Basic .NET	5.287%	-1.17%
7	6	⬇	JavaScript	2.451%	-0.85%
8	8		PHP	2.405%	-0.28%
9	15	⬆	Swift	1.795%	+0.61%
10	9	⬇	SQL	1.504%	-0.77%
11	18	⬆	Ruby	1.063%	-0.03%
12	17	⬆	Delphi/Object Pascal	0.997%	-0.10%
13	10	⬇	Objective-C	0.929%	-0.85%
14	16	⬆	Go	0.900%	-0.22%
15	14	⬇	Assembly language	0.877%	-0.32%

<https://www.tiobe.com/tiobe-index/>



Java's Popularity

Language Ranking: IEEE Spectrum

Rank	Language	Type	Score
1	Python	  	100.0
2	Java	  	96.3
3	C	  	94.4
4	C++	  	87.5
5	R		81.5
6	JavaScript		79.4
7	C#	   	74.5

<https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2019>

Java's Popularity

Worldwide, Feb 2020 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	29.88 %	+4.1 %
2		Java	19.05 %	-1.8 %
3		Javascript	8.17 %	+0.1 %
4		C#	7.3 %	-0.1 %
5		PHP	6.15 %	-1.0 %
6		C/C++	5.92 %	-0.2 %
7		R	3.74 %	-0.2 %
8		Objective-C	2.42 %	-0.6 %
9		Swift	2.28 %	-0.2 %
10	↑	TypeScript	1.84 %	+0.3 %
11	↓	Matlab	1.83 %	-0.1 %
12	↑↑↑↑↑	Kotlin	1.64 %	+0.5 %
13		VBA	1.31 %	-0.1 %

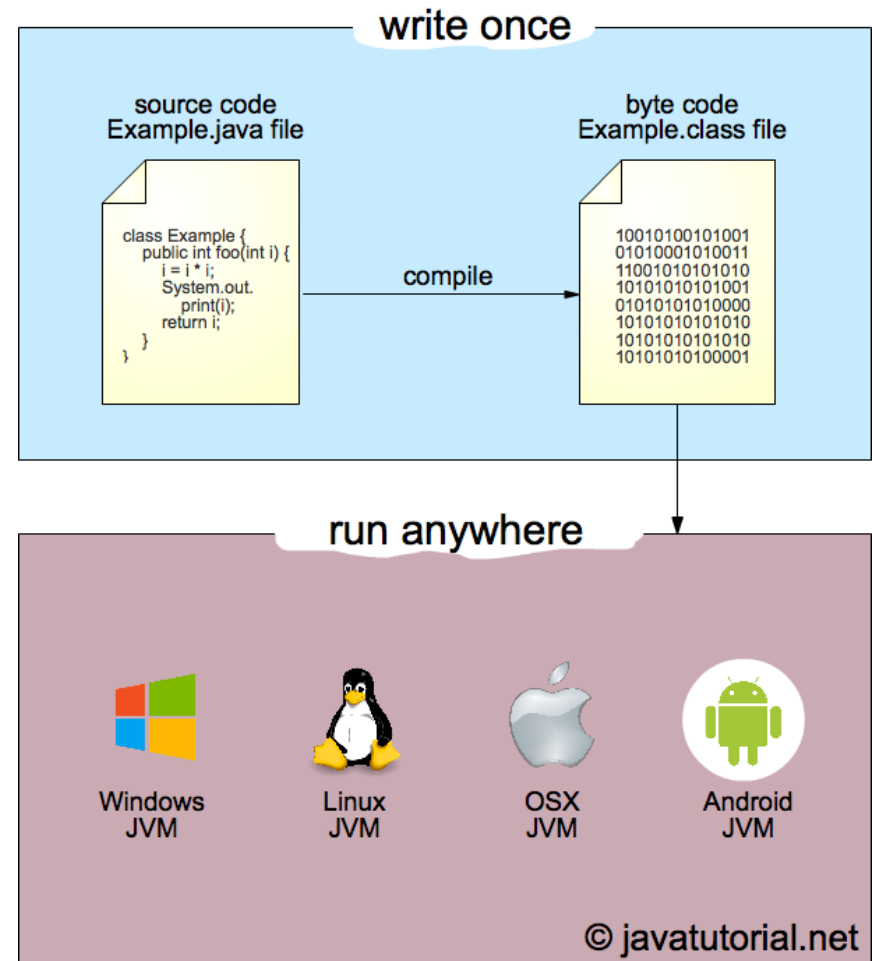
<http://pypl.github.io/PYPL.html>

Java

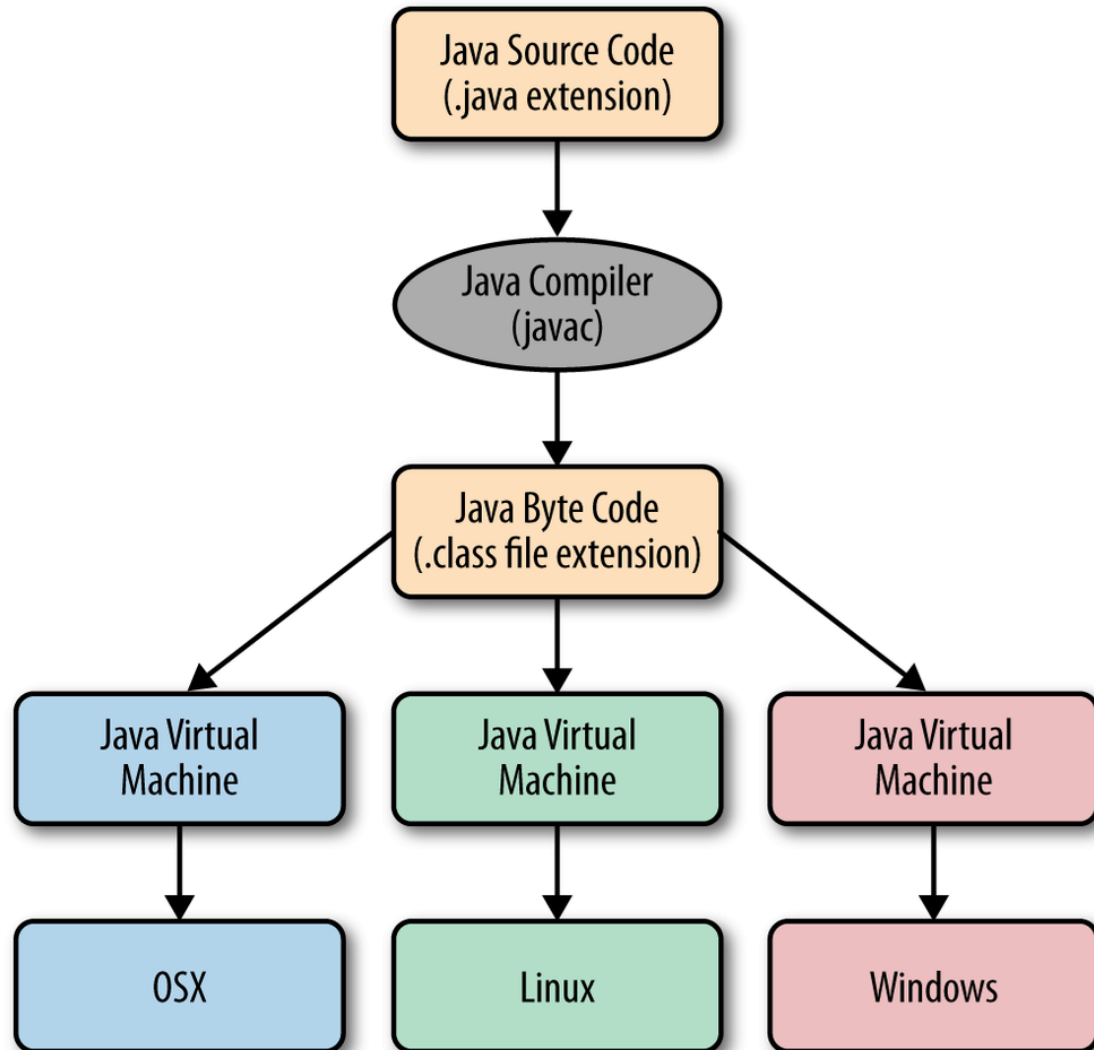
- ❑ Simple & Readable
 - Easy to learn for beginners
- ❑ Robust
 - well specified grammar
- ❑ High Performance
 - Advanced (Just-In-Time) JIT Compiler
- ❑ Up to date and Modern
 - Includes features required by modern software
- ❑ Platform independent and portable
 - "Write once, run everywhere"

Java – How it Works

- ❑ Java files
 - .java file extension
- ❑ Compiled to
 - Java bytecode
 - .class file extension
- ❑ Bytecode
 - Intermediate code
 - Runs on Java Virtual Machine (JVM)



<https://javatutorial.net/jvm-explained>



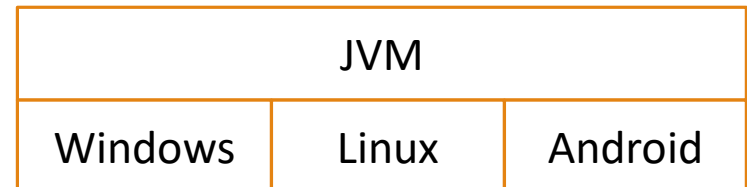
Java Bytecode and JVM

```
outer:
for (int i = 2; i < 1000; i++) {
  for (int j = 2; j < i; j++) {
    if (i % j == 0)
      continue outer;
  }
  System.out.println (i);
}
```

Compile

```
0: iconst_2
1: istore_1
2: iload_1
3: sipush 1000
6: if_icmpge 44
9: iconst_2
10: istore_2
11: iload_2
12: iload_1
13: if_icmpge 31
16: iload_1
17: iload_2
18: irem      # remainder
19: ifne 25
22: goto 38
25: iinc 2, 1
28: goto 11
31: getstatic  #84; //Field java/lang/System.out:Ljava/io/PrintStream;
34: iload_1
35: invokevirtual #85; //Method java/io/PrintStream.println:(I)V
38: iinc 1, 1
41: goto 2
44: return
```

runs



More about Java

- ❑ https://www.java.com/en/download/faq/whatis_java.xml
- ❑ <http://www.bestprogramminglanguagefor.me/why-learn-java>
- ❑ <https://www.pearsonfrank.com/blog/why-is-java-so-popular-developers/>

روش بالا به پایین در حل مسئله

- ابتدا نگاه کلی به مسئله
- اضافه کردن جزئیات به صورت مرحله ای
 - شکستن مسئله به مسایل ریزتر
 - درک بهتر مسئله
 - آسانتر شدن ارائه راه حل
- ارائه راه حل برای مسایل ریزتر
- جمع بندی و ارائه راه حل کلی

نمونه مسئله ۱

□ نمایش الگوی زیر بر مبنای ورودی کاربر

برای ورودی ۵

*

* *

* * *

* * * *

* * * * *

نگاه بالا به پایین – نمونه مسئله ۱

*

* *

* * *

* * * *

* * * * *

□ بررسی الگو

■ تعداد سطر

○ برابر با ورودی از کاربر

• ورودی از کاربر

■ تعداد ستاره‌ها در هر سطر

○ برابر با شماره سطر

نگاه بالا به پایین – نمونه مسئله ۱

- * تولید و نمایش الگو
- * * دریافت تعداد سطور از کاربر
- * * * تعداد سطور < نیاز به متغیر < متغیر عدد صحیح
- * * * * اعلام به کاربر
- * * * * * دریافت از کاربر

حلقه ← نمایش سطور به تعداد ورودی از کاربر

○ نمایش هر سطر

حلقه ← نمایش ستاره به تعداد شماره سطر

- شماره سطر < نیاز به متغیر < متغیر عدد صحیح
- تعداد ستاره < نیاز به متغیر < متغیر عدد صحیح


```
#include <stdio.h>
int main()
{
    int numRows;
    printf("Enter number of rows: ");
    scanf("%d", &numRows);
    printf("You entered: %d.\nHere is the pattern with %d rows:\n",
numRows, numRows);
    for(int row = 1; row <= numRows; row++)
    {
        for(int starCount = 1; starCount <= row; starCount++)
        {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

Enter number of rows: 10

You entered: 10.

Here is the pattern with 10 rows:

*

**

Sample Code

```
#include <stdio.h>
```

```
int main()
```

```
{
```


```
    // printf() displays the string inside quotation
```

```
    printf("Hello, World!");
```

```
    return 0;
```

```
}
```

شروع برنامه



Java Hello, Students!

```
public class WelcomeToCE105 {  
    public static void main(String[] args) {  
        System.out.println("Hello, Students!");  
    }  
}
```

javac WelcomeToCE105.java  WelcomeToCE105.class

java WelcomeToCE105  Hello, Students!

سوال؟
