

# ITI 1120

## Introduction to Computing I

### Laboratory Manual

*Gilbert Arbez, Diana Inkpen,  
Hussein Al Osman, Shervin Shirmohammadi*



# Table of Contents

ITI 1120 .....	1
Introduction to Computing I.....	1
Laboratory Manual.....	1
Lab Facilities.....	3
Course accounts .....	3
Generating a course account .....	4
Logging In.....	5
Logging Out .....	6
The EECS Environment.....	6
Printing from EECS labs.....	7
Programming Standards.....	9
Input / Output.....	9
Variables .....	9
Program Structure .....	10
Restrictions .....	11
The Java SDK (Software Development Kit) .....	12
Java JDK documentation .....	12
The Dr. Java Environment .....	13
Installing Microsoft Visio from Microsoft DreamSpark .....	18

## **Lab Facilities**

Labs for ITI 1120 will be held in rooms STE 0130, STE 0131, STE 2052 or STE 2060. The STE 0110 general lab is available for use at other times during its open hours. All four of these rooms have similar computers, and you will be able to access your files from any of these rooms.

The computers are various configurations of IBM personal computers. The operating system is Windows XP on all computers. The operating system is being migrated to Windows 7. Some of the computers in the labs have already been updated.

Please see <http://www.site.uottawa.ca/local/labinfo> for complete information about the lab facilities.

It is also possible to work on your own computer, if you install the Java Development Kit (JDK). See the instructions at the end of the lab manual for details on installing the Java SDK and Dr Java

## **Course accounts**

Using any of the EECS labs requires an EECS computer account. This is different from your general university account. Your university account user identification (user name or “userid”) will be similar to:

awill1039                      (a contraction of first initial, family name, and three digits)

The EECS account will have the following form of userid:

u0618234                      (‘u’ followed by the student number)

The general university account will be used for official e-mail communications – even for the ITI 1120 course – and e-mail should be checked there regularly. The EECS account is only for access to the EECS workstations and network.

If your major is computer science or software engineering, your EECS account will be active for the entire length of your studies. If your major is not one of the above, your EECS account will be activated only during terms in which you take a course with course codes starting with ITI, CSI, or SEG.

## Generating a course account

Generating an account is an operation that needs to be done only once at the start of the term. When you generate an account, space on a central disk server will be reserved for you. You will be able to access files stored on this server from any EECS lab computer.

To start to generate an account, you need to access the login screen. You may need to move the mouse on a computer if the screen is blank or a screen saver is running. Otherwise, you should see the “Welcome to Windows XP” screen.

Press all three of the CTRL, ALT, and DEL keys simultaneously to access the login screen. [This is referred to as “control-alt-delete”.] The EECS login window should appear.

Normally, you would login with your EECS userid and password. To generate your account, use “autopass” as the userid, and “autopass” as the password. As you type the characters of your password, they will appear as • characters so that your password cannot be read from the screen. Click the “Login” button to proceed.

The Account Generation system will ask for your student number, and then prompt you twice for a lab password of your choice. It will then ask you to authenticate yourself using a PIN (personal identification number). This is your InfoWeb student services password, which you should have received as part of your registration package. No other password will work. Your Infoweb Student Services password (or PIN) will remain unchanged; it has nothing to do with your EECS account password. The lab password that you have supplied will be the one used for your EECS lab account.

If you do not know what your InfoWeb password is, click on the 'PIN?' button. This will take you to a page that will allow you to obtain your InfoWeb password.

If all went well, the account generation program will indicate which labs your account will enable you to use.

### Problems?

- If your PIN (Infoweb password) was not correct, you will need to find out what is your PIN (it was sent to you in your registration package). You should contact InfoService, at <http://www.uottawa.ca/student/infoservice/> or at 562-5700.
- If you are not registered to a course that grants access to an account, the system will indicate this. Note that the registration information that is used to generate accounts may be few days old. If you have registered just recently, come back in a day or two.
- If you forget your password, simply regenerate your account using the same procedure.
- If the Account Generation system fails because of system errors (this is unfortunately unavoidable), report this to the consultant on duty, then try again later.

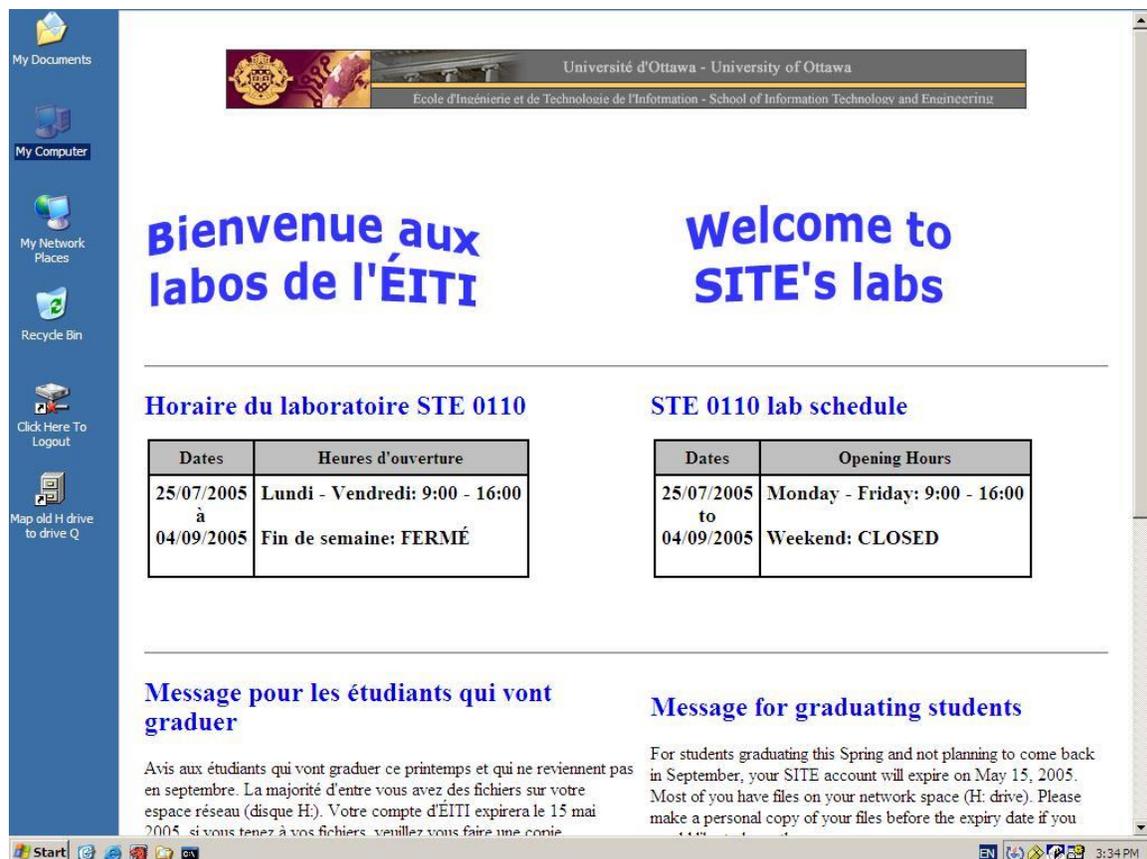
## Logging In

An unoccupied computer should be showing the “Welcome to Windows XP” screen. If the computer has been idle for a while, the screen may be blank and you should wiggle the mouse or press any key to turn on the screen.

Press all three of the CTRL, ALT, and DEL keys simultaneously to access the login screen. [This is referred to as “control-alt-delete”.] The EECS login window should appear.

Enter your userid in the “user name” area. Type your password in the password box. As you type the characters of your password, they will appear as • characters so that your password cannot be read from the screen. Click on the “Login” button or hit enter to log in.

The login process may take a few seconds. When the login process has finished, the screen should be similar to what is shown in Figure 1, although the information shown is updated regularly with the latest news.



Universit  d'Ottawa - University of Ottawa  
Ecole d'Ingenierie et de Technologie de l'Information - School of Information Technology and Engineering

**Bienvenue aux labos de l' ITI**

**Welcome to SITE's labs**

**Horaire du laboratoire STE 0110**

Dates	Heures d'ouverture
25/07/2005	Lundi - Vendredi: 9:00 - 16:00
�	
04/09/2005	Fin de semaine: FERM�

**STE 0110 lab schedule**

Dates	Opening Hours
25/07/2005	Monday - Friday: 9:00 - 16:00
to	
04/09/2005	Weekend: CLOSED

**Message pour les  tudiants qui vont graduer**

Avis aux  tudiants qui vont graduer ce printemps et qui ne reviennent pas en septembre. La majorit  d'entre vous avez des fichiers sur votre espace r seau (disque H:). Votre compte d' ITI expirera le 15 mai 2005. si vous tenez   vos fichiers, veuillez vous faire une copie

**Message for graduating students**

For students graduating this Spring and not planning to come back in September, your SITE account will expire on May 15, 2005. Most of you have files on your network space (H: drive). Please make a personal copy of your files before the expiry date if you

Start 3:34 PM

Figure 1: Screen after a successful login

You can now use the computer, and you will have access to your files stored on the central file server.

## Logging Out

When you are finished your session, it is critical that you log out, to protect your files from the next user. Double-click on the following icon to log out:

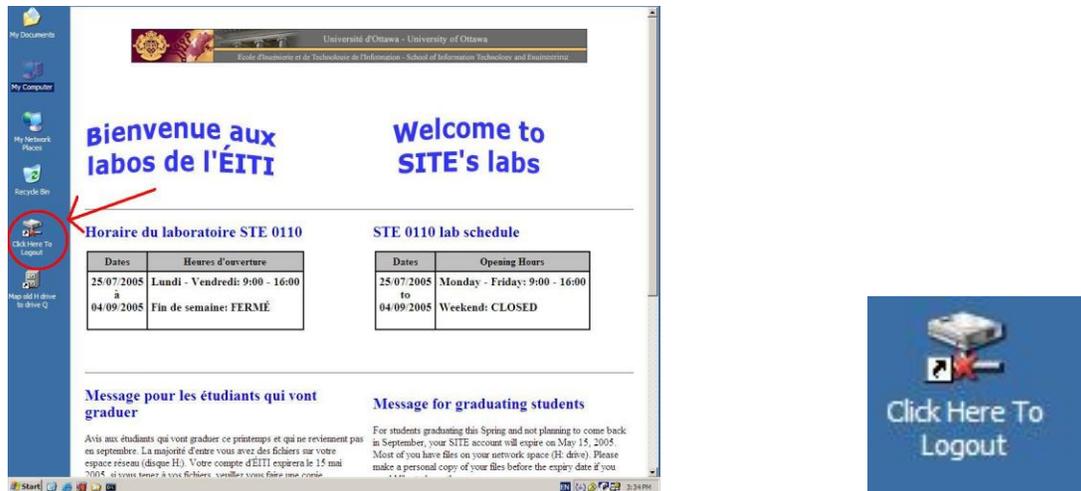


Figure 2: The logout icon

Do NOT turn the power off or shut down the machines.

## The EECS Environment

Here are some important features of the EECS lab environment that may be different from your own computer.

On a Windows system, there are several different methods of storing files on your computer. These are represented by “disk drive” letters. While these can vary among different computers, you may be familiar with “C” being the internal disk drive, “A” being a diskette drive, and “D” for a CD or DVD drive.

If you store your files in the “My Documents” folder, these files are normally stored on the C drive. The C drive usually also includes all the software installed on a computer.

There are some differences in the EECS lab configuration for several reasons:

- You may be using a different computer during your next visit to the lab, and you will need access to your files.
- We want to be sure that you have access to only your own files during your session, and not those of students who used the computer previously.
- In case an important system file is deleted accidentally, we want to be able to restore the computer to its original state.

- We want to protect the EECS lab computers against malicious software such as viruses, worms, spyware, and ad-ware.

Therefore, keep the following in mind as it is very important:

**When you log out from the computer, the C drive is restored to a standard configuration. This means that any files that you stored on the C drive will be erased at the end of a session. This includes files stored in the “My Documents” folder. These files cannot be recovered once erased.**

To provide permanent secure file storage, and to provide access to your files from any EECS lab computer, your files on the central file server will be made available on drive H when you log in. Here are the notable drive letters you will see on a EECS lab computer.

- A: The diskette drive. This can be used to transfer small files to other systems with similar drives. Although EECS lab computers have diskette drives, most newer computers do not include a diskette drive.
- C: The local internal disk drive of the computer. The “My Documents” folder is on this drive. The contents of the C drive are not available from other computers.
- E: The compact disc (CD) drive. The CD drives in the EECS labs are read-only; you will not be able to burn a CD from these computers, nor can you read from a DVD.
- F: If you plug in a flash memory stick to the universal serial bus (USB) port, it will appear as drive F.
- H: File space reserved for your account on the central file server. It will be available from any EECS lab computer, and can also be accessed from the Internet. This is where you should store files that you want to keep permanently.

You can store files temporarily for a session on the C drive, but any files that you want to keep permanently should be copied to the H drive (or put on a diskette or flash memory stick) before you log out.

There are other drives that store various software applications that are available on the computers.

## **Printing from EECS labs**

EECS provides line printers and laser printers for EECS students. Line printers are free for both graduate and undergraduate students. All Students have access to Relmon printers and must purchase a copy card to print documents using the Relmon printers.

Relmon printers are maintained by Computing & Communications Services (CCS).

Information about the names, locations and usage of EECS's printers are summarized in the following table:

Printer name	Location	Description
uglp1	STE 0110	High speed line printer
uglp2	STE 0110	High speed line printer
relmon	CBY Hall, 1st floor. STE 0110 and STE level 2.	Pay-per-use laser printers. Requires a photocopy card.

Note that the high speed line printers are **not** capable of handling graphics or advanced fonts. If accented letters are sent, they will be printed without the accent. Use the line printers to print programs and rough copies of documentation.

When you print a file using the Relmon system, your file is sent to a central print server. The print server holds your files until you are ready to print (and pay) for them. You can use any of the Self-Service printers located around campus to print. The printers are always available, the only limit being the hours in which the building is unlocked. At each printer there is a terminal to display and select the file(s) you want to print, and a photocopy card reader to accept payment.

You pay for printing in the same manner as when you use photocopiers on campus. There are copy card dispensers at various locations on campus from which you can purchase a copy card and load it with a monetary value. The card is then debited as you use the self-serve photocopiers or printers

Additional information is available from the following web sites:

- EECS printing documentation:

<http://www.eecs.uottawa.ca/local/labinfo/printing.shtml>

- Guide to self-service printers on campus, and their locations.

<http://www.uottawa.ca/services/ccs/docs/cyberleap/relmon/relmone.html>

## ***Programming Standards***

In industry, coding conventions help to reduce errors during program maintenance; in most cases, it will be someone else who has to understand the program well enough to change it without causing new problems. Many companies also have mandatory code inspections for quality control, where you are not allowed to contribute program code to a software product until it has been read out loud by someone else in a meeting attended by several software developers. Using standard coding conventions helps make your programs easier to understand by others (especially teaching assistants!).

For ITI 1120, approximately 10% of the grade for each assignment will be for following the course programming standards, and marks will be deducted for violations.

It is recommended to follow the general form of the Java code conventions at:

<http://java.sun.com/docs/codeconv/html/CodeConvTOC.doc.html>

The above web site is very detailed, and includes many parts of Java that are beyond the scope of ITI 1120.

In addition, for ITI 1120 the following standards **MUST** be respected:

### **Input / Output**

- Output should be nicely spaced and easy to understand. In particular, it should be easy to understand the output **WITHOUT** looking at the program listing. This means that for every value in your output you must print a short message explaining the meaning of the value.
- Before your program requests input values from the user, it must print a prompt describing what the user is required to enter.

### **Variables**

- Identifiers should be as meaningful as possible. Follow the standard upper-case/lower-case conventions for Java programs.
  - Method names and variables usually start with a lower case letter and then any additional words that are part of the name are capitalized. There are no spaces or underscores between words. The first word of a method name is often a verb.
    - Examples: `calculateStudentGrade( ), aNumber`
  - Class names should start with an upper case letter. Class names are generally a noun.
    - Examples: `Student, Course`

- Only declare variables that are actually used.
- Do not use the same variable for different purposes.
- Every variable should have a comment describing its purpose. Be brief. This is called a data dictionary - it should be placed at the beginning of every method, either before or beside the declarations.
- Do not use "global" variables. All values needed by a method must be passed into it as parameters and all values it produces or modifies must be returned as results or passed through parameters.

## Program Structure

- Good structure is very important. Decompose your algorithms/methods into meaningful sub-algorithms/methods whenever this improves clarity.
- Each method (including "main") should be preceded by comments describing its purpose and the algorithm it uses (if the algorithm is very simple, it need not be described). These descriptions should be brief.
- Your program must be indented in a systematic way that reflects the nesting of its statements.
- The following are some exceptions to the coding conventions at Sun's web site:
  - Branch statements must always follow the following form – namely, the braces must always be present and formatted as shown even when the Java language specification allows them to be omitted. The else clause is optional and may be omitted. Multiple if {} else if {} else {} statements should not be used.

```

if ( condition )
{
    // the true part
}
else
{
    // the false part
}

```

- If the else part requires no action and is included, it should be as follows:

```

{
    ; // do nothing
}

```

- Loops must follow one of the two forms – namely the braces must always be present and formatted as shown even when the Java language specification allows them to be omitted.

```
while ( condition )
{
    // the loop body
}
```

(or)

```
for ( initialization; condition; increment )
{
    // the loop body
}
```

- The following should not be used:

do

switch

break

+=, -=, \*=, /=

condition ? trueClause : falseClause

- Increment and decrement operators can only be used in the following forms

```
aVariable++; // no other code on this line
```

```
aVariable--; // no other code on this line
```

```
for ( initialization; condition; loopIndex++ )
```

## Restrictions

- Everything you submit (files and programs) must compile and run using the Java Development Kit (JDK) version on PCs in the EECS labs. “It works on my own computer,” is not a valid excuse if there are compatibility problems with your program.
- You may only use features and library methods of Java that have been presented in class or labs at the time an assignment is due. For example, you may not use Java's arrays before they are covered.

## **The Java SDK (Software Development Kit)**

The Java system used in the labs for ITI 1220 is the Java 2 Platform Standard Edition (J2SE). To create programs, you will need to get the Java Development Kit (JDK). The documentation is available on-line, but if you want to view the documentation off-line, it can also be downloaded. There is no charge for the use of this software or documentation.

If you work on your own computer, be sure that your programs will run on the machines in STE 0110 / 0130 / 0131 / 2052 / 2060. The teaching assistants will be running your assignments in one of those labs, and you are responsible to ensure that your program works in that environment.

You should be aware that there is some confusion with the version numbers and terminology of Java installations. When going from the previous Java version 1.4.2 to the “next” version, the new version has been identified as both 5.0 and 1.5.0. The current version of Java is 7.0 (also 1.7.0) This page will use 7.0 as the version number. Also, the terms SDK (Software Development Kit) and JDK (Java Development Kit) have been used interchangeably; at present, the current preferred term is JDK.

The most recent JDK version available on line is 7.0 update 2. The JDK download file is about 85 Megabytes in size; a high-speed connection is highly recommended to download the file.

- [Installation instructions for a Windows system.](#) (PLEASE READ)
- [Go to the Java Software Development Kit download page.](#)
  - On that page, scroll to **Java SE Development Kit (JDK) 7 Update 2**, and click on the download button to the right.

The download file is a self-installing executable (.exe) file. If your system will automatically execute such files, most of the work will happen automatically. If your system does not automatically execute downloaded files, you will have to follow the complete procedure described in the installation instructions. **In either case, read the instructions and follow step 5 carefully so that your system is set up to compile programs easily.** If you can type `javac` or `java` at a command line prompt and a list of options is given, you have installed everything correctly. If you get a message similar to “javac is not recognized as a command (etc.)”, then the installation is not complete, and you should check if step 5 of the installation instructions was done correctly.

### **Java JDK documentation**

There are two types of documentation: the Application Programmer’s Interface (API), and technical documentation about the JDK. The Application Programmer’s Interface is the documentation you will most likely use: it has descriptions of all the pre-defined classes and methods that come with Java. Both types of documentation are available on line. **Unless** you expect to be working off-line frequently (for example, on a portable computer), **downloading the documentation is not necessary.**

- [View the Application Programmer’s Interface \(API\) documentation for version 7.0.](#)
- [View the complete JDK documentation.](#)

To **download** both sets of documentation, click [here](#) to go to the Java Development Kit download page. On that page, scroll to **J2SE 7.0**

**Documentation** (under Additional Resources). Click on “download”. The

download file is about 60 Megabytes in size, and you will need an additional 165 Megabytes of disk space. Click [here](#) for complete download and installation instructions.

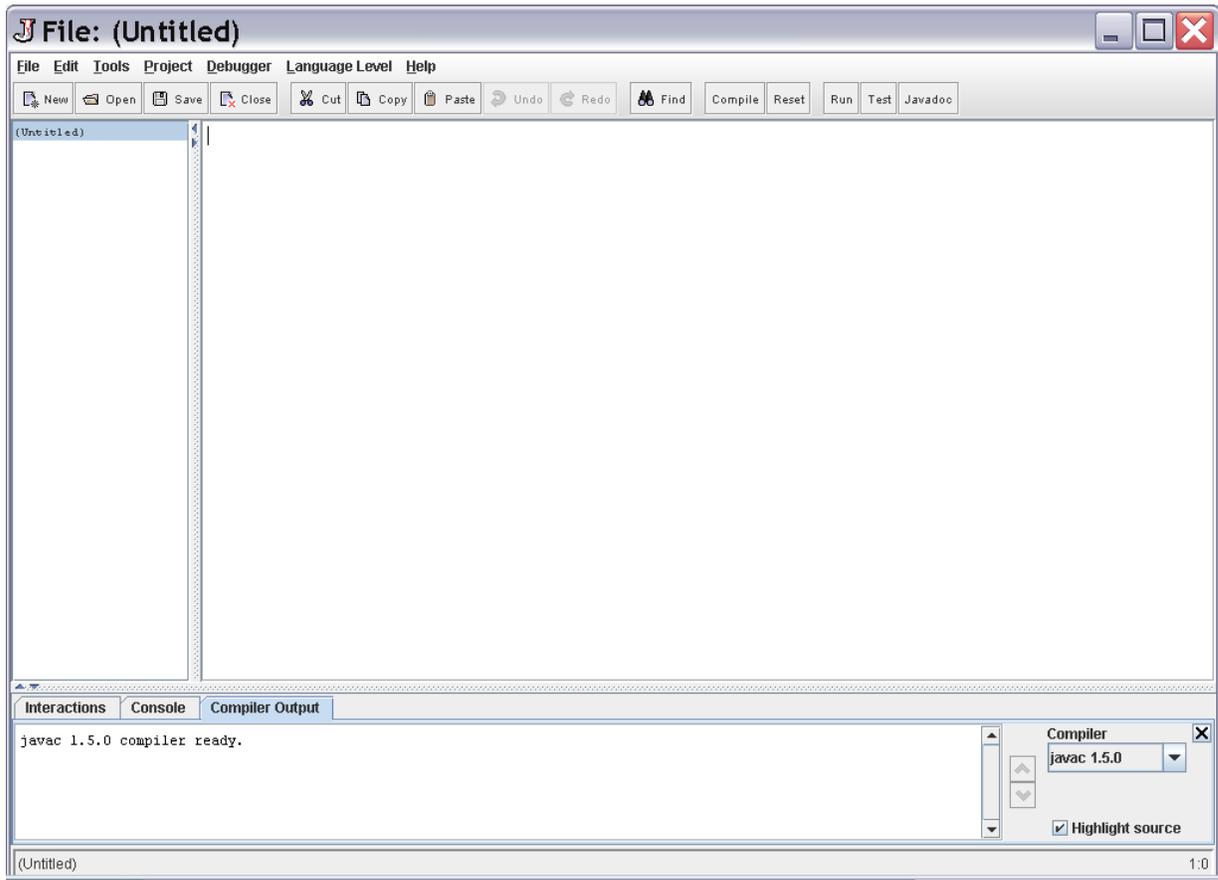
## ***The Dr. Java Environment***

The Dr. Java environment is an “integrated development environment” (IDE) that will be used for the ITI 1120 course. Dr. Java is available in the EECS labs, and you can also download your own copy for free from <http://www.drjava.org>. Dr. Java has been developed at Rice University to help students who are new to programming.

[Note that you can use any method that you wish to enter and compile programs. However, the teaching assistants will be able to give you help on using Dr. Java. If you use another IDE, you are on your own to learn how it works.]

If you start up Dr. Java, you will see a screen similar to the figure 5 bellow for the latest stable version (that is probably the one installed in the lab).

You may not see the line numbers; if you want to turn them on, click on the Edit menu and select “Preferences”. In the categories area, click on “Display Options” and then check the “Show all line numbers” box.



**Figure 5: Dr. Java window (latest stable version)**

The main screen area is partitioned into three parts. The upper right area is an editing area where you type in Java code. Dr. Java will help with Java code entry in the following ways:

- Java keywords, comments, and strings will be highlighted in various colours.
- Code will be indented properly.
- If there are problems compiling, the line with the problem will be highlighted.

The area on the left is a list of files that you are currently editing. In figure 5, (Untitled) appears in this area because there is no file associated with the empty editing window. You can edit several files at once, and click on a file name in the list to move between files.

The area at the bottom has three views indicated by the “tabs”: Interactions, Console, and Compiler Output. Currently, the “Interactions” view is selected, and you can move from one view to the other by clicking on a tab. Sometimes, Dr. Java will change the

view automatically when it is appropriate. When you compile a program, the output from the compiler will appear in the “Compiler Output” view. The “Console” is the area for text-based input and output for a Java program, and this is where you will see the results of running your program.

It is a tradition among programmers that when learning a new language that the first program you write will print “Hello, world!” In Java, the following program will print this message to the console:

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

Be very careful to type this exactly as shown, including the semicolon on the line with the message to print. By entering the above, you have created a “class” called `Hello`. Java programs always consist of one or more classes.

You are now ready to save the file. Note that a \* has appeared after (untitled) to indicate that there are changes that have not been saved. Click on the “Save” button to save your file. You can choose any folder you wish. Notice that in the “File name:” and “Files of type:” areas, Dr. Java has filled in “Hello” – the name of the class – and Java source files. This means that the proposed name of your file is `Hello.java`; **it is important for Java that the name of the file ends in .java to indicate Java source code, and it is also very important that the name of the file matches the name of the class inside the file.** Therefore, you should accept the proposed names by clicking Save.

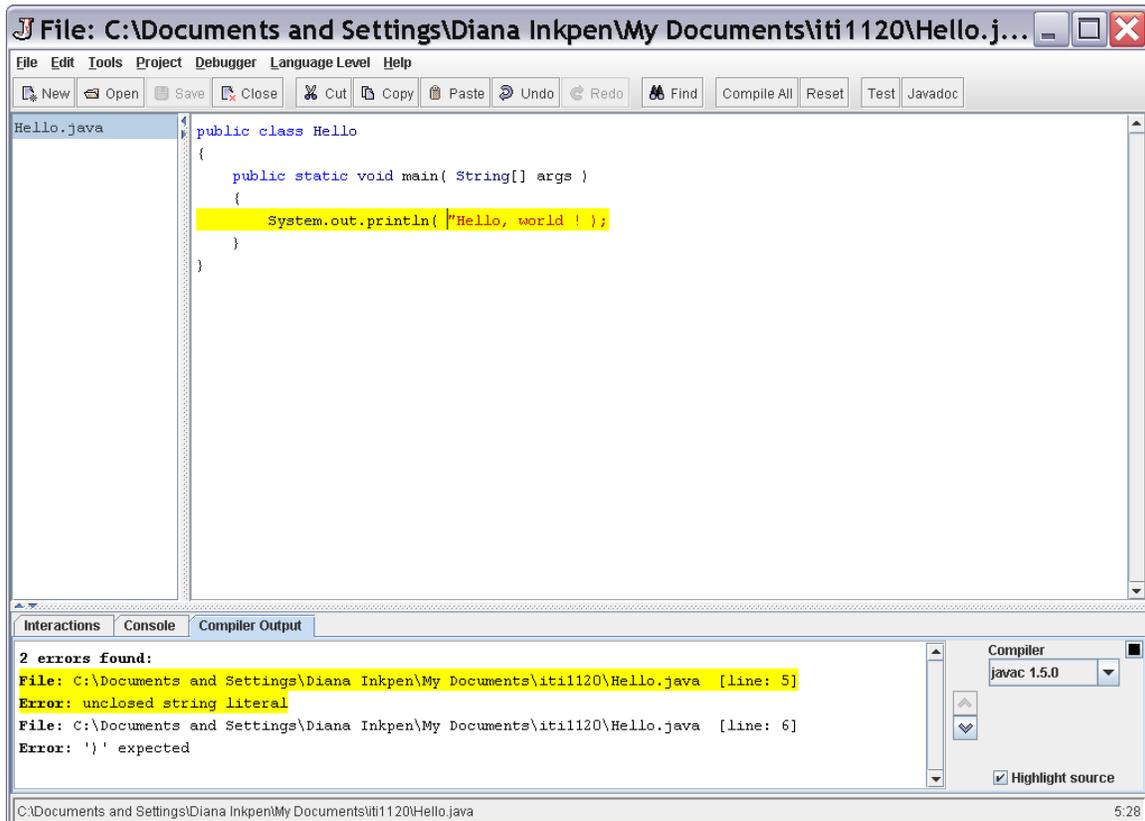
Note that (untitled)\* in the left area has changed to `Hello.java`, and the \* has been removed to indicate that there are no unsaved changes. The complete folder path of the file also appears at the bottom of the screen.

The buttons in the Dr. Java window related to operations with code are the “Compile”, “Run”, “Test”, and “Javadoc”. A Java program must be compiled before it can be run. The compiler checks that your program obeys the rules of the Java language, and then translates the code into a format that the machine can understand. Clicking the “Compile” button will compile the current file. Output from the compiler will appear in the lower area. If everything was okay, you will see the message “Last compilation completed successfully” in the compiler output view.

To run a program in Dr. Java, you can use the “Run” button from the left corner of the screen. To run a program from a console window, you could have typed `java Hello`, where `java` is the command to run a Java `.class` file. This command has appeared in the Interactions window as a result of selecting to run the method from the menu. Some people prefer to use menus, while others prefer to type commands.

Beneath the command is the output from your program. We have successfully printed “Hello, world!” If you want to see a view of the program output without the interaction commands, click on the Console tab.

When there is a compile error, you may see something like Figure 6 (the figure shows an older version of Dr. Java):



**Figure 6: Result when there is a compile error**

In this case, the closing " has been omitted in the `println( )` statement.

For each problem, the compiler message indicates the file and line number of the problem, followed by a message of what the compiler believes is the problem. For the first problem, both the message and the corresponding line in the file are highlighted in yellow. If you click on the second problem in the compiler output window, the highlighting will move to the location of that problem, as shown in Figure 7.

The message **unclosed string literal** means that the compiler could not find the closing double-quote character to indicate the end of what to print on the screen.

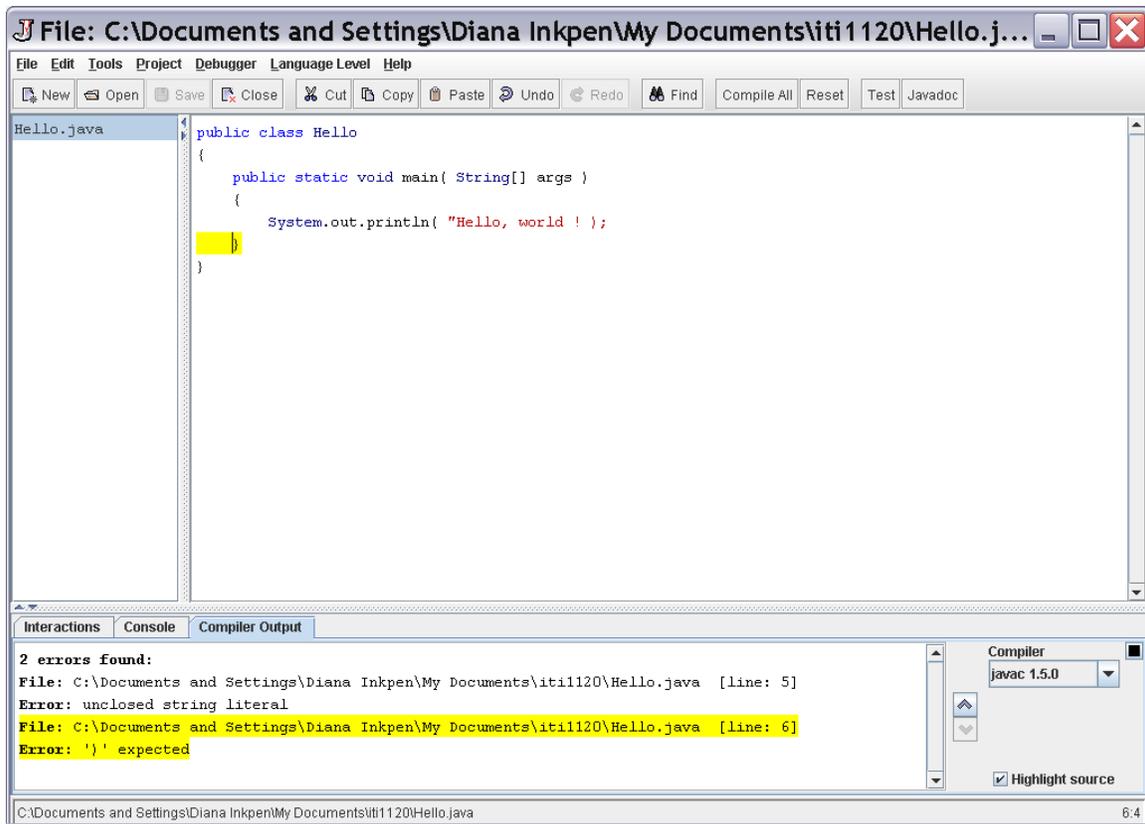


Figure 7: Locating errors

The second error message, `)' expected` is related to the same problem, and can be confusing because the `)` is actually present in the program. If you look carefully at Figure 6 and 7, you may notice that the characters `)` and `;` are coloured in red, and this means that the compiler is assuming that these characters are part of what is included in the message to be printed on the screen. It is only after the compiler reaches the end of the line that it realizes that the `"` is missing. Now, the compiler wants to match the `)` character for the `println( )` statement, but the next character is a brace on the following line and the compiler reports an error at that point.

In fact, restoring the `"` will eliminate both error messages.

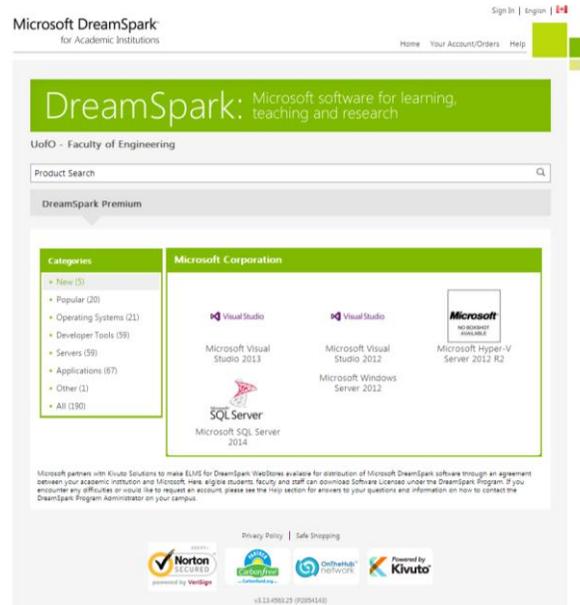
As you can see, interpreting compiler messages can be tricky and this is one of the things you will have to learn in the course. Sometimes, as in the case we have seen here, several compiler messages can result from just one problem in the code. In other cases, after fixing one compiler problem, more problems that were obscured by the first problem can appear during the next compile. If the messages from the compiler do not appear to make any sense, it helps to remember that the compiler reads one character of your program at a time in the forward direction, and will not back up. The compiler can only print a message when it is sure there is a problem, and that is why it cannot tell that the `)` was not meant to be part of the message to be printed.

## Installing Microsoft Visio from Microsoft DreamSpark

Please follow these steps to download Microsoft Office Visio Professional 2007:

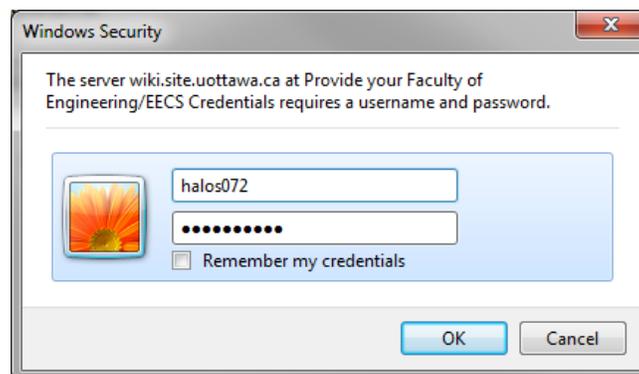
1. Go to the Microsoft DreamSpark for Academic Institutions software center:

<http://e5.onthehub.com/WebStore/ProductsByMajorVersionList.aspx?ws=f911074f-b08b-e011-969d-0030487d8897&vsro=8>

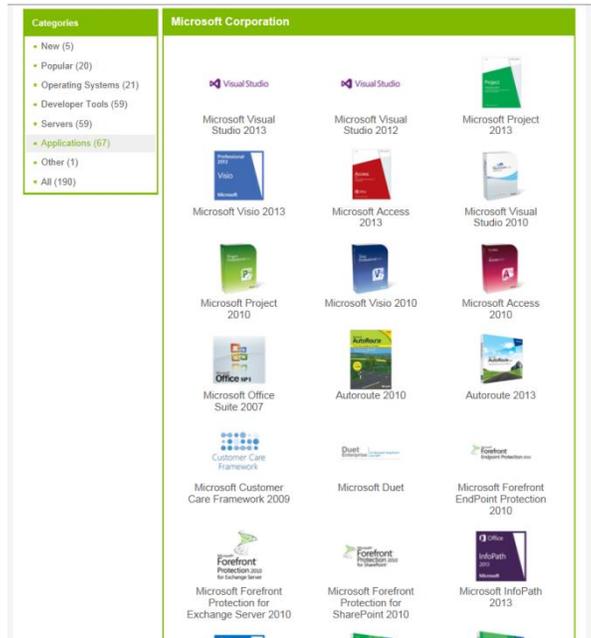


2. Use the “Sign In” link on the top of the page to log into your DreamSpark account (use your EECS computer account information). If you do not currently have an EECS computer account, you can use the following website to generate one:

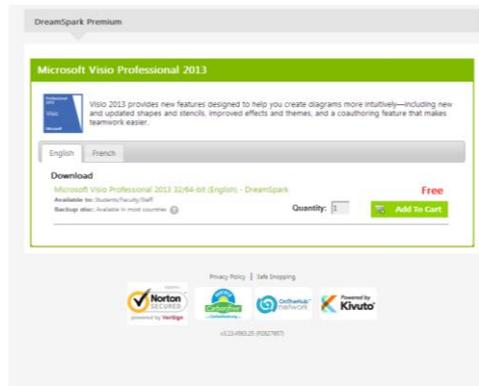
<https://www.site.uottawa.ca/autopass/>



3. Click on the “Applications” link from the “Categories” side menu



4. Choose “Microsoft Visio 2010” or “Microsoft Visio 2013” and in the next page, Click on the “Add to Cart” button for “Microsoft Office Visio Professional 2010” or “Microsoft Office Visio Professional 2013”.



5. Follow the website’s instructions to download the application.