

Providence College High School Programming Contest 2008 Reference

Electronic *PDF* version and zip file of practice problems
are at <http://www.providence.edu/mcs/rhc/>

Table of Contents




Introduction.....	1
<i>PC²</i> Reference	3
C++ Reference	5
<i>Java Reference</i>	7
<i>VB .NET</i> Reference	10
<i>QBasic</i> Reference.....	12
<i>Visual Basic 6</i> Reference.....	13

Introduction

As described below, the contest uses *Windows XP* with desktop icons for the languages and program submission. Instant messaging during the contest is prohibited and grounds for disqualification.

Desktop:

The desktop system is *Windows XP*. Your computer will already be started, so please do not start another one. The computer will have desktop icons for submitting a problem, the languages, *Windows Explorer*, *Notepad*, *WordPad*, and this document as shown below:

	← Start <i>PC²</i> (program submission)		← Start <i>VB .NET</i>
	← Start <i>Java</i> (<i>JCreator LE</i>)		← Start <i>Windows Explorer</i>
	← Start <i>C++</i> (<i>Visual Studio</i>)		← Start <i>Notepad</i>
	← Start <i>QBasic</i>		← Start <i>WordPad</i>
	← Start <i>VB6</i>		← This document

Disk Drives:

- Drive E This is the *USB Flash Drive*. Keep all your program files here.
Do not move or remove the flash drive – it is grounds for disqualification.
If your computer fails, it will be moved to the new computer.
- Drive D May be used, but there is no need to.
- Drive C DO NOT USE – contents are lost on reboot
- Drive A The floppy drive is not be used during the contest.

Languages:

The contest supports the following languages:

- C++* uses *Visual Studio .NET 2005*
ap include files are not supported (they were not ported to *.NET 2005*)
- Java* uses *JCreator LE* with the *jdb* debugger and *Java 1.6.0_01*
contestants write *applications* not *applets*
- QBasic*
V(isual)B(asic) 6
VB .NET

Program Submission:

Programs are submitted electronically using the *PC² System* from the *University of California, Sacramento*. The judges will run a program just as you do, but will use their data.

Printing:

Printing is allowed – see the reference for a particular language.

Logon and Practice:

Log onto to the *PC² System* (see [PC² Reference](#)). Open the languages you will be using and try the *Practice* problem(s) (see each language reference). If you are using many languages, it is recommended that only one or two be open at a time. Try printing.

There is a practice session before the contest. Try submitting the *Practice* problem in the language(s) you plan to use. Submissions of the *Practice* problem during the contest will be ignored. Try submitting a *Clarification*.

If the computer fails or is much too slow:

Call the lab supervisor. You will be moved to another computer with your *USB Flash Drive*.

If the *PC²* server stops:

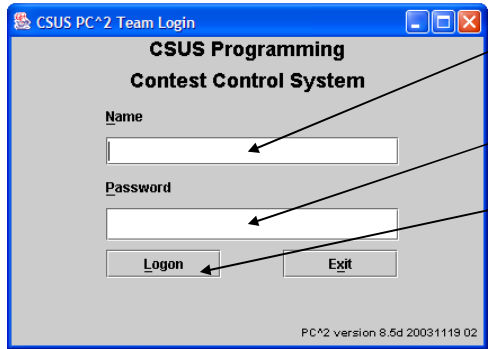
The contest clock will stop until the server is restarted, but you can continue to work. You will be asked to *Exit* and *Logon* again when the server is restarted.

PC² Reference

The programs will be submitted electronically using the PC² System from the *University of California, Sacramento*.

Logon:

Double click on the *TEAM* icon (black *T*) on the desktop and the following pop-up will appear.



Type *teamnn* (*nn* is your team number) in the *Name* box.
Examples: *team1 team12*

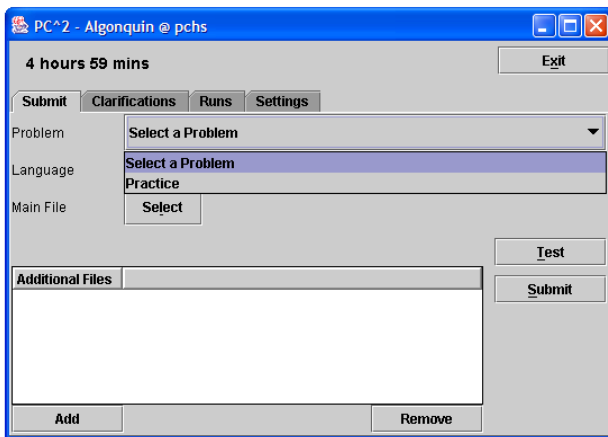
Type the team number in the *Password* box.

Click on *Logon*.

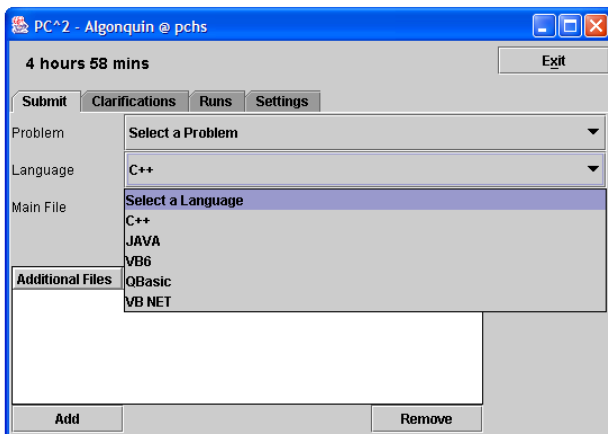
When the *Team* dialog box appears, click on the *Settings* tab and immediately change the password.

Submit a Program:

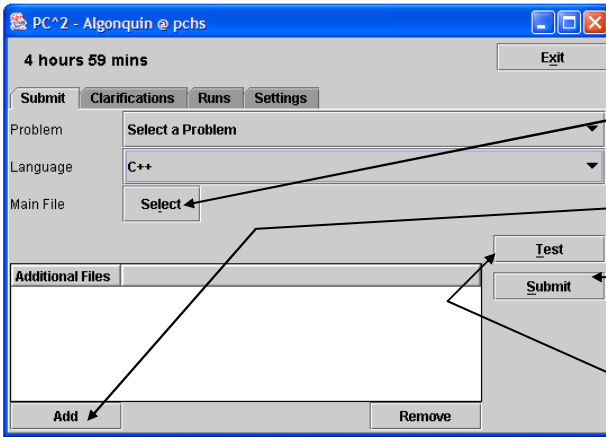
If needed, click on the *Submit* tab in the *Team* dialog box and proceed as follows:



← 1. Select the problem from the *Problem* drop down



← 2. Select the language from the *Language* drop down



3. Click on *Select* and browse for the source file (varies with language, see language reference).
 4. For *VB6* only, click on *Add* and browse for the form (.frm) file. (see [VB6 Reference](#))
 5. Click on *Submit* and then click on *Submit* in the pop-up. A confirmation will be sent by the server. A separate message will be sent when the problem is judged.
- Do not use the *Test* button.
It is used for a different style of contest and does not work.

View Status of Submissions:

Click on the *Runs* tab in *Team* dialog box. Click *Refresh*.

Request a Clarification:

To *Request* a clarification, click on the *Clarification* tab, click on *Request Clar*.

In the *Clarification Request* pop-up:

Use the *Select a Problem* drop-down to choose either a *General* clarification or a clarification for a specific problem.

Enter your question.

Click on *Submit*.

The reply will appear in a pop up and can also be viewed as described below.

View a Clarification Reply:

A clarification reply (yours and those broadcast to all) will appear in a pop-up and can also be viewed by selecting the *Clarifications* tab, selecting a specific clarification, and clicking on *View Clar*.

Contest Clock:

The clock on the *Team Dialog Box* is unofficial. The time will be announced periodically.

If the *PC²* server stops:

The contest clock will stop until the server is restarted, but you can continue to work. You will be asked to *Exit* and *Logon* again when the server is restarted.

C++ Reference

The contest uses C++ in *Microsoft Visual .NET 2005*. The *ap* include files are not in *VS .NET 2005*.

To Start C++:

Double click in the *Visual Studio C++* icon on the desktop.

To Open or Create a C++ Program:

Click on the *PCGO* button in the tool bar (see *Figure 1*)

To open an existing C++ program:

Select the desired file in the *PC New/Open* window

Click on *Open*

To create a new C++ program:

In the *File name* box:

Type the program name (NO BLANKS, NO CAPS, <=8 CHARS)

No need to type the *.cpp* suffix

Click on *Open*

Wait for the typing icon and the expression "*Global Scope*" to appear. (Slow first time.)

Special note about File/Save As: ← Important !!

The *PCGO* button creates all the underlying *Solution* files. If you create a new file by using *File/Save As*, you must create a new *Solution* by clicking on *PCGO* which will reload the program. *Figure 1* shows how to detect a mismatch between the source (*.cpp*) file and *Solution* files.

To Run a Program:

Select *Debug/Start Without Debugging* or strike *Ctrl + F5*.

Select *Yes* in the pop-up.

To Print a Program:

Select *File/Print*

To Close a Program:

Select *File/Close Solution*

(It is not necessary to *Close* before using the *PC* button to open another program.)

Submitting a C++ Program:

Here is the script, also see the [PC² Reference](#).

If needed, click on the *Submit* tab in the *Team* dialog box.

Click on the *Problem* drop-down and select the problem.

Click on the *Language* drop down and select C++.

Click on the *Select* button and browse for the *.cpp* file.

Click on *Submit*, then click on *Submit* in the pop-up.

You will receive a confirmation message in reply.

A separate message will be sent when the program is judged.

Do not use the *Test* button – not needed and does not work.

Practice:

E:\HS_practice has sub-directory *c++_practice* which contains *practice.cpp*. Open and run it. Try submitting the program during the practice session (choose *Practice* in the *Problem* drop down). Here is a listing of *practice.cpp*:

```
// practice.cpp
#include <iostream>
#include <string>
using namespace std;

int main()
{
    int x;
    string s;

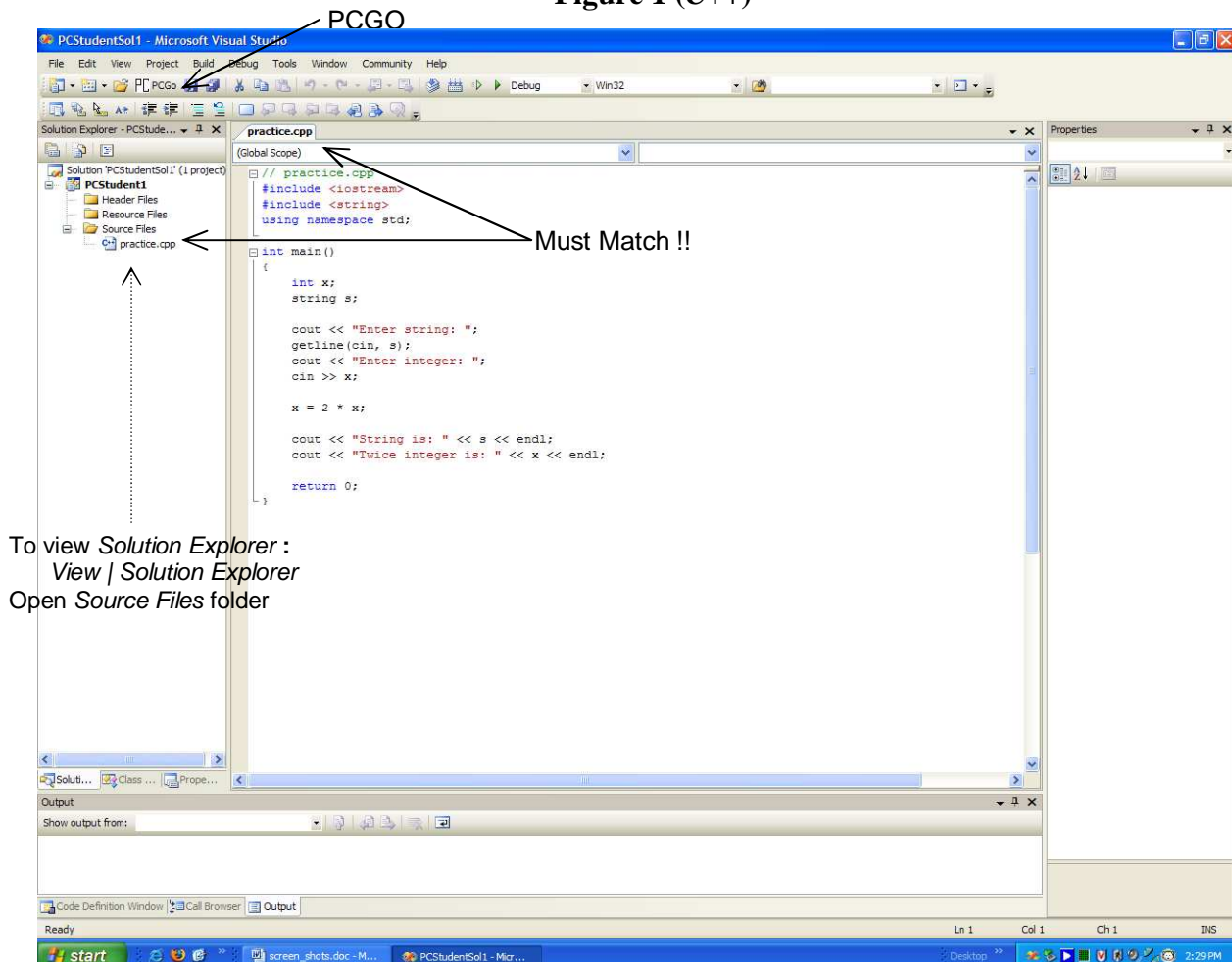
    cout << "Enter string: ";
    getline(cin, s);
    cout << "Enter integer: ";
    cin >> x;

    x = 2 * x;

    cout << "String is: " << s << endl;
    cout << "Twice integer is: " << x << endl;

    return 0;
}
```

Figure 1 (C++)



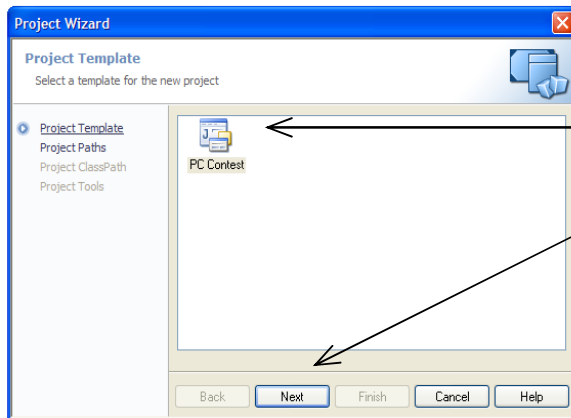
Java Reference

The contest uses *JCreatorV4 LE* with the *jdb* debugger and *java 1.6.0_01*. Only *Java applications* will be used.

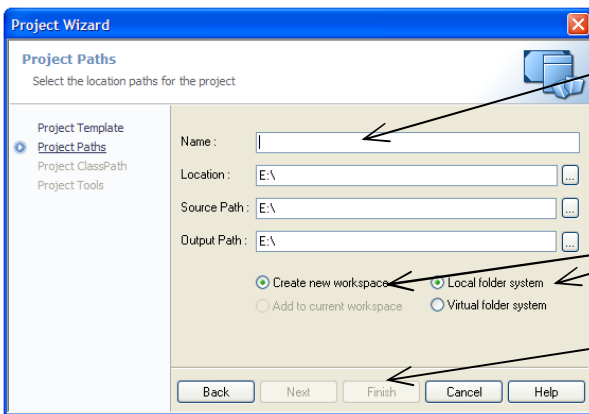
ONLY 1 FILE PER PROJECT AND ONLY 1 PROJECT PER WORKSPACE PLEASE!!

To Start JCreator LE: Double click on the *JCreator LE* icon on the desktop.

To Create a Workspace:



1. Select *File|New Project* – you will see the first pop-up
2. Click on the *PC Contest* template
3. Click on *Next* – a new pop-up will appear



4. Type the *Name* of the project.
NO BLANKS, NO CAPS, <=8 CHARS IN Name !!
A directory of that name will be created.
As you type, the other boxes will auto complete.
Do not change the other fields !!
5. *Create New Workspace* is already selected
Click on *Local Folder System*, if needed.
6. Click on *Finish*.
It may prompt to save the current workspace.
The *Project Creation* pop-up will appear.
Click on *Finish* when it completes.

JCreator LE creates a program shell – here it is for the *practice* project:

```
// PC Contest Shell
// Project: practice

// TODO, insert import(s)

public class practice { ←————— Creates public class - same name as project
    public static void main(String[] args) {
        // TODO, add your application code
    }
}
```

To Open an Existing Workspace:

Select *File/Open Workspace*

Browse for the *.jcw* file in the project's folder and double click.

To Close a Workspace: Select *File/Close Workspace* and click *Yes* or *Save* as needed in the pop-ups.

To Compile: Select *Build/Compile Project* or strike *F7*.

To Execute: Select *Build/Execute Project* or strike *F5*.

To Save a File: Select *File/Save* or strike *Ctrl+S* (do this often)

To Print: Click on *File/Print*.

To Debug:

The *jdb* debugger is available.

To set up *jdb*:

If needed, open the program's workspace.

Select *Project/Project Settings*

In the pop-up:

Click on *JDK Tools*

Click on *Run Application* in the *Select Tool Type* drop-down

Click on the *Debug* box

Click on *OK*

To use *jdb*:

Execute the program as usual and a debugging window will appear.

Type the *jdb* commands such as *step* in the debugging window.

Be careful: The debug window can cover dialog boxes – move it to see the dialogs.

When done, close debug window.

To stop using *jdb*:

Use the *Set up jdb* script above, but click on the *Default* box instead of *Debug*.

Tips:

1. It is difficult to detect that a *Java* program is in a loop. If a program does not complete, strike *Alt+Ctrl+Del* and open *Task Manager*. Look for too many *java* executions or executions that consume too much *CPU* time. Call a lab assistant the first time you try it.
2. The *jdb* debugger can be difficult to use. Practice first if you plan to use it.

Submitting a Java Program:

Here is the script, also see the [PC² Reference](#).

If needed, click on the *Submit* tab in the *Team Dialog Box*.

Click on the *Problem* drop-down and select the problem.

Click on the *Language* drop down and select *Java*.

Click on the *Select* button and browse for the *.java* file.

Be careful: The *.java* file is in the project directory

Example: If the project is named *problem1*, then the *.java* file is in the *problem1* directory

Click on *Submit*, then click on *Submit* in the pop-up.

You will receive a confirmation message in reply.

A separate message will be sent when the program is judged.

Do not use the *Test* button – not needed and does not work.

Practice:

E:\HS_practice has 3 Java sub-directories of the form: *java_practice_N* (N is 1,2,or 3)

Each directory has a sub-directory named *practice* which contains *practice.java* and a *Workspace*.

Each program is a different style for the same problem.

Use any of the styles, but create only 1 .java file (no separate files for *JFrames*, etc.)

practice.java in *java_practice_1\practice*:

Open the *practice* workspace under *java_practice_1/practice*

Submit *practice.java* under *java_practice_1/practice*

Here is the listing:

```
// PC Contest Shell
// Project: practice

// TODO, insert import(s)
import java.util.*;

public class practice {
    public static void main(String[ ] args) throws {
        // TODO, add your application code
        Scanner in = new Scanner(System.in);
        String s;
        int x;
        System.out.println( "Enter string: ");
        s = in.nextLine();
        System.out.println( "Enter integer: ");
        x = in.nextInt();
        x = 2 * x;
        System.out.println("String is: " + s);
        System.out.println("Twice integer is: " + x);
    }
}
```

Uses *Scanner*, *nextLine*, *nextInt*, and *println*.

Do NOT USE *print* - system may hang!!

practice.java in *java_practice_2\practice*:

Open the *practice* workspace under *java_practice_2/practice*

Submit *practice.java* under *java_practice_2/practice*

Here is the listing:

```
// PC Contest Shell
// Project: practice

// TODO, insert import(s)
import javax.swing.*;

public class practice {
    public static void main(String[ ] args) {
        // TODO, add your application code
        String s, sx;
        int x;
        s = JOptionPane.showInputDialog(null, "Enter a string", "String", JOptionPane.QUESTION_MESSAGE);
        if(s != null) {
            sx = JOptionPane.showInputDialog(null, "Enter an integer", "Integer", JOptionPane.QUESTION_MESSAGE);
            if(sx != null) {
                x = Integer.parseInt(sx);
                x = 2 * x;
                JOptionPane.showMessageDialog(null, s+"\n"+Integer.toString(x), "String and Twice Integer",
                    JOptionPane.INFORMATION_MESSAGE);
            }
        }
    }
}
```

Uses Dialog boxes

practice.java in *java_practice_3\practice*:

Open the *practice* workspace under *java_practice_/practice*

Submit *practice.java* under *java_practice_3/practice*

Here is the listing:

```
// PC Contest Shell
// Project: practice

// TODO, insert import(s)
import javax.swing.*;

public class practice {
    public static void main(String[] args) {
        // Code Goes Here ...
        TextWindow textWindow = new TextWindow();
        textWindow.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        textWindow.setSize(520,520);
        textWindow.setVisible(true);
    }
}

class TextWindow extends JFrame {           //JFrame friendly subclass - MUST BE SAME IN FILE AS main !!
    JTextArea textArea;
    String s, sx;
    int x;

    public TextWindow() {                   //default constructor
        super("practice");                  //call constructor with frame title
        textArea = new JTextArea();        //construct textArea with default params
        add(new JScrollPane(textArea));    //add textArea wrapped in a scroll pane

        s = JOptionPane.showInputDialog(null, "Enter a string", "String", JOptionPane.QUESTION_MESSAGE);
        if(s != null) {
            sx = JOptionPane.showInputDialog(null, "Enter an integer", "Integer", JOptionPane.QUESTION_MESSAGE);
            if(sx != null) {
                x = Integer.parseInt(sx);
                x = 2 * x;
                textArea.append(s + "\n");
                textArea.append(Integer.toString(x));
            }
        }
    }
}
}
```

Uses subclass of *JFrame* as friendly class in the same *.java* file!!

ONLY 1 FILE!!

VB .NET Reference

To Start :

Double click on the *Visual Studio VB.NET* icon on the desktop.

It will open slowly the first time as *Visual Studio* switches to *VB. NET*.

To Create a Program:

Select *File/New/Project*

In the *New Project* pop-up:

Click on *Console Application*

In the *Name* box, type the project name

Click on *OK*

NO BLANKS, NO CAPS,
<=8 CHARS IN NAME !!

Immediately save the project to the flash drive by selecting *File/Save All*:

In the *Save Project* pop-up: change *Location* to *E:* and click on *Save*

Immediately insert *Imports System* as the first statement in the *Module* file:

(otherwise the program will not work when submitted)

To Open an Existing Program:

Select *File/Open/Project*

Find the directory containing the project and open it.

Double click on the project file.

If needed, double click on the module (.vb) file in the *Solution Explorer* pane.

To Run a Program:

Select *Debug/Start without Debugging* or strike *Ctrl+F5*.

To Print:

Select *File/Print*

To Close:

Select *File/Close Solution*

Known Bug:

Do not use *Console.Read* inside a function call. Example: `function_name(Console.Read(. . .))`

Submitting a VB .NET Program:

Before submitting, be sure the first statement is *Imports System*.

Here is the script, but also see the [PC² Reference](#).

If needed, click on the *Submit* tab in the *Team Dialog Box*.

Click on the *Problem* drop-down and select the problem.

Click on the *Language* drop down and select *VB NET*.

Click on the *Select* button and browse for the “module” .vb file. (Not the *AssemblyInfo.vb* file)

Click on *Submit*, then click on *Submit* in the pop-up.

You will receive a confirmation message in reply.

A separate message will be sent when the program is judged.

Do not use the *Test* button – not needed and does not work.

Practice:

The directory *E:\HS_practice* contains the directory *vbnet_practice* which in turn contains the *practice* directory. Try opening project in the *practice* directory and running the program. Try submitting the program (*module1.vb*) during the practice session (choose *Practice* in the *Problem* drop down). Here is a listing of *module1.vb*:

Imports System ← Be sure *Imports System* is first!!
Module Module1

```
Sub Main()
  Dim s As String
  Dim x As Integer
  Console.Write("Enter string: ")
  s = Console.ReadLine()
  Console.Write("Enter integer: ")
  x = Console.ReadLine()
  x = 2 * x
  Console.WriteLine("String is: " + s)
  Console.Write("Twice integer is: ")
  Console.WriteLine(x)

```

End Sub

End Module

***QBasic* Reference**

QBasic runs in *DOS* mode and can be run either in *Windows XP* or in a separate *DOS* window. The judges use a different *QBasic* version that produces an executable, but there have been no compatibility problems.

To Start:

- Double click on the desktop icon.
- Strike *Esc* to clear the initial pop-up.
- Strike *Alt+Enter* to switch to a full *DOS* window.
- QBasic* will appear as a full screen.

To Switch Between *DOS* and *Windows*: Strike *Alt+Enter*

To Use the *QBasic* Menu:

- Either use the mouse to position to *File*, etc. or strike the *Alt* key and the first letter (e.g., *Alt+F* for *File*).
- Use the mouse, arrow keys, or highlighted letter to access an item within a menu.

To Save or Open a File:

- Select *File/Save* or *File/Save As* or *File/Open* as needed
- Click on the scrollbar under *Dirs/Drives* to find drive *E*

← Drives are at bottom of list !!
NO BLANKS, NO CAPS,
<=8 CHARS IN NAME !!

To Run a Program: Select *Run/Start* or strike *Shift + F5*

To Print:

- Do not print from *QBasic*, because the system will hang.
- If needed, switch to *Windows XP*.
- Use the desktop icon to open *Notepad*.
- Select *File/Open* and then select *All Files* in the *Files of Type* dropdown.
- Open the *.bas* file and select *File/Print*

To Close *QBasic*: Select *File/Exit* in *QBasic*

Submitting a *QBasic* Program:

Here is the script, also see the [PC² Reference](#).

- If needed, click on the *Submit* tab in the *Team Dialog Box*.
- Click on the *Problem* drop-down and select the problem.
- Click on the *Language* drop down and select *QBasic*.
- Click on the *Select* button and browse for the *.bas* file.
- Click on *Submit*, then click on *Submit* in the pop-up.
- You will receive a confirmation message in reply.
- A separate message will be sent when the program is judged.

Do not use the *Test* button – not needed and does not work.

Practice:

E:\HS_practice has sub-directory *qb_practice* which contains *practice.bas*. Open and run it. It has a *CLS* statement which is allowed, but it is easier for the judges if it is not used. Try submitting the program during the practice session (choose *Practice* in the *Problem* drop down). Here is a listing of *practice.bas*:

```
CLS
INPUT "Enter string: "; s$
INPUT "Enter integer: "; x
x = 2 * x
PRINT "String is: "; s$
PRINT "Twice integer is: "; x
END
```

Visual Basic 6 Reference

To Start: Double click on the *Visual Basic 6* icon.

To Create a Project:

Select *File/New Project*

Double click on the *Standard EXE* icon ← NO BLANKS, NO CAPS, <=8 CHARS
IN PROJECT NAME !!

To Open an Existing Project:

Select *File/Open Project*

Browse for the project (.vbp) file

If needed, double click on the form (.frm) file in the *Project* pane.

To Run a Program: Select *Run/Start*, strike *F5*, or click on the *Start* button.

To Stop a Program: Select *Run/End* or click on the *End* icon.

To Print: Select *File/Print*

To Save the Project: Select *File/Save Project* or *File/Save Project As*

Submitting a VB6 Program:

Here is the script, also see the [PC² Reference](#).

If needed, click on the *Submit* tab in the *Team Dialog Box*.

Click on the *Problem* drop-down and select the problem.

Click on the *Language* drop down and select *VB6*.

Click on the *Select* button and browse for the project(.vbp) file.

Now add the form file: Click on *Add* and browse for the form (.frm) file.

Click on *Submit*, then click on *Submit* in the pop-up.

You will receive a confirmation message in reply.

A separate message will be sent when the program is judged.

Do not use the *Test* button – not needed and does not work.

Practice:

E:\HS_practice contains has sub-directory *vb6_practice* which contains project file *practice.vbp* and form file *practice.frm*. Open and run it. Try submitting the program during the practice session (choose *Practice* in the *Problem* drop down). Here is a listing of *practice.frm*:

```
Private Sub Command1_Click()  
    Dim s As String  
    Dim x As Integer  
    s = Text1.Text  
    x = Text2.Text  
    x = 2 * x  
    Print "String is : " + s  
    Print "Twice integer is: " + CStr(x)  
End Sub
```