Lab 4 Part III Using Arrays and FOR Loops

Instruction Conventions

⇨	When the "process" icon appears in the box on the left, then this box will contain one or more instructions that you will need to follow. The instructions will be a specific as is practical but could be different for different users and computer configurations.
📝	When the "note" icon appears in the box to the left, this box will contain notes, hints, or tips that may be helpful to the lab activities.
⌨	<user input>
	When the keyboard icon appears in the box to the left, the box above will contain a line of input to be entered by the user, and this box will contain an explanation of what the user input will do.

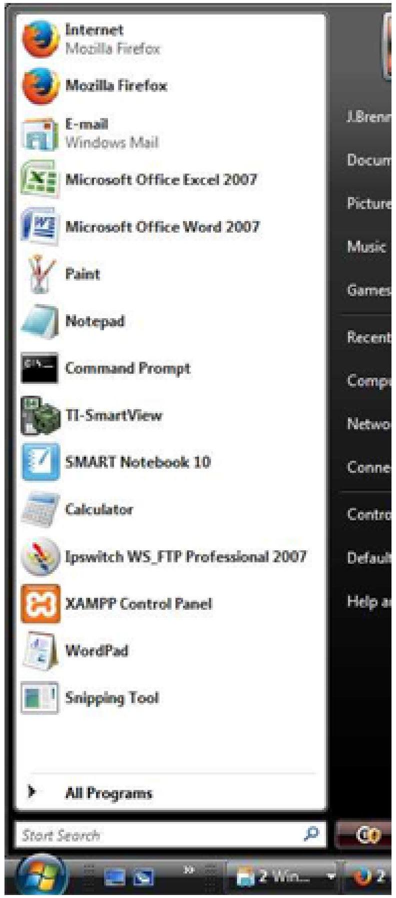
0. Using a web browser, goto HWMath.net/IBCS and open the link Trace-3.bas file.

   This is a program shell with an outline of the task you are asked to perform. Save this to your C: drive where you have blassic.exe installed and other BASIC programs.

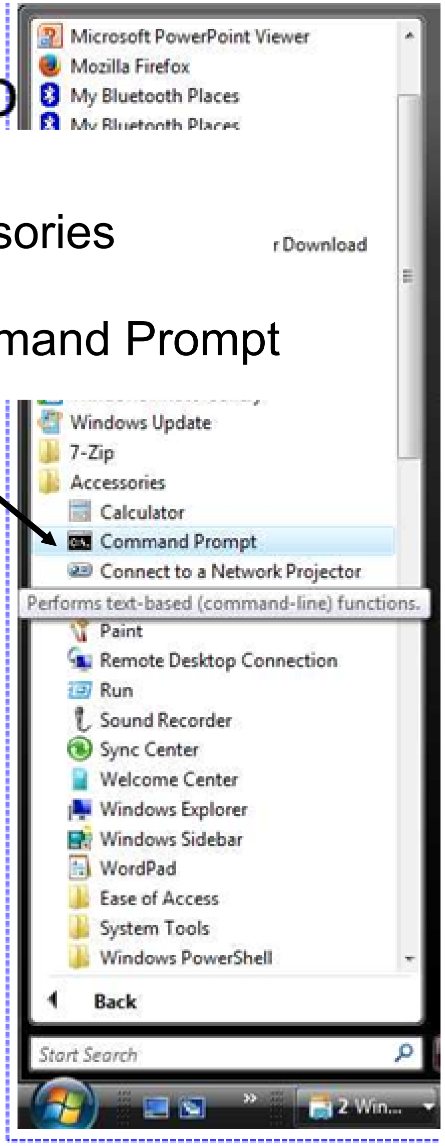
   You can do this by starting Notepad, copying the text from the open Trace-3.bas program, or by right-clicking on the link and using:
   Save Link As

Save the file with the new name
Primes.bas

1) From the start menu select All Programs

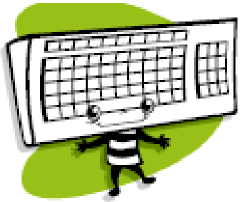


2) From Accessories Select Command Prompt



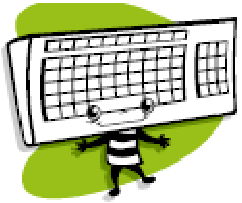
If you are using a different computer than you have used before, be sure to copy blassic.exe from the CD into your work space.

```
copy d:\blassic\blassic.exe
```



1. Update the file header:
   a. Update the File name to Primes.bas
   b. put your name as a REMark on the second line of the file.
   c. Update the real purpose of the program

For this version of Primes.bas we will keep the I and J loops as they are - using GOTO statements at the end of the loops to jump back to the start of the loops.

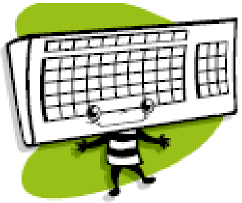


2. Keep track of the prime numbers using an array

```
700 IF F > 0 THEN 800
701    REM If f = 0 then do this
710    PRINT "P: ",I
720    P = P + 1
```

a. As prime numbers are found, put them into an array.

b. It is okay to print the primes as you develop your program - but the final product should not print them out here.



3. After the program finds all N prime numbers print them out in reverse order - from highest to lowest, using a FOR loop.

- Somewhere after line 900 would be good to put a FOR loop to print out all of the N prime numbers.

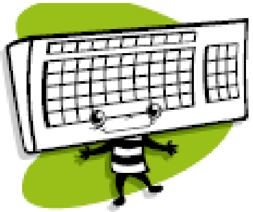
- For a first version of your program you could use a loop such as

```
FOR I = 1 TO N
```

- For the final version of your program you could use a loop such as

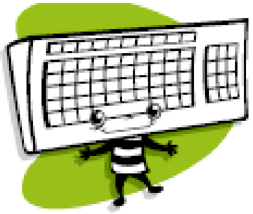
```
FOR I = N TO 1 STEP -1
```

- It is safe to re-use the variable `I` because it is no longer in use.



4. Test your program to make sure it finds (and prints) N prime numbers. Since N controls how many primes are found and printed, while you are developing the program you could set N lower.

5. When you are satisfied with your program set N to be 20 and re-run your program. Make a list of the 20 prime numbers that your program displays.



ooops

6. When you use an array, BASIC will allocate enough space for 10 entries. You can have more, but you need to declare the size of your array using the DIM statement.

   It is good practice to declare variables near the top of your program. Somewhere before line 200 add a line similar to :

   ```
   190  DIM A(100)
   ```

   This declares an array named A and gives it enough space to hold 100 prime numbers.

   You can keep the name of your array that holds primes, just give it a bigger size 🙂

```
100 REM File : Trace-3.bas
101 REM Trace the program to determine its output.
102 REM You may want to keep a table of variable values and line numbers
110 N = 10
115 P = 0
120 I = 2
200 REM Top-of-loop
210 IF P = N THEN 900
220 F = 0
230 M = I \ 2
250 J = 2
500 IF J > M THEN 700
503    R = I MOD J
510    IF R > 0 THEN 600
511       REM if r = 0 then do this:
520       F = F + 1
600    J = J + 1
620    GOTO 500
700 IF F > 0 THEN 800
701    REM If f = 0 then do this
710    PRINT "P: ",I
720    P = P + 1
800 I = I + 1
810 GOTO 200
900 REM End-of-Loop
910 PRINT "Program terminating."
```