A Computer Program in Hydrology for Landscape Architecture.

The Document - toward the partial completion
of the requirements for the Degree of
Bachelor of Landscape Architecture.
VPI and SU

By:
Timothy R. Quillen
Rt.4 Box 317
Waynesboro, VA. 22980

June, 8 1987

Approved By:

Mr. Mike Deeter	ADVISOR
Mr. Dean Bork	INSTRUCTOR
Dr. Patrick A. Miller	CHAIRMAN

Attention Patron: The accompanying computer disk is no longer available. We regret any inconvenience.

Dedication:

I dedicate this project to my loving parents. Without their love and support, this project would not exist.

Also to all the others who kicked me in the butt now and then to keep me going fairly smoothly. Of those with the largest butt kicking boots, I would especially like to thank my advisor, Mike Deeter, for all his time, patience and help. The other major contributors to my project are Dean Bork, Will Shepherd, Dr. Patrick Miller, Dr. Jerry Mann, David Gayer, Andrew Roy, Hammond Eve, Ben Johnson and anyone else who wants to share the blame.

Thank you.

Preface

Last year I was working on a project for one of my classes and storm water management was one of the issues to be addressed. I looked everywhere for a complete manual, and could not find any. I had to use three copies of the same book in order to get all the charts I needed to complete the task. In order to correctly, complete the task of storm water design, the authors of this book provides an example for the user to follow, this example only helps with the very basic things like, "What am I suppose to do now?", rather than the more complicated question of,"How do I do this?". Well, after getting help from everybody that was around and still not receiving an acceptable answer, I finally had to ask Dean Bork, our Annex hydrologist. That is when I decided there must be a better way to complete the task of storm water design.

Luckily, I just got my computer and I realized that it would be possible to put the Rational Method into a program. There would be no need for a book, therefore you would not have to worry about lost pages. You would be asked by the program, what it needs to know to complete the task. All of the charts will be incorporated into the program, so the user will not have to figure out what chart is used where, and how to use them. That is how this project came into existence. If you use it, I hope you enjoy it.

TABLE OF CONTENTS

Introduction

Background

Storm water management is a problem every land planner has to deal with. The problem is we do not have to do it very often; therefore, we have to re-learn the process every time we do it because processes are forgotten.

Being interested in computers and computer applications, I decided to write a program to assist the designer with storm water calculations. This program will be referred to as "Pipe". I believe that completing these calculations on a computer will make life as a Landscape Architect a lot more enjoyable and profitable.

One might ask the question, "How can the computer help?" The computer operates via programs. These programs follow a certain logical progression in order to instruct the computer on what to do next. The Rational Method also follows a logical progression to arrive at the correct answers. Hence, we can easily follow the Rational Method on the computer. Pipe will be a user - friendly program, and will take the user through the entire process of storm water calculations via a variety of choices and questions. Pipe will be written in Turbo Pascal so that it can be run quickly on a wide variety of machines.

The machine I have been using is a Tandy 1200 with a ten megabyte hard drive. It has a color graphics adapter and a Tandy RGB (red-green-blue) high resolution color monitor. Inside the computer, there are two 360 kilobyte floppy disk drives, an internal 1200 baud modem, 2 serial ports, 2 communications ports, 1 parallel printer port, a multifunction board with 640 kilobytes of memory, a clock and a game port with a joystick. The printer is a Tandy DMP 130 (dot matrix printer) which was used to print this document.

To let you know a little more about me, the computer buff, I will discuss the other activities I use the computer for. I also have a Mouse Systems Mouse attached for use with AutoCad and PC-Paint. In the near future, I plan to get a tablet and 8087 math co-processor chip for use with AutoCad. The programs I use the most are MS DOS 2.11, WordPerfect 4.2, PC-Paint 2.00, Turbo Pascal 3.01, Sidekick, Procomm 2.42 (communications package) and AutoCad 2.52. During the past two years, I have been collecting as many useful programs as possible. The present total is over 500 different programs. These are mostly public domain programs downloaded from local bulletin boards (a host computer that is called via a telephone in order to exchange information and programs). Throughout the course of this project, I have been very interested in learning a great deal about computers and what makes them work. And I still have lots to learn.

This document covers a lot of information about why I wrote Pipe, how I wrote Pipe, and the resources I used in order to write Pipe. So, let's start from the beginning with the history, assumptions and limitations of the Rational Method.

History of the Rational Method

This program uses the Rational Method for determining storm water peak runoff. The data is then used to size pipe that is used in construction of storm drainage systems. The origin of this formula is somewhat obscure. In America, the formula was first mentioned in 1889 by Emil Kuichling to determine peak runoff for storm sewer design in Rochester, New York, during the period from 1877 to 1888.

The rational formula is Q = CIA, where Q is the peak discharge in cubic feet per second, C is a runoff coefficient depending on characteristics of the drainage basin, I is the rainfall intensity in inches per hour, and A is the drainage area in acres.

There are six major assumptions involved in the rational formula. They are:

(1) The rate of runoff resulting from any rainfall intensity is a maximum when the rainfall intensity lasts as long or longer than the time of concentration; that is, the largest quantity of water possible is on the site when the rainfall intensity is equal to or greater than the time of concentration.

(2) The maximum runoff resulting from a rainfall intensity, with a duration equal to or greater than the time of concentration, is a simple action of such rainfall intensity; that is, it assumes a straight line relation between Q and I, and Q = 0 when I = 0.

(3) The frequency of peak discharge is the same as that of the rainfall intensity for the given time of concentration; that is, the same design storm is used for both.

(4) The relationship between peak discharge and size of discharge area is the same as the relationship between duration and intensity of rainfall.

(5) The coefficient of runoff is the same for storms of various frequencies. That is, the coefficient has to do with the land cover, not storms.

(6) The coefficient of runoff is the same for all storms on a given watershed.

Model Used

The model used in this program to calculate discharge, is The Virginia Erosion and Sediment Control Handbook, (hereafter termed, VESCH). This model was chosen over similar models because it was prepared by the Virginia Soil and Water Conservation Commission, therefore it is Virginia specific, with greater accuracy than a more general model. The function of the handbook is to establish minimum standards for the control of erosion and sedimentation from land disturbing activities in Virginia, and to provide guidelines for the implementation of those standards in accordance with state law. Chapter 5 in the VESCH contains recommended procedures for calculating runoff, determining detention storage volumes, and designing and analyzing the adequacy of open channels. While the design of large or complex drainage systems and structures should be left to qualified professional engineers, it is felt that the calculation procedures presented in chapter 5 and the program I wrote,'Pipe', are adequate for the design of most erosion and sediment control measures.

The model used to calculate pipe size and velocity is the Design Databook for Civil Engineers by Seelye. This book has been and probably always will be, one of the best and most used references for Civil Engineers and Landscape Architects who need general information on engineering calculations.

Objective of Study

To write a workable program in Turbo Pascal for the purpose of computing the entire process of storm drainage, using the Virginia Erosion and Sediment Control Handbook's chapter five, as the model. And to apply it in a former hydrology project.

Final Products

The final products will be:

- a computer disk with:
  1. the Pipe program
  2. source code
  3. support programs
  4. document files

  all to be run on an MS DOS machine.
- a two board abstract in 11 inches by 14 inches horizontal format
- a document, discussing everything I have done this year on the project

Process of Programming

Selecting Turbo Pascal

When I first developed this project, I was going to program with the Basic language. I already knew Basic, therefore, I would not have to learn a new language. But Basic has major drawbacks, such as:

1. Most basic programs are written without structure; therefore, an interpreter is needed rather than a compiler. This means, from the DOS prompt you must first start one program to run another program.
   But with a compiler, from the DOS prompt you can jump right into a program. This is due to the fact that a compiled program is stored as a COM file in machine code, and a interpreted file is converted into machine code as the interpreter winds its way through the program.
2. Due to the above information, it would follow that compiled programs are faster, which is indeed the case. But remember, compiled Basic programs are still slower than a compiled program written in a high level language.
3. A COM file can also be used on any MS-DOS machine, whereas, almost every machine has a slightly different Basic interpreter.

I felt that the above limitations were severe enough to warrant me to write 'Pipe' in a different language. I considered many languages including; Fortran, Pascal, PL/1, APL, and C.

After discussions with Andrew Roy and Rob Bardsley, a couple of computer contacts I have, about these languages, I was advised which language to consider and which language to avoid. Of the above languages Fortran, PL/1, and APL were those I was advised to avoid, simply because they are not very structured languages. Therefore, these languages are slower and less efficient. My choice was between Pascal and C, and upon further inquiries I was advised that C may be too advanced for me to learn after knowing Basic. A form of Pascal from Borland International Inc. would be a much better transition from an unstructured language like Basic to a structured high level language. This language is called Turbo Pascal (which I will call Turbo). From everything I have heard since I chose Turbo, I made a very good choice. Turbo has more than 750,000 users worldwide, it won PC Magazine's award for technical excellence, it was named "Most Significant Product of the year" (1986) by PC Week and it is the fastest Pascal development environment on the planet. The only languages that are considered better than Turbo Pascal is Turbo C and Turbo Prolog, both of which are put out by Borland International. These are considered <u>very</u> high level languages; therefore, they are more difficult to learn.

Using Turbo Pascal to Program the Rational Method

This is the combination process of learning Turbo and writing the program, while researching and converting the charts and nomographs to usable formulas and processes.

After defining the problem and deciding on a viable solution, I started learning Turbo. I am one of those people that likes to learn by doing, so that is what I did. The first part of the program was simple enough to begin learning the fundamentals while writing the source code. It is fairly easy to write information or questions to the screen and retrieve an answer from the user, but as time went on I had to continue learning about the many commands, functions and procedures that were needed to achieve the desired outcome.

It was not very long before I had to start reducing charts and nomographs from the VESCH's chapter 5 to formulas and functions. These conversions took up most of the winter quarter. The first chart converted was plate 5-2 (see appendix F), which is the nomograph used to find time of concentration. This was accomplished with the assistance of Dr. Jerry Mann, a professor in statistics. It took him about three weeks to find formulas for time of concentration (TOC), pipe size, and velocity (see appendix F). The formulas he came up with are:

* $\text{TOC} = .0086938064 \text{ x } L^{1.140313996} / H^{.3790459807}$

Where 'L' is hydraulic length, or the distance from the most remote point in the drainage area to the drain inlet in feet. 'H' is the difference in elevation of the most remote point and the drain inlet in feet.

- Pipe size = 16.01513 x (n x Q)$^{.375}$ / S$^{.1875}$

  Where 'n' in the pipe roughness coefficient (see appendix G). 'Q' is the discharge of the area in cubic feet per second. And 'S' is the proposed slope of the pipe in percent.

- Velocity = 0.7148 x Q$^{.25}$ x S$^{.375}$ / n$^{.75}$

  Where 'Q' is the discharge in cubic feet per second. 'S' is the proposed slope of the pipe in percent. And 'n' is the pipe roughness coefficient (see appendix F).

While Dr. Mann was busy with those, I was trying to figure something out for the rainfall intensity charts (plates 5-3 through 5-7 in appendix F). I considered plotting the charts on the screen or printer, but accuracy would be minimal. Another thought was to interpolate the curves and put the data into separate files according to city and design storm. This method would take too much time and too much work. I knew there had to be a better and more accurate way. I asked Dr. Mann if he thought he could come up with a formula, but he declined.

While studying Hydrology by Meyers, I came across some formulas for rainfall intensities in the Boston area. Boston is close enough to Virginia to allow these formulas to be manipulated in such a way as to match the curves exactly. This was done by interpolating numbers off the charts and plugging them into the formulas. Then, using the trial and error method of equation solving, I finally developed the correct equation for each of the curves.

Throughout the entire year I kept running into problems dealing with Turbo and the conversion of the process from manual to computer assisted. Some of the problems I ran into were:

1. designing the questions, and procedures for asking the right questions at the right times in order to get the correct answers.
2. putting exponents into Turbo. This is a function not documented very well by Turbo. For three weeks I was stuck, until Andrew Roy enlightened me on the answer. It seems Turbo uses the mathematic principle,
   $A^B = 10^{(B \times \log A)}$, which in Turbo is written as,
   A^B = Exp(ln(A) x (B)).
3. how to re-run the program for multiple pipes. I choose to remedy this by putting a label at the beginning of the program, and using a goto statement where you have the choice of re-running the program.
4. how to check velocity in case the answer is technically too high or too low. This was simply done with an if - then statement.
5. how to save the appropriate information to a file and how to print the file (ie. pipe schedule). Without getting too much into programming, it is hard to explain the solution, so let us just say it took a lot of experimentation. Also, if the program is left before printing the file, it can be printed from the DOS prompt by typing 'copy pipe.dat prn' and pressing enter.

6. I had a hard time finding a suitable history of the Rational Method. The one taken from Hydrology by Chow is the best example found.

7. I had to choose between four different formulas for time of concentration. After studying and comparing them all, I decided on the one Dr. Mann came up with. Mainly because it allows you to multiply by a surface cover factor, that results in a more accurate answer.

8. I had major problems designing the program to calculate multiple pipes. It took a long time and a lot of trial and error. *But the added flexibility was worth it.*

Future Possible Additions

If I had five years to spend on writing programs for engineering calculations, I would have indeed done it. But, being limited to my one and only fifth year, I did not have the time to work on other problems.

In the future, these other areas of construction calculations can be addressed as additions to this program; retaining walls, open channel flow, decking, roadway alignment, surveying, water retention ponds, etc. These can be done as senior projects or just for the fun of it.

These sub-programs can be bound into a powerful construction calculations package. This would be easy to do by starting the process with a menu selection of which service the user needs. This package would be an asset to students, faculty and practitioners.

Comparison of Computer and Manual Techniques

Time for Tasks

I realized that in order to see if this program worked, it would have to be compared to the manual method. In comparing them I was checking on three main areas; speed, accuracy, good and bad points of each method. All three of these subjects are discussed in the following text.

The compiled data in minutes represent the time it takes to complete the methods for one area and one pipe. The times are probably faster than most people could do it, simply because I have been working with both methods intimately for the past year and know the process very well.

Tasks	Manual in minutes	Computer in minutes
Find acreage with planimeter	13:47	13:47
Runoff coefficient multiple two	2:49	1:20
Figure time of concentration	2:41	0:44
Figure intensity	0:42	0:07
Calculate Q	0:20	0:001
Size pipe	2:34	0:10
Makes pipe schedule	11:42	0:32
Make manhole schedule	3:45	0:35
Total	36:20	16:35

Time was checked with a stop watch for each of the steps shown in the previous chart. I will now explain the steps.

The acreage was found by using a planimeter. A planimeter is an instrument where the perimeter of an irregular area is traced in order to find the acreage. The time is the same because this must be done manually for both methods.

The runoff coefficient is found by choosing numbers off of table 5-2, (appendix F). For multiple conditions, a weighted *average must be done. Manually, you must have* a calculator that works, and go through the weighted average process, (see page V-4 in appendix F). On the computer, a weighted average is preformed automatically, taking half the time of the manual method.

Time of concentration, manually, is calculated from the nomograph on plate 5-2 in appendix F. This is easily calculated with the program by using the computerized method. Time is over three and a half times faster.

Rainfall intensity is determined from plates 5-3 through 5-7 in appendix F. These charts are fairly easy to use but are somewhat hard to work with due to loose papers and keeping up with them. Entering the bottom of the chart with the time of concentration, the lines are followed up to the correct design storm, and then a ninety degree turn is made over to the correct rainfall intensity. Using the computer, only two questions are asked, "What town are you near?" and "What year design storm are you designing for?". The computer method is over six times faster.

To calculate peak discharge (Q), you must multiply Acreage times runoff coefficient times rainfall intensity. This is accomplished in about a half of a second by the computer.

Pipesize and velocity is figured from the nomograph marked Drainage & Sewerage - Hydraulic Computation in appendix F. With Q, a slope, and a pipe roughness coefficient, one can read diameter and velocity. The computer asks what kind of pipe the user wishes to use and gives the user a selection. The program then asks the user "What percent slope do you want to try first?". Then, it automatically figures diameter and velocity. This is calculated with the computer, over fifteen times faster than manually.

If you want a pipe schedule and/or a manhole schedule printed on your printer so it can be stickybacked and placed on your plans, the computer does it automatically in about one minute. To do this manually it took me about fifteen minutes to draw it up.

The total times are how long it takes to calculate one area and one pipe. The totals do not seem extremely worthwhile here, (the computer being only twice as quick), but project this to an area with ten pipes and the user will save over two hours.

Accuracy

Accuracy is one of the hardest checks to measure. I will just discuss what could happen with the manual technique, that would not happen with the computer technique.

1. The charts can be read wrong.

2. The wrong charts can be read.

3. The right charts can be read right but the numbers copied wrong.

4. The calculators batteries might be low, giving the wrong answers.

5. When punching calculator keys, you might hit the wrong keys. (the computer lets you check the numbers before they are processed)

6. When copying numbers they can be transposed.

Good and Bad Points of Each Technique

Manual Technique

Good

1.) less expense ($10 book)
2.) very portable

Bad

1.) large book to keep track of
2.) many pages in large book to keep track of
3.) see accuracy section above
4.) charts are less accurate than formulas
5.) very stressful and hard to follow
6.) need to remember or relearn process

Computer Technique

Good

1.) all of the information needed is on one small disk
2.) very accurate
3.) less stressful
4.) faster
5.) no need to relearn process

Bad

1.) you must have access to a computer

Conclusions

From this report, one can tell the process I went through to complete this project. The final product is more advanced, (as far as features go), than I had originally planed. The features that were not planned were the pipe and manhole schedules, and the ability to calculate more than one pipe.

One can see there is a need for this type of program in the landscape architecture profession. With the increase of computer use in the office, it shouldn't be very difficult to locate a computer to use. Hopefully this program will be improved upon and added to in the coming years, either as a senior project or because peoples interest was sparked by the complexities and logical problem solving abilities that are used in designing the programs. Programming is similar to solving a puzzle. It is fun, as well as productive.

If you use Pipe, I hope it is beneficial to you.

References and Sources

Hydrology

Chow, Ven Te. Handbook of Applied Hydrology. CA: McGraw-Hill, 1964.

Leopold, L.B. and Dunne, Thomas. Water in Environmental Planning. CA: W.H. Freeman, 1978.

Meyers, Charles. Hydrology. NY: Foundation Press.

Munson, Albe E. Construction Design for Landscape Architects. CA: McGraw-Hill, 1974

Seelye, E.E. Design Databook for Civil Engineers. NY: Wiley, 1951.

Virginia Soil and Water Conservation Commission. Virginia Erosion and Sediment Control Handbook. VA: Virginia Soil and Water Conservation Commission, 1980.

Computers

Borland International. Turbo Pascal Reference Manual. CA, 1985.

Faulk, Ed. The Turbo Pascal Handbook. NC: Compute!, 1986.

Garrison, Paul. Turbo Pascal for Basic Programers. IN: Que Corp., 1985.

Jourdain, Robert. Turbo Pascal Express. NY: Prentice Hall Press, 1987.

Norton, Peter. Inside the IBM PC. New York: Prentice Hall Press, 1986.

Research and Education Association. Handbook and Guide for Comparing and Selecting Computer Languages.

Rugg, Tom and Feldman, Phil. Turbo Pascal Program Library. IN: Que Corp, 1986.

Swan, Tom. Mastering Turbo Pascal. New Jersey: Hayden Publishing Co, 1986.

Wood, Steve. Using Turbo Pascal. Berkeley, CA., McGraw-Hill, 1986.

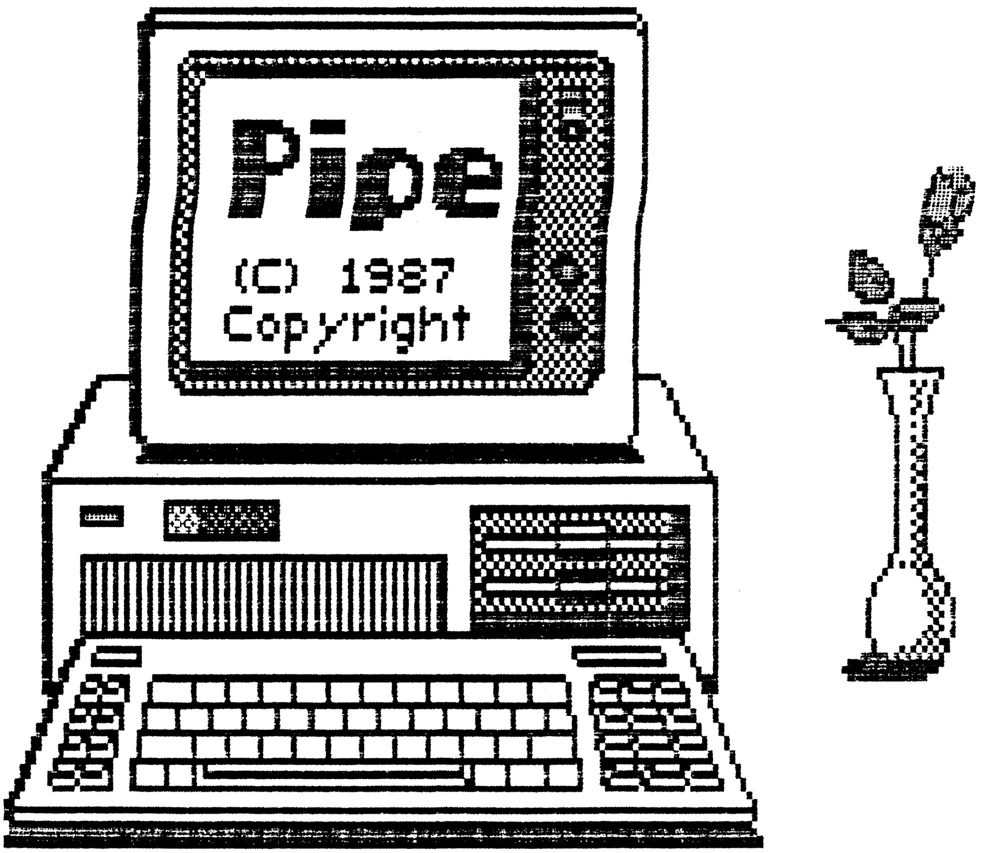
Discussions with

Dr. Jerry Mann	*Professor Statistics*	*961-4598*
Will Shepherd	Professor LAR at VPI	961-6683
Dean Bork	Professor LAR at VPI	961-5487
Mike Deeter	Professor LAR at University of Arizona	(602)-621-1004
Dr. Patrick Miller	Chairman LAR at VPI	961-5506
David Gayer	Senior LAR at VPI	951-4755
Don Law	*Retired Engineer*	*949-0318*
Andrew Roy	Programmer for AT&T	un-listed
Rob Bardsley	Computer Science Major at U.S.C.	803-886-6175

Appendix

A

Computer disk with: 1. the Pipe program
2. source code
3. support programs
4. document files



TO START PROGRAM

With computer off: place disk in drive A: and turn on computer.

With computer on: place disk in drive A: - the DOS prompt 'A>' should be on the screen, type 'pipe' and press the <Enter> key.

To view the text files: with disk in drive A: - type 'list <filename.ext>' (ie.. list pipe.doc)

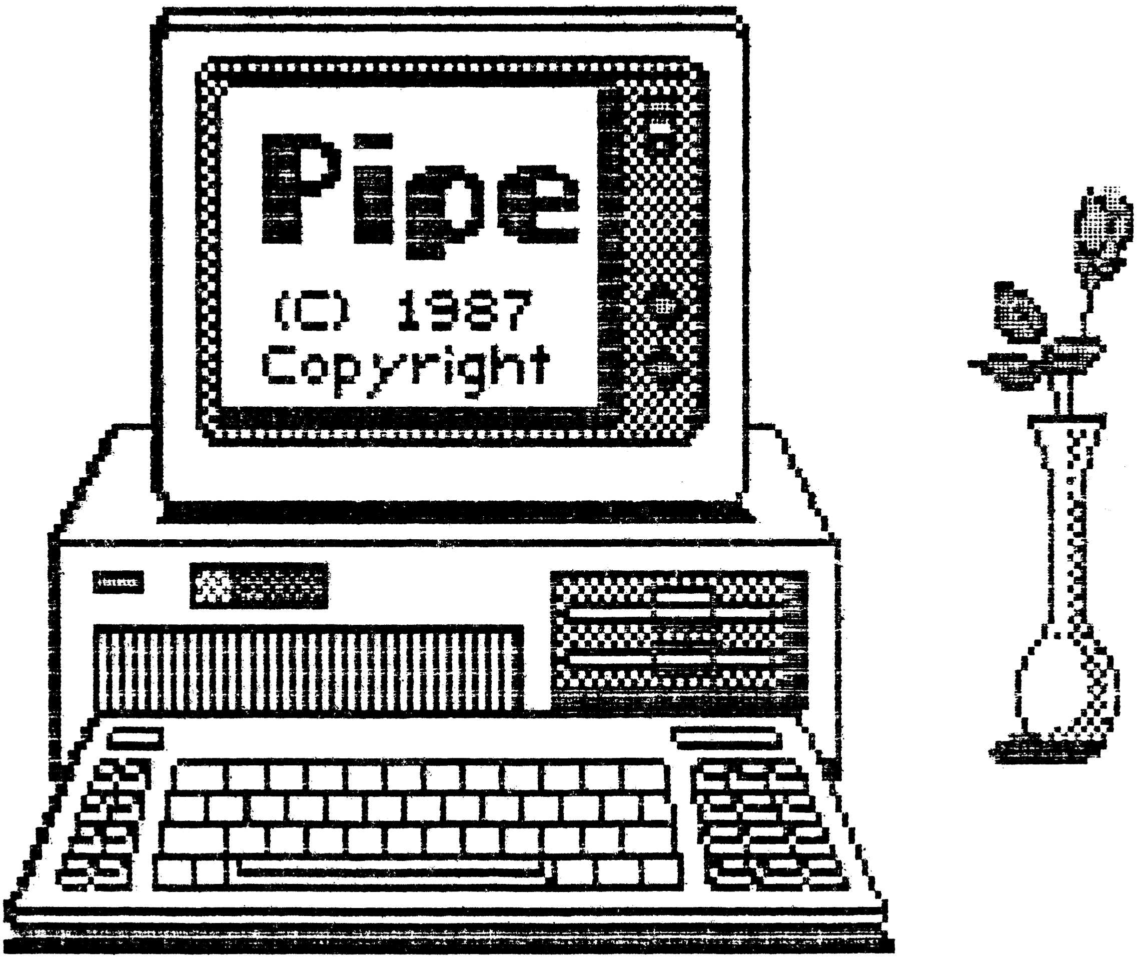
To copy the text files to the printer: with disk in drive A: - type 'copy <filename.ext> prn' (ie.. copy pipe.doc prn)

NOTE:
All text files have the .DOC extention.

Appendix

B

Printed Source Code



```
{*************************************************************************}
{*                             Pipe                                      *}
{*                       (C) Copywrite 1987                              *}
{*                       by:  Timothy Quillen                            *}
{*************************************************************************}


program pipe;


label  skip,rerun,reman,jump,apple,bannana,lava,storm;

var
  area,rough,slp,discharge,diameter,intensity,dist,slope,time,vel:real;
  qut,bigger,runnum,Q,YN,Y,N,big,choice,couldbe,userpick,usechoice:char;
  material,po,pi:array[1..20] of char;
  mh,cb,di:array[1..4] of char;
  cond,e,redo,x,w:integer;
  C: Array[1..15] of Real;  {Coeff}
  L: Array[1..15] of Real;  {Land areas}
  RunParm: Array[1..15] of Real;  {Land area times coeff}
  GotoPrinter: Boolean;
  MenuSelection,chinten : Char;
  Quit : Boolean;
  DataRight: Boolean;
  Coeff,invin,invout,slope2: Real;
  aa,ss:string[6];
  filevar,filevariable: text;
  peach,nmb:real;
  lgn:real;



{$I ibmtype.pas}
{$I dosvers.pas}
{$I numsrial.pas}
{$I numparll.pas}
{$I gameport.pas}




procedure machinesetup;
begin
writeln;
if machinetype = 1 then writeln('This is an IBM PC.');
if machinetype = 2 then writeln('This is an IBM PC XT or Portable PC.');
if machinetype = 3 then writeln('This is an IBM PCjr.');
if machinetype = 4 then writeln('This is an IBM PC AT.');
if machinetype = 5 then writeln('This is an IBM Convertible.');
if machinetype = 6 then writeln('This is not a true IBM.');
writeln;
write('The DOS version you are using is ');
```

```
if dosversion < 200 then write('not acceptable, you must use 2.00 or
later.');
if dosversion = 200 then write(' 2.0.');
if dosversion = 210 then write(' 2.1.');
if dosversion = 211 then write(' 2.11.');
if dosversion = 220 then write(' 2.2.');
if dosversion = 202 then write(' 2.02.');
if dosversion = 300 then write(' 3.0.');
if dosversion = 310 then write(' 3.1.');
if dosversion = 320 then write(' 3.2.');
if dosversion = 330 then write(' 3.3.');
if dosversion > 330 then write(' later than 3.3.');
writeln;
writeln;
writeln('Number of serial ports : ',numberserial);
writeln;
writeln('Number of parallel ports : ',numberparallel);
writeln;
write('There is ');
if gameport then
write('a')
else write('no');
writeln(' game port present');
writeln;
textcolor(11);
writeln('Press <Enter> to continue');
end;



procedure logo;
begin
    writeln;
    textcolor(1);
    writeln('           ██████▄   ██   ██████▄   ███████  (C)');
    writeln('           ██    ██          ██    ██  ██      ');
    textcolor(2);
    writeln('           ██    ██  ██   ██    ██  █████   ');
    writeln('           ██████▀   ██   ██████▀   ██      ');
    textcolor(6);
    writeln('           ██        ██   ██        ██      ');
    writeln('           ██        ██   ██        ███████');
    writeln;
end;



procedure credit;
begin
clrscr;
textcolor(15);
gotoxy(1,5);
```

```
writeln;
writeln('            ┌───────────┬────────────────────┬───────────────────┐');
writeln('            │           │  Pipe version 1.0  │                   │');
VRITELN('            │           └────────────────────┘                   │');
writeln('            │             (C) Copyright  1987                    │');
writeln('            │             All rights reserved                    │');
writeln('            │                                                    │');
writeln('            │       Programmer: Timothy R. Quillen               │');
writeln('            │                                                    │');
writeln('            │     For his fifth year senior project              │');
writeln('            │        in Landscape Architecture.                  │');
writeln('            │                                                    │');
writeln('            │ To see documentation type ''list pipe.doc''         │');
writeln('            │             at the A> prompt.                      │');
writeln('            │                                                    │');
writeln('            │ Please send any comments or recomendations         │');
writeln('            │          to:                                       │');
writeln('            │             Tim Quillen                            │');
writeln('            │             Rt.4 Box 317                           │');
writeln('            │             Waynesboro, VA. 22980                  │');
writeln('            │                                                    │');
writeln('            │  << Press the <Enter> key to continue >>           │');
writeln('            └────────────────────────────────────────────────────┘');
writeln;
read(kbd,qut);
end;

procedure The_End;
begin
  textcolor(11);
  clrscr;
  gotoXY(1,5);
  writeln('It has been nice working with you, I hope you have found pipe to
be');
  writeln('beneficial.  Have a nice day.');
  writeln;
  logo;
  halt;
end;

procedure Complete_Intro;
begin
clrscr;
textcolor(11);
writeln('     This program uses the Rational Method of storm water
calculations ');
writeln('for calculating peak runoff, in order to size pipe.  The origin of
this');
writeln('formula is somewhat obscure.  In America, the formula was first
mentioned');
WRITELN('in 1889 by Emil Kuichling to determine of peak runoff for sewer
design');
WRITELN('in Rochester, New York, during the period from 1877 to 1888.  In
England,');
```

```
WRITELN('the method using the rational formula is often referred to as the
Lloyd -');
WRITELN('Davis method.');
WRITELN('     The rational formula is Q = CIA, where Q is the peak
discharge in');
WRITELN('cubic feet per second, C is a runoff coefficient depending on
characteristics');
WRITELN('of the drainage basin, I is the rainfall intensity in inches per
hour,');
WRITELN('and A is the drainage area in acres.');
WRITELN('     According to D.B. Krimgold, the assumptions involved in the
rational');
WRITELN('formula are:');
WRITELN;
WRITELN('(1)  The rate of runoff resulting from any rainfall intensity is a
maximum');
WRITELN('     when the rainfall intensity lasts as long or longer than the
time');
WRITELN('     of consentration; that is, the largest quantity of water
possible is');
writeln('     on the site is when the rainfall intensity is greater than or
equal to the');
writeln('     time of consentration.');
WRITELN;
WRITELN('(2)  The maximum runoff resulting from a rainfall intensity, with
a duration');
WRITELN('     equal to or greater than the time of concentration, is a
simple fraction');
WRITELN('     of such rainfall intensity; that is, it assumes a straight
line relation');
WRITELN('     between Q and I, and Q = 0 when I = 0.');
textcolor(5);
WRITE('Press the <Enter> key to contiue.');
readln(qut);
clrscr;
gotoxy(1,5);
textcolor(11);
WRITELN('(3)  The frequency of peak discharge is the same as that of the
rainfall');
WRITELN('     intensity for the given time of concentration; that is, the
same design');
writeln('     is used for both.');
WRITELN;
WRITELN('(4)  The relationship between peak discharge and size of discharge
area');
WRITELN('     is the same as the relationship between duration and
intensity of rainfall.');
WRITELN('(5)  The coefficient of runoff is the same for storms of various
frequencies.');
writeln('     that is, the coefficient has to do with the land cover, not
storms.');
WRITELN;
WRITELN('(6)  The coefficient of runoff is the same for all storms on a
given watershed.');
```

```
WRITELN;
textcolor(5);
WRITELN('Press the <Enter> key to continue.');
read(qut);
textcolor(11);
end;

procedure Intro;
begin
  clrscr;
  gotoXY(2,10);
  writeln;
  textcolor(9);
  writeln('This program is to determine the pipe sizes needed in a given
drainage');
  writeln('area, using the Rational Method, Q=CIA.');
  writeln;
  textcolor(15);
  writeln('Notes:');
  textcolor(11);
  writeln;
  writeln('You may want to consider using water as a natural amenity rather
than');
  writeln('piping it underground?  For instance, you can use an open
channel of');
  writeln('various forms to get water from one place to another.  or you
might');
  writeln('consider using ponding to collect water for recreational use.
There are');
  writeln('many ways to avoid the expense of running a pipe, but if you
have no');
  writeln('other option, then we can continue this program, but remember to
think ');
  writeln('about other opportunities.');
  writeln;
  textcolor(3);
  write('Press any key to continue or Q to quit.');
  read(kbd,qut);
  if upcase(qut) = 'Q' then
  the_end
else
end;


procedure Acreage(var area : real);
var
e:integer;
aa:string[5];
begin
    clrscr;
    gotoXY(2,10);
    textcolor(2);
    write('What is the drainage area acreage ?  ');
    repeat
    readln(aa);
    val(aa,area,e);
  until e = 0;
  writeln;
 textcolor(5);
    writeln('To check, is this the correct acerage? ',area:3:2,
(Y,N)');
    repeat
    read(kbd,YN);
  until upcase(YN) in ['Y','N'];
  case upcase(YN) of
    'N' : acreage(area);
    'Y' : if area > 200 then
          begin
            writeln;
            sound(100);
            delay(500);
            sound(800);
            delay(500);
            sound(95);
            delay(500);
            sound(750);
            delay(1000);
            nosound;
            textcolor(12);
            writeln(' The area you are working with is too large for the
Rational Method');
            writeln(' to get reasonably accurate answers for you.  If you
wish to continue');
            writeln(' press the >Return< key, But remember your data may be
faulty over an');
            writeln(' acceptable margin.  A solution is to divide the
drainage area into several');
            writeln(' different smaller basins and re-enter the acreage, by
pressing the A key.');
            writeln(' Or press the Q key to quit.  ');
            read(kbd,qut);
            writeln;
            if upcase(qut) = 'Q' then
              The_End;
            if upcase(qut) = 'A' then
              acreage(area);
          end;
  end;
  textcolor(11);
end;


procedure coefficient_chart(Var ToPrint: Boolean);
begin
  repeat
   clrscr;
   gotoXY(2,10);
   writeln;
   textcolor(9);
   writeln('Do you want the coefficient chart to be printed on your
printer? (Y,N)');
   textcolor(11);
   writeln('(if not, it will appear on your screen)');
   read(kbd,YN);
 until upcase(YN) in ['Y','N'];
 case upcase(YN) of
   'N' : ToPrint := False;
   'Y' : ToPrint := True
 End
end;

procedure Note;
BEgin
  clrscr;
  writeln;
  textcolor(15 + blink);
  writeln('NOTE:');
  textcolor(11);
  writeln;
  writeln('The designer must use judgement to select the appropriate C
value');
  writeln('within the range on the following chart.  Generally, areas
with');
  writeln('permeable soils, flat slopes and dense vegetation should have
the');
  writeln('lowest C values.  Areas with dense soils, moderate to steep
slopes,');
  writeln('and sparce vegetation should be assigned the highest C
values.');
  writeln;
  textcolor(3);
  writeln('Press any key to continue.');
  read(kbd,qut);
  begin;
    clrscr;
    writeln('Land use                            C           Land use
      C');
    writeln('--------------------------------------
---------------------------------');
    writeln('*    BUSINESS:                                  - Drives and Walks
   .75-.85');
    writeln('  downtown areas         .70-.95           - Roofs
   .75-.95');
    writeln('  neighborhood areas     .50-.70           *   LAWNS:
         ');
    writeln('*   RESIDENTIAL:                                 sandy soil, flat,
0-2%  .05-.10');
    writeln('  single family          .30-.50             sandy, average, 2-7%

  .10-.15');
    writeln('  multi units, detached  .40-.60             sandy, steep,  > 7%
   .15-.20');
    writeln('  multi units, attached  .60-.75             heavy soil, flat,
0-2%  .13-.17');
    writeln('  suburban                    .25-.40          heavy, average, 2-7%

  .18-.22');
    writeln('*   INDUSTRIAL:                                heavy, steep,  > 7%
   .25-.35');
    writeln('  light areas                 .50-.80        *   AGRICULTURAL LAND:

          ');
    writeln('  heavy areas                 .60-.90          Bare packed soil
          ');
    writeln('- Parks, Cemeteries           .10-.25            smooth
   .30-.60');
    writeln('- Playgrounds                 .20-.35            rough
   .20-.50');
    writeln('- Railroad Yard Areas         .20-.40          Cultivated rows
          ');
    writeln('- Unimproved Areas            .10-.30            heavy soil no crop

  .30-.60');
    writeln('- Woodlands                   .05-.25            heavy soil with
crop  .20-.50');
    writeln('*   STREETS:                                     sandy soil no crop

  .20-.40');
    writeln('  asphalt                     .70-.95            sandy soil with
crop  .10-.25');
    writeln('  concrete                    .80-.95          Pasture
          ');
    writeln('  brick                       .70-.85            heavy soil
   .15-.45');
    writeln('                                                 sandy soil
   .05-.25');
    textcolor(4);
    writeln('********* COPY APPROPRIATE AMOUNTS AND PRESS ENTER TO CONTINUE
*********');
    Repeat
    Begin
      Read(kbd,qut);
    End;
    Until True;
    textcolor(15);
  End
end;


procedure printnote;
Begin
  writeln(lst,'Land use                          C          Land use
      C');
  writeln(lst,'-----------------------------------
------------------------------');
  writeln(lst,'*   BUSINESS:                              - Drives and Walks
   .75-.85');
  writeln(lst,'  downtown areas              .70-.95     - Roofs
   .75-.95');
  writeln(lst,'  neighborhood areas          .50-.70     *   LAWNS:
          ');
  writeln(lst,'*   RESIDENTIAL:                               sandy soil, flat,
0-2% .05-.10');
  writeln(lst,'  single family               .30-.50       sandy, average 2-7%

  .10-.15');
  writeln(lst,'  multi units, detached       .40-.60       sandy, steep,  > 7%

  .15-.20');
  writeln(lst,'  multi units, attached       .60-.75       heavy soil, flat,
0-2% .13-.17');
  writeln(lst,'  suburban                    .25-.40       heavy, average,
2-7%   .18-.22');
  writeln(lst,'*   INDUSTRIAL:                               heavy, steep,  > 7%

  .25-.35');
  writeln(lst,'  light areas                 .50-.80     *   AGRICULTURAL
LAND:          ');
  writeln(lst,'  heavy areas                 .60-.90       Bare packed soil
        ');
  writeln(lst,'- Parks, Cemeteries           .10-.25         smooth
   .30-.60');
  writeln(lst,'- Playgrounds                 .20-.35         rough
   .20-.50');
  writeln(lst,'- Railroad yard areas         .20-.40       Cultivated Rows
        ');
  writeln(lst,'- Unimproved Areas            .10-.30         heavy soil no
crop   .30-.60');
  writeln(lst,'*   STREETS:                                    heavy soil with
crop .20-.50');
  writeln(lst,'  asphalt                     .70-.95         sandy soil no
crop   .20-.40');
  writeln(lst,'  concrete                    .80-.95         sandy soil with
crop .10-.25');
  writeln(lst,'  brick                       .70-.85       Pasture
        ');
  writeln(lst,'- Woodlands                   .05-.25         heavy soil
   .15-.45');
  writeln(lst,'                                              sandy soil
   .05-.25');
  writeln(lst);
  writeln(lst,'NOTE:');
  writeln(lst,'     The designer must use judgement to select the
appropriate C value');
  writeln(lst,'within the range.  Generally, areas with permeable soils,
flat');
  writeln(lst,'slopes and dense vegetation should have the lowest C
values.');
  writeln(lst,'Areas with dense soils, moderate to steep slopes, and
sparce');
  writeln(lst,'vegetation should be assigned the highest C values.');
  writeln(lst,'');
end;

procedure Chart_Choice(Var UserChoice: Char);
Begin
clrscr;
gotoXY(1,5);
```

```
  textcolor(11);
  writeln('IF THE LAND USE AND SOIL COVER IS HOMOGENEOUS OVER THE ENTIRE
DRAINAGE AREA');
  writeln('THEN A (C) VALUE CAN BE DETERMINED DIRECTLY FROM THE PREVIOUS
CHART.');
  writeln;
  writeln('IF THERE ARE MULTIPLE SOIL COVER CONDITIONS, A WEIGHTED AVERAGE
MUST BE DONE.');
  writeln;
  textcolor(4);
  writeln('Choose One:');
   textcolor(9);
  writeln('<1> HOMOGENEOUS');
  writeln('<2> MULTIPLE CONDITIONS');
  writeln('<3> SEE CHART AGAIN');
  writeln('<4> QUIT');
  textcolor(15);
  writeln;
  write('CHOICE?');
  Repeat
    Read(kbd, UserChoice);
  Until (UserChoice in ['1','2','3','4'])
end;

procedure Homo;
begin
repeat
  clrscr;
  gotoXY(2,10);
  textcolor(5);
  write('What is the runoff coefficient ?  ');
  readln(aa);
  val(aa,coeff,e);
  until e = 0;
  writeln;
  textcolor(13);
    writeln('To check, is this the correct coefficient? ',coeff:1:4,'
(Y,N)');
  repeat
    read(kbd,YN);
  until upcase(YN) in ['Y','N'];
  case upcase(YN) of
    'N' : homo;
  end;
  textcolor(11);
end;

procedure Mult;
Var
  Pointer: Integer;
begin;
  clrscr;
  gotoXY(1,10);
  textcolor(13);
  writeln('For this section, enter the AREA of the Land use and press
Enter,');
  writeln('then enter the corresponding (C) value and press Enter again.');
  writeln;
  textcolor(11);
  writeln('Upto 15 Land uses are allowed, if more than 15, combine the
areas');
  writeln('of the ones with the same (C) value.');
  textcolor(12);
  writeln;
  writeln('Please be sure the combined areas equal the total area you
stated earlier.');
  writeln;
  textcolor(11);
  For Pointer := 1 to 15 Do
  Begin
    C[Pointer] := 0;
    L[Pointer] := 0
  End;
  Pointer := 1;
Repeat
  BEgin
  textcolor(10);
    write('What is the area of land use # ',Pointer:1,' ? ');
    readln(L[Pointer]);
    Write('And corresponding (C) value ? ');
    readln(C[Pointer]);
    writeln;
    Pointer := Pointer + 1;
    If Pointer > 2 Then
    Begin
      repeat
  textcolor(4);
        writeln('Are there any more?  (Y,N)');
        read(kbd,YN);
      until upcase(YN) in ['Y','N'];
    End
  End;
  Until ((Pointer > 2) AND (Upcase(YN) = 'N')) OR (Pointer > 15);
  textcolor(4);
  writeln('Thank you.');
  textcolor(15);
  writeln
end;


procedure Method(Var Correct: Boolean);
VAr
  Loop: Integer;
begin
  clrscr;
  gotoXY(2,10);
  repeat
    writeln;
    textcolor(4);
    writeln('Are these all correct?  (Y,N)');
    writeln;
    writeln(' Area     (C)');
    writeln(' ----     ---');
    textcolor(11);
  writeln;
  For Loop := 1 to 15 Do
    Writeln(L[Loop]:5:2,'   ',C[Loop]:5:2);
  textcolor(15);
  writeln;
  read(kbd,YN);
  until upcase(YN) in ['Y','N'];
  case upcase(YN) of
    'N' : Correct := False;
    'Y' : Correct := True
  End;
end;


procedure Rational(Var Coeff: Real);
VAr
  TotalA, TotalB: real;
  Loop: Integer;
  correct: boolean;
begin
  TotalA := 0;
  TotalB := 0;
  For Loop := 1 to 15 Do
begin
    TotalA := TotalA + L[Loop];
    RunParm[Loop] := L[Loop] * C[Loop];
    TotalB := TotalB + RunParm[Loop];
    coeff := TotalB/TotalA;
  end;
    textcolor(9);
    writeln;
    writeln('The multiple condition runoff coefficient is ',coeff:1:3,'.');
    textcolor(11);
    writeln;
    write('Press <Enter> to continue ');
    read(kbd,qut);
    writeln;
  End;


procedure TOC(var time : real);
label  redo;
begin
redo:
repeat
repeat
clrscr;
gotoXY(1,10);
```

```
textcolor(6);
writeln('Now we need to determine the Time Of Concentration (TOC).');
writeln('TOC is the time of flow from the most remote point in the');
writeln('drainage area to the drain inlet, in minutes.');
writeln;
writeln;
textcolor(11);
writeln('What is the distance in feet, from the most remote point in the
');
write('drainage area to the drain inlet?   ');
    read(aa);
    val(aa,dist,e);
   until e = 0;
if aa = '' then goto redo;
writeln;
writeln;
writeln('What is the drop, in feet, from the most remote point, to the
drain inlet?   ');
read(aa);
val(aa,slope,e);
until e = 0;
if aa = '' then goto redo;
time := 0.0086938064 *
exp(ln(dist)*(1.140313996))/exp(ln(slope)*(0.3790459807));
writeln;
writeln;
textcolor(9);
writeln('What kind of overland flow surface is there? ');
writeln('Choose the most common one.  If they are all of similar
quantity');
writeln('then choose selection <3>.');
writeln;
writeln('<1>  Bare Earth');
writeln('<2>  Grass');
writeln('<3>  Concrete or Asphalt');
writeln;
writeln('Choice:  ');
repeat
read(kbd,choice);
until (choice in ['1','2','3']);
case choice of
  '1' : time := time;
  '2' : time := time * 2;
  '3' : time := time * 0.4;
end;
writeln;
writeln('What kind of channel flow surface is there? ');
writeln;
writeln('<1>  Grass');
writeln('<2>  Concrete');
writeln('<3>  Stream Flow');
writeln('<4>  None');
writeln;
write('Choice:  ');
```

```
repeat
read(kbd,choice);
until (choice in ['1','2','3','4','5']);
case choice of
  '1' : time := time;
  '2' : time := time * 0.2;
  '3' : time := time;
  '4' : time := time;
end;
if time < 5 then time := 5;
writeln;
writeln;
textcolor(13);
writeln('Time of Concentration is ',time:4:3,' minutes.');
writeln;
textcolor(14);
writeln('Press Enter to continue.');
read(kbd,qut);
  End;

procedure piperough(var  rough:real);
begin
clrscr;
gotoxy(2,10);
textcolor(11);
writeln('What kind of pipe do you want to use?  ');
writeln;
textcolor(9);
WRITELN('If the one you want is not listed, choose the one of similar
texture.');
WRITELN;
writeln('<1>  CONCRETE - 0.013');
writeln('<2>  CAST-IRON - 0.013');
writeln('<3>  CORRUGATED STEEL - 0.015');
writeln('<4>  VITRIFIED CLAY - 0.012');
writeln('<5>  PLASTIC - PVC - 0.010');
writeln;
textcolor(15);
write('CHOICE:  ');
repeat
read(kbd,choice);
until (choice in ['1','2','3','4','5']);
case choice of
  '1' : rough := 0.013;
  '2' : rough := 0.013;
  '3' : rough := 0.015;
  '4' : rough := 0.012;
  '5' : rough := 0.010;
end;
writeln;
END;

procedure year2(var intensity : real);
begin
```

```
clrscr;
gotoxy(1,10);
textcolor(2);
writeln('For a 2 year storm.');
writeln;
writeln('Which city are you closest to?');
writeln;
textcolor(9);
writeln('<1>  Lynchburg - Roanoke - Harrisonburg - Winchester');
writeln('<2>  Norfolk - Virginia Beach');
writeln('<3>  Richmond - Petersburg - Arlington - Charlottesville');
writeln('<4>  Wytheville - Blacksburg - Marion');
textcolor(3);
writeln;
write('Choice? ');
repeat
read(kbd,choice);
until (choice in ['1','2','3','4']);
case choice of
 '1' : intensity := (141 / (time + 25));
 '2' : intensity := (141 / (time + 19));
 '3' : intensity := (141 / (time + 20));
 '4' : intensity := (105 / (time + 15));
end;
writeln;
writeln;
textcolor(14);
writeln('The correct intensity for the design storm and location');
writeln('you choose is ',intensity:4:4,' inches per hour.');
writeln;
textcolor(6);
writeln('Press Enter to continue.');
read(kbd,qut);
end;

procedure year5(var intensity : real);
begin
clrscr; gotoxy(1,10);
textcolor(4);
writeln('For a 5 year storm.');
writeln;
writeln('Which city are you closest to?');
writeln;
writeln('<1>  Lynchburg - Roanoke - Harrisonburg - Winchester');
writeln('<2>  Norfolk - Virginia Beach');
writeln('<3>  Richmond - Petersburg - Arlington - Charlottesville');
writeln('<4>  Wytheville - Blacksburg - Marion');
writeln;
textcolor(3);
write('Choice? ');
repeat
read(kbd,choice);
until (choice in ['1','2','3','4']);
case choice of
```

```
 '1' : intensity := (163 / (time + 20));
 '2' : intensity := (160 / (time + 20));
 '3' : intensity := (160 / (time + 19));
 '4' : intensity := (163 / (time + 22));
end;
writeln;
writeln;
writeln('The correct intensity for the design storm and location');
writeln('you choose is ',intensity:4:4,' inches per hour.');
writeln;
writeln('Press Enter to continue.');
read(kbd,qut);
end;

procedure year10(var intensity : real);
begin
clrscr; gotoxy(1,10);
textcolor(5);
writeln('For a 10 year storm.');
writeln;
writeln('Which city are you closest to?');
writeln;
textcolor(11);
writeln('<1>  Lynchburg - Roanoke - Harrisonburg - Winchester');
writeln('<2>  Norfolk - Virginia Beach');
writeln('<3>  Richmond - Petersburg - Arlington - Charlottesville');
writeln('<4>  Wytheville - Blacksburg - Marion');
writeln;
textcolor(3);
write('Choice? ');
repeat
read(kbd,choice);
until (choice in ['1','2','3','4']);
case choice of
 '1' : intensity := (191 / (time + 22));
 '2' : intensity := (191 / (time + 22));
 '3' : intensity := (191 / (time + 22));
 '4' : intensity := (191 / (time + 23));
end;
writeln;
writeln;
textcolor(10);
writeln('The correct intensity for the design storm and location');
writeln('you choose is ',intensity:4:4,' inches per hour.');
writeln;
writeln('Press Enter to continue.');
read(kbd,qut);
end;

procedure year25(var intensity : real);
begin
clrscr; gotoxy(1,10);
textcolor(14);
writeln('For a 25 year storm.');
```

```
writeln;
writeln('Which city are you closest to?');
writeln;
textcolor(3);
writeln('<1>  Lynchburg - Roanoke - Harrisonburg - Winchester');
writeln('<2>  Norfolk - Virginia Beach');
writeln('<3>  Richmond - Petersburg - Arlington - Charlottesville');
writeln('<4>  Wytheville - Blacksburg - Marion');
writeln;
textcolor(12+blink);
write('Choice? ');
textcolor(11);
repeat
read(kbd,choice);
until (choice in ['1','2','3','4']);
case choice of
 '1' : intensity := (233 / (time + 24));
 '2' : intensity := (233 / (time + 24));
 '3' : intensity := (233 / (time + 24));
 '4' : intensity := (233 / (time + 25));
end;
writeln;
writeln;
writeln('The correct intensity for the design storm and location');
writeln('you choose is ',intensity:4:4,' inches per hour.');
writeln;
writeln('Press Enter to continue.');
read(kbd,qut);
end;

procedure year50(var intensity : real);
begin
clrscr; gotoxy(1,10);
textcolor(4);
writeln('For a 50 year storm.');
writeln;
writeln('Which city are you closest to?');
writeln;
writeln('<1>  Lynchburg - Roanoke - Harrisonburg - Winchester');
writeln('<2>  Norfolk - Virginia Beach');
writeln('<3>  Richmond - Petersburg - Arlington - Charlottesville');
writeln('<4>  Wytheville - Blacksburg - Marion');
writeln;
textcolor(3);
write('Choice? ');
repeat
read(kbd,choice);
until (choice in ['1','2','3','4']);
case choice of
 '1' : intensity := (255 / (time + 23));
 '2' : intensity := (252 / (time + 23));
 '3' : intensity := (252 / (time + 23));
 '4' : intensity := (242 / (time + 23));
end;
```

```
writeln;
writeln;
textcolor(13);
writeln('The correct intensity for the design storm and location');
writeln('you choose is ',intensity:4:4,' inches per hour.');
writeln;
writeln('Press Enter to continue.');
read(kbd,qut);
end;

procedure year100(var intensity : real);
begin
clrscr; gotoxy(1,10);
textcolor(14);
writeln('For a 100 year storm.');
writeln;
writeln('Which city are you closest to?');
writeln;
textcolor(12);
writeln('<1>  Lynchburg - Roanoke - Harrisonburg - Winchester');
writeln('<2>  Norfolk - Virginia Beach');
writeln('<3>  Richmond - Petersburg - Arlington - Charlottesville');
writeln('<4>  Wytheville - Blacksburg - Marion');
writeln;
textcolor(3);
write('Choice? ');
repeat
read(kbd,choice);
until (choice in ['1','2','3','4']);
case choice of
 '1' : intensity := (275 / (time + 23));
 '2' : intensity := (333 / (time + 30));
 '3' : intensity := (299 / (time + 24));
 '4' : intensity := (299 / (time + 26));
end;
writeln;
writeln;
writeln('The correct intensity for the design storm and location');
writeln('you choose is ',intensity:4:4,' inches per hour.');
writeln;
textcolor(2);
writeln('Press Enter to continue.');
read(kbd,qut);
end;

procedure qdischarge(var discharge: real);
begin
  discharge := Area * Coeff * Intensity;
end;

procedure pipesize(var diameter: real);
begin
diameter := 16.01513 * Exp(Ln(rough * discharge)*(0.375))/exp(ln(slp) *
(0.1875));
```

```
if diameter < 6 then diameter := 6;
if (diameter > 6) and (diameter < 8) then diameter := 8;
if (diameter > 8) and (diameter < 10) then diameter := 10;
if (diameter > 10) and (diameter < 12) then diameter := 12;
if (diameter > 12) and (diameter < 15) then diameter := 15;
if (diameter > 15) and (diameter < 18) then diameter := 18;
if (diameter > 18) and (diameter < 21) then diameter := 21;
if (diameter > 21) and (diameter < 24) then diameter := 24;
if (diameter > 24) and (diameter < 27) then diameter := 27;
if (diameter > 27) and (diameter < 30) then diameter := 30;
if (diameter > 30) and (diameter < 33) then diameter := 33;
if (diameter > 33) and (diameter < 36) then diameter := 36;
if (diameter > 36) and (diameter < 39) then diameter := 39;
if (diameter > 39) and (diameter < 42) then diameter := 42;
if (diameter > 42) and (diameter < 48) then diameter := 48;
if (diameter > 48) and (diameter < 54) then diameter := 54;
if (diameter > 54) and (diameter < 60) then diameter := 60;
if (diameter > 60) and (diameter < 66) then diameter := 66;
if (diameter > 66) and (diameter < 72) then diameter := 72;
if (diameter > 72) and (diameter < 78) then diameter := 78;
if (diameter > 78) and (diameter < 84) then diameter := 84;
if (diameter > 84) and (diameter < 90) then diameter := 90;
if (diameter > 90) and (diameter < 96) then diameter := 96;
if (diameter > 96) and (diameter < 108) then diameter := 108;
if (diameter > 108) and (diameter < 120) then diameter := 120;
end;

procedure velosity(var vel: real);
label next,dog;
begin
vel := 0.7148 * exp(ln(discharge)*(0.25)) *
exp(ln(slp)*(0.375))/exp(ln(rough)*(0.75));
writeln;
if (vel < 2.5) or (vel > 9.0) then
begin
    textcolor(12 + blink);
    writeln('WARNING:');
    textcolor(10);
    writeln;
    writeln('The velocity is ',vel:2:2,'.  If the velocity is less than 2.5
the pipe');
    writeln('may silt up.  If the velocity is greater than 9.0 the pipe may
be corroded.');
    writeln('You can either continue or enter a new slope for the pipe.
');
    writeln;
    textcolor(11);
    write('Press Enter to continue or C to change slope.');
    read(kbd,qut);
    if qut = 'C' then goto next;
    if qut = 'c' then goto next;
end;
redo := 2;
goto dog;
```

```
next:
redo := 1;
dog:
end;

procedure aretheseok(var couldbe: char);
begin
repeat
clrscr;
gotoxy(1,5);
writeln;
textcolor(13);
    writeln('The answers are :');
    textcolor(11);
    writeln;
    writeln('                 area =  ',area:3:2,' acres');
    writeln('                slope =  ',slp:1:4,'.');
    writeln('             velocity =  ',vel:2:2,' feet per second');
    writeln('             pipesize =  ',diameter:3:0,' inches');
    writeln('       Q or discharge =  ',discharge:4:2,' cubic feet per
second');
    writeln('   runoff coefficient =  ',coeff:1:4,'.');
    writeln('time of concentration =  ',time:3:2,' minutes');
    writeln('    rainfall intesity =  ',intensity:2:2,' inches per hour');
    writeln('pipe roughness coeff. =  ',rough:1:4,'.');
    writeln;
    textcolor(13);
    writeln('Are all of these acceptable?  (Y,N)');
    writeln('(if not then everything must be re-entered)');
    read(kbd,couldbe);
  until upcase(couldbe) in ['Y','N'];
  textcolor(11);
end;

procedure savefile;
begin
assign(filevar,'pipe.dat');
rewrite(filevar);
writeln(filevar,'Pipe  Diameter  Distance  Type or material     Slope
Invert in  Invert out');
writeln(filevar,'no.   in inches in feet   of pipe
elevation  elevation');
writeln(filevar,'----  --------  --------  -------------------  ------
---------  ----------');
close(filevar);
end;

procedure additions;
begin
append(filevar);
write(filevar,'',nmb:2:0,'     ',diameter:3:0,'       ',lgn:4:2,'
',material,' ',slp:1:4,'  ');
write(filevar,''      ,invin:4:2,'      ',invout:4:2,'');
writeln(filevar);
```

```
close(filevar);
end;

procedure printfile;
const
scratch : string[128] = 'Pipe.dat';
begin
assign(filevar,scratch);
{$I-} reset(filevar); {I+}
while not Eof(filevar) do
repeat
readln(filevar,scratch);
writeln(lst,scratch);
until scratch = '';
writeln(lst);
close(filevar);
end;

procedure printmanhole;
const
junk : string[128] = 'manhole.dat';
begin
assign(filevariable,junk);
{$I-} reset(filevariable); {$I+}
while not Eof(filevariable) do
repeat
readln(filevariable,junk);
writeln(lst,junk);
until junk = '';
writeln(lst);
close(filevariable);
end;

procedure setupmanhole;
label  manhole,catch,drain,finish,menu,premh,precb,predi;
begin
assign(filevariable,'manhole.dat');
rewrite(filevariable);
writeln(filevariable,'---------------------------------------------------------------
----------');
close(filevariable);
menu:
begin
clrscr;
gotoxy(1,10);
writeln('Which of the following is being used?');
writeln;
writeln('Choose one: ');
writeln('<1> Manhole');
writeln('<2> Catch Basin');
writeln('<3> Drain Inlet');
writeln('<4> Quit Program');
writeln;
write('Choice: ');
```

```
repeat
read(kbd,choice);
until (choice in ['1','2','3','4']);
case choice of
  '1' : goto premh;
  '2' : goto precb;
  '3' : goto predi;
  '4' : the_end;
end;
end;
premh:
begin
append(filevariable);
writeln(filevariable);
writeln(filevariable,'Manhole      Pipes in                Pipes out');
writeln(filevariable,'number       numbers                 numbers');
writeln(filevariable,'-----------  --------------------
--------------------');
close(filevariable);
end;
manhole:
clrscr;gotoxy(1,10);
write('What number Manhole is this?  ');
read(mh);
writeln;
writeln('What is the numbers of the pipes coming into the Manhole?  ');
writeln('(seperate pipe numbers with comma''s)');
read(pi);
writeln;
writeln('What is the numbers of the pipes exiting the manhole?  ');
writeln('(seperate pipe numbers with comma''s)');
read(po);
writeln;
writeln;
writeln('This is manhole number ',mh,'');
writeln('The numbers of the pipes intering are ',pi,'');
writeln('The numbers of the pipes exiting are ',po,'');
writeln;
writeln('Are these all correct?  (Y,N)');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
case upcase(YN) of
  'N' : goto manhole;
end;
begin
append(filevariable);
writeln(filevariable,'',mh,'          ',pi,'   ',po,'');
close(filevariable);
end;
writeln;
writeln('Are there any more manholes to do?  (Y,N)');
repeat
read(kbd,YN);
```

```
 until upcase(YN) in ['Y','N'];
 case upcase(YN) of
   'Y' : goto manhole;
 end;
 writeln;
 writeln('Do you want to go to the menu for other selections?  (Y,N)');
 repeat
 read(kbd,YN);
 until upcase(YN) in ['Y','N'];
 case upcase(YN) of
   'Y' : goto menu;
   'N' : goto finish;
 end;
 precb:
 begin
 append(filevariable);
 writeln(filevariable);
 writeln(filevariable,'Catch basin  Pipes in                    Pipes out');
 writeln(filevariable,'number       numbers                    numbers  ');
 writeln(filevariable,'-----------  --------------------
--------------------');
 close(filevariable);
 end;
catch:
clrscr;gotoxy(1,10);
write('What number Catch basin is this?  ');
read(cb);
writeln;
writeln('What is the numbers of the pipes coming into the Catch basin?  ')
writeln('(seperate pipe numbers with comma''s)');
read(pi);
writeln;
writeln('What is the numbers of the pipes exiting the Catch basin?  ');
writeln('(seperate pipe numbers with comma''s)');
read(po);
writeln;
writeln;
writeln('This is catch basin number ',cb,'');
writeln('The numbers of the pipes intering are ',pi,'');
writeln('The numbers of the pipes exiting are ',po,'');
writeln;
writeln('Are these all correct?  (Y,N)');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
case upcase(YN) of
  'N' : goto catch;
end;
begin
append(filevariable);
writeln(filevariable,'',cb,'         ',pi,'  ',po,'');
close(filevariable);
end;
writeln;
```

```
writeln('Are there any more catch basins to do?  (Y,N)');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
case upcase(YN) of
  'Y' : goto catch;
end;
writeln;
writeln('Do you want to go to the menu for other selections?  (Y,N)');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
case upcase(YN) of
  'Y' : goto menu;
  'N' : goto finish;
end;
predi:
begin
append(filevariable);
writeln(filevariable);
writeln(filevariable,'Drain inlet  Pipes in           Pipes out');
writeln(filevariable,'number       numbers            numbers  ');
writeln(filevariable,'-----------  -----------------
-----------------');
close(filevariable);
end;
drain:
clrscr;gotoxy(1,10);
write('What number Drain inlet is this?  ');
read(di);
writeln;
writeln('What is the numbers of the pipes coming into the Drain inlet?
writeln('(seperate pipe numbers with comma''s)');
read(pi);
writeln;
writeln('What is the numbers of the pipes exiting the Drain inlet?  ');
writeln('(seperate pipe numbers with comma''s)');
read(po);
writeln;
writeln;
writeln('This is Drain inlet number ',di,'');
writeln('The numbers of the pipes intering are ',pi,'');
writeln('The numbers of the pipes exiting are ',po,'');
writeln;
writeln('Are these all correct?  (Y,N)');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
case upcase(YN) of
  'N' : goto drain;
end;
begin
append(filevariable);
writeln(filevariable,'',di,'       ',pi,'  ',po,'');
```

```
close(filevariable);
end;
writeln('Are there any more drain inlets to do?  (Y,N)');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
case upcase(YN) of
  'Y' : goto drain;
end;
writeln;
writeln('Do you want to go to the menu for other selections?  (Y,N)');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
case upcase(YN) of
  'Y' : goto menu;
  'N' : goto finish;
end;
finish:
end;

procedure setupschedule;
begin
clrscr;
gotoxy(1,10);
repeat
repeat
write('What is the length of pipe number ',nmb:2:0,'. ');
read(aa);
val(aa,lgn,e);
until e = 0;
writeln;
Write('What type of material will the pipe be made of?  ');
readln(material);
writeln;
repeat
write('What will the invert in elevation be?  ');
read(aa);
val(aa,invin,e);
until e = 0;
writeln;
repeat
write('What will the invert out elevation be?  ');
read(aa);
val(aa,invout,e);
until e = 0;
writeln;
writeln;
writeln('    length = ',lgn:4:2,'');
writeln('  material = ',material,'');
writeln(' invert in = ',invin:4:2,'');
writeln('invert out = ',invout:4:2,'');
writeln;
writeln('Are these all correct?  (Y,N)');
```

```
read(kbd,YN);
until upcase(YN) in ['Y','N'];
case upcase(YN) of
  'N' : setupschedule;
end;
end;

procedure sloper(var slp: real);
begin
repeat
clrscr;
gotoxy(1,5);
writeln('In choosing pipe sizes, we must experiment with different
slopes.');
writeln;
writeln('What percent slope do you want to try first?  ');
write('(ie.  for 2% enter as 0.02)  ');
readln(ss);
val(ss,slp,e);
until e = 0;
writeln;
end;

{********************** main program ***********************************}
{$I vdmode.pas}
Begin
 textcolor(11);
  clrscr;
  gotoxy(1,10);
  credit;
  vidio_mode;
nmb := 1;
{********************** introduction ***********************************}
    clrscr;
    gotoXY(1,10);
    logo;
    writeln;
    writeln;
    writeln('Do you want to see the complete Rational Method introduction?
(Y,N)');
repeat
    read(kbd,YN);
  until upcase(YN) in ['Y','N'];
  if upcase(YN) = 'Y' then
   Complete_Intro;
{********************** acreage *****************************************}
rerun:
begin
   intro;
    acreage(area);
{********************** coefficient chart *******************************}
    coefficient_chart(GotoPrinter);
    If GotoPrinter Then
      printnote
    Else
      note;
    BEgin
      Chart_Choice(MenuSelection);
      lava:
      Case MenuSelection of
        '1': Homo;
        '2': Begin
               Repeat
                 Mult;
                 Method(DataRight);
               Until (DataRight);
               Rational (Coeff);
             end;
        '3': begin
                 Note;
                chart_choice(menuselection);
                goto lava;
             end;
        '4': The_End;
      End
    End;
{*********************** figuring intensity *****************************}
  TOC (time);
begin
storm:
clrscr;
gotoxy(1,10);
writeln('What year storm are you designing for?');
writeln;
textcolor(4);
writeln('<1>   2 year');
writeln('<2>   5 year');
writeln('<3>  10 year - (used by most Virginia localities)');
writeln('<4>  25 year');
writeln('<5>  50 year');
writeln('<6> 100 year');
writeln;
textcolor(3);
write('Choice? ');
repeat
read(kbd,choice);
until (choice in ['1','2','3','4','5','6']);
writeln;
textcolor(11);
writeln;
 write('To check : is it correct you selected ');
 case choice of
  '1' : write('a 2 year design storm? ');
  '2' : write('a 5 year design storm? ');
  '3' : write('a 10 year design storm? ');
  '4' : write('a 25 year design storm? ');
  '5' : write('a 50 year design storm? ');
  '6' : write('a 100 year design storm? ');
```

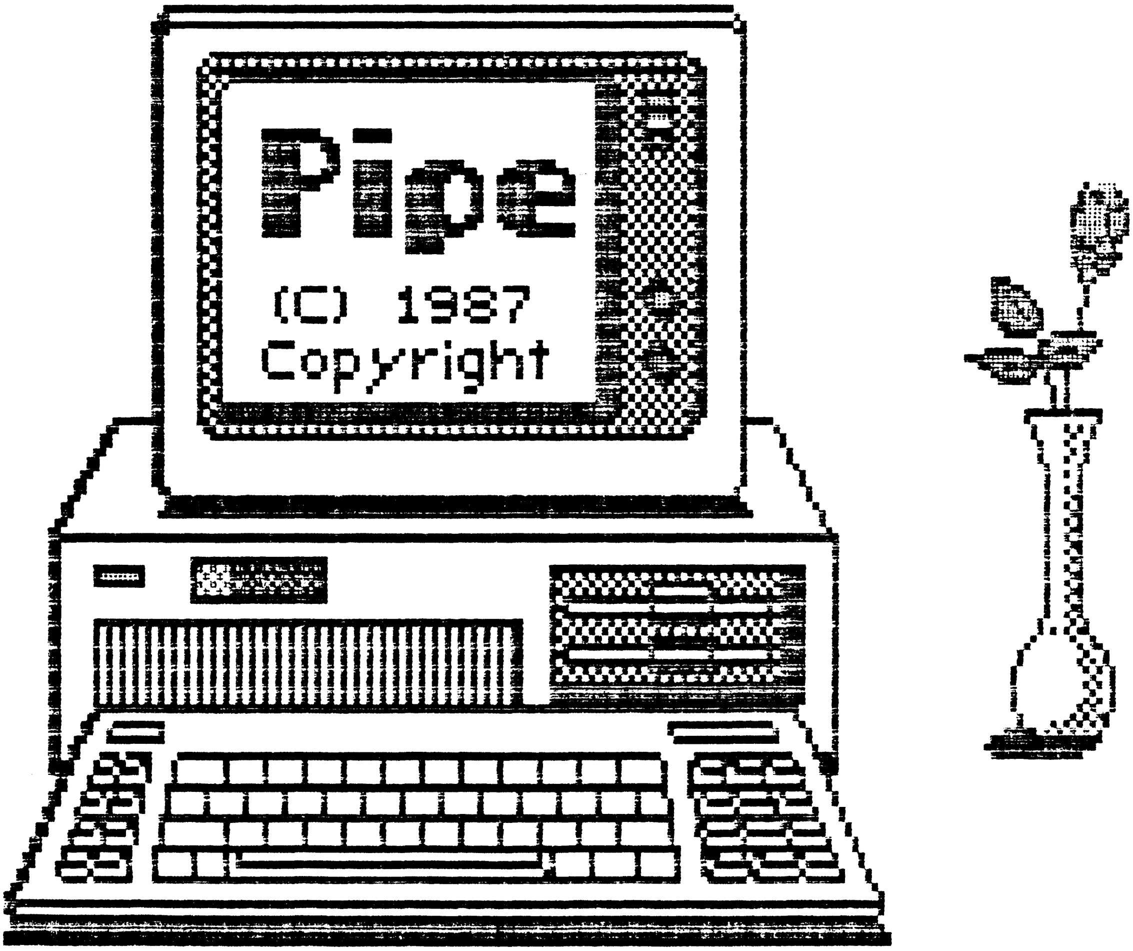
```
end;
write('  (Y,N)   ');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
if upcase(YN) = 'N' then goto storm;
end;
case choice of
 '1': year2(intensity);
 '2': year5(intensity);
 '3': year10(intensity);
 '4': year25(intensity);
 '5': year50(intensity);
 '6': year100(intensity);
 end;
{*********************** calculations ************************************
w := 0;
if nmb = 1 then goto apple;
writeln;
writeln('Is this pipe connected to the last pipe you entered? (Y,N)');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
if upcase(YN) = 'N' then goto apple;
peach := discharge;
  qdischarge(discharge);
 discharge := discharge + peach;
 w := 1;
apple:
if w = 1 then goto bannana;
  qdischarge(discharge);
  bannana:
w := 0;
  piperough(rough);
repeat
sloper(slp);
pipesize(diameter);
velosity(vel);
until redo = 2;
aretheseok(couldbe);
if couldbe = 'N' then goto rerun;
if couldbe = 'n' then goto rerun;
{*********************** pipe schedule ***********************************}
if nmb <> 1 then goto skip;
savefile;
skip:
setupschedule;
additions;
nmb := nmb + 1;
writeln;
repeat
write('Do you have any more pipes to figure out?  (Y,N)   ');
read(kbd,big);
until upcase(big) in ['Y','N'];
if upcase(big) = 'Y' then goto rerun;
begin
writeln;
write('Do you want to print the pipe schedule?  (Y,N) ');
repeat
read(kbd,bigger);
until upcase(bigger) in ['Y','N'];
writeln;
end;
case upcase(bigger) of
 'Y' : begin
       writeln('Make sure printer is ready, and press Enter.');
       read(kbd,qut);
       printfile;
       end;
 'N' : writeln;
end;

{**************************** manhole schedule ************************};
clrscr;
gotoxy(1,10);
writeln('Do you want to setup and print a Mänhole, Catch basin,');
write('Drain inlet schedule?  (Y,N)');
repeat
read(kbd,YN);
until upcase(YN) in ['Y','N'];
case upcase(YN) of
  'Y' : begin
        setupmanhole;
        writeln;
        writeln('Make sure your printer is ready, and press Enter.');
        read(kbd,qut);
        printmanhole;
        the_end;
        end;
  'N' : the_end;
end;
end;
end.
}
```

Appendix

C

Summary of files on disk



This is an explanation of the programs and directories you see on this disk.

PROGRAMS IN ROOT DIRECTORY

COMMAND.COM > Command processor program (this is all you need to know about it)

READ-ME.DOC > A document file that explains what the programs on the disk do.

PIPE.BAT > The batch file that begins the program. Just type "pipe" and press Enter.

LIST.COM > A program to take place of the type command, type 'list <filename.ext>'.

AUTOEXEC.BAT > The batch file that automatically 'boots' the computer. Just put the pipe disk in drive A: and turn on the power. It will load automatically.

PIPE.DOC > The documented procedures I went through to complete this project (this document).

TWO SUB-DIRECTORIES

PIPE <DIR> > The directory that contains the operating files of the program.

*** This section explains what each of the files in the *** Pipe directory do

PIPE.DAT > The data file for the pipe schedule. Can be viewed or printed as is.

MANHOLE.DAT > The data file for the manhole, catch basin, drain inlet schedule. Can be viewed or printed as is.

PIPE.COM > The compiled 'ready to run' main program.

NCPIPE.COM > The black and white version for monochrome monitors.

PIPE.PIC > The leading picture when the program is evoked.

POP.COM > The program used to show PIPE.PIC.

ASK.COM > The program used to ask if color or mono is being used.

SOURCE <DIR> > The directory that contains all the required Turbo Pascal source code. To see the files type 'cd\source' and press Enter. Then type 'dir' Enter. To get out type 'cd\' Enter.
The files in this directory are explained below.

*** This section explains what each of the files in the Source directory do ***

PIPE.PAS > This is the main program file.

NCPIPE.PAS > This is the main program file for non-color screens.

IBMTYPE.PAS > MachineType returns a code number that tells the kind of IBM machine that the program is running on.

The codes are:
1- IBM PC
2- IBM PC XT or Portable PC
3- IBM PCjr
4- IBM PC AT
5- IBM Convertible
6- Undetermined

NOTES:
1. The value 6 ("Undetermined") may be returned by some IBM compatibles.
2. Early models of the PC XT may return the code for a standard PC.
3. The code is a conversion of the value found in memory at F000:FFFE.

DOSVERS.PAS > DOSVersion returns the number of the DOS version under which the program is running. A version number consists of two parts. For version 2.1, the 2 is the major version number, and the 1 is the minor version number. The latter should actually be regarded as the two digit number 10. Rather than return separate values for the two parts of the version number, this routine returns a three digit decimal number. Thus, version 2.1 is reported as 210.

NUMSRIAL.PAS > NumberSerial returns the number of serial ports.

NUMPARLL.PAS > NumberParallel returns the number of parallel ports.

NOTES:
The values returned by the two previous functions merely reflect the presence of the port hardware; they do not indicate whether devices are connected to the ports.

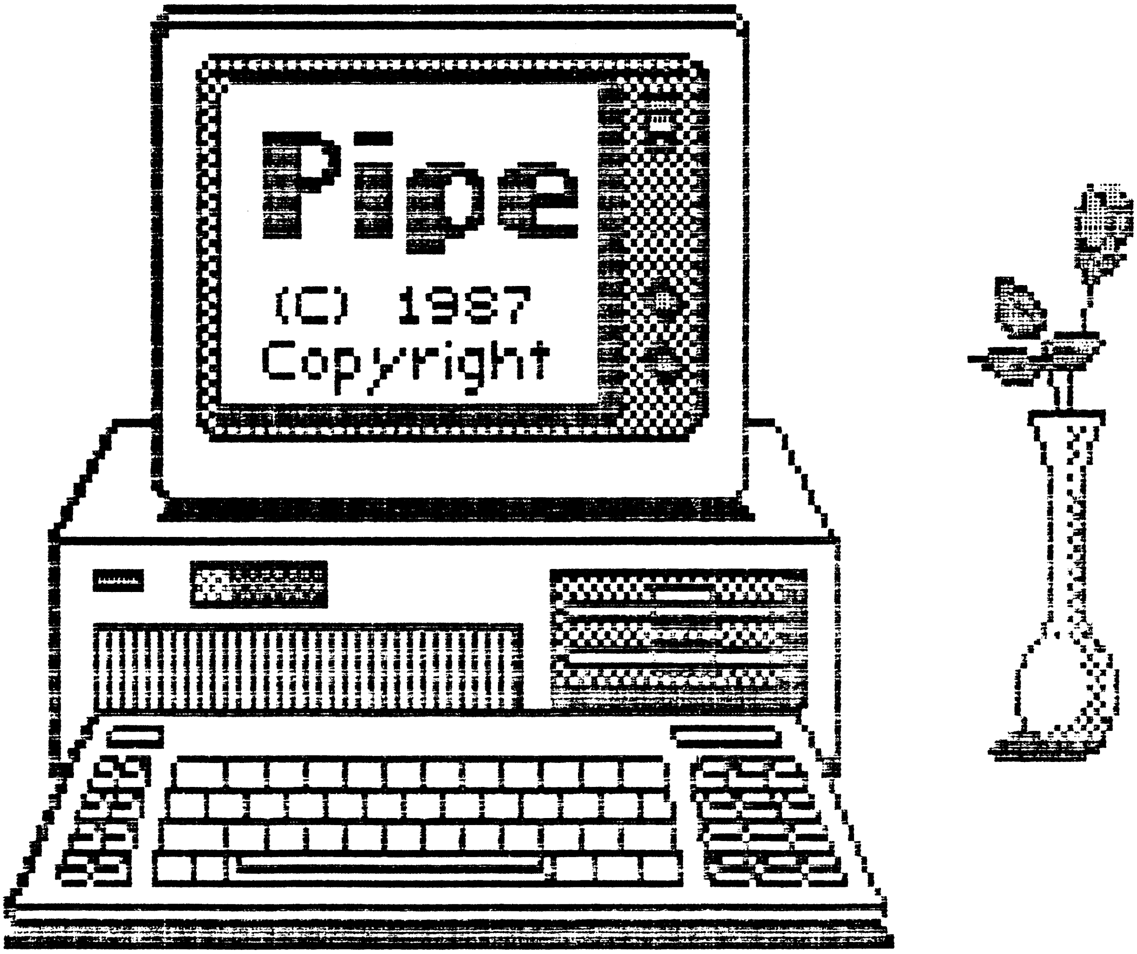
GAMEPORT.PAS > GamePort returns true when the machine has a game port. It does not indicate weather any device is connected to the port. This routine tests the port (201h) directly.

VDMODE.PAS > VideoMode tells what mode your monitor is in.

Appendix

D

Definition of Terms



Definition of Terms

Low level language - easy to learn but very slow and confusing to follow (Basic)

High level language - structured, technical language, easy to follow but difficult to learn without training, very fast run time (Pascal)

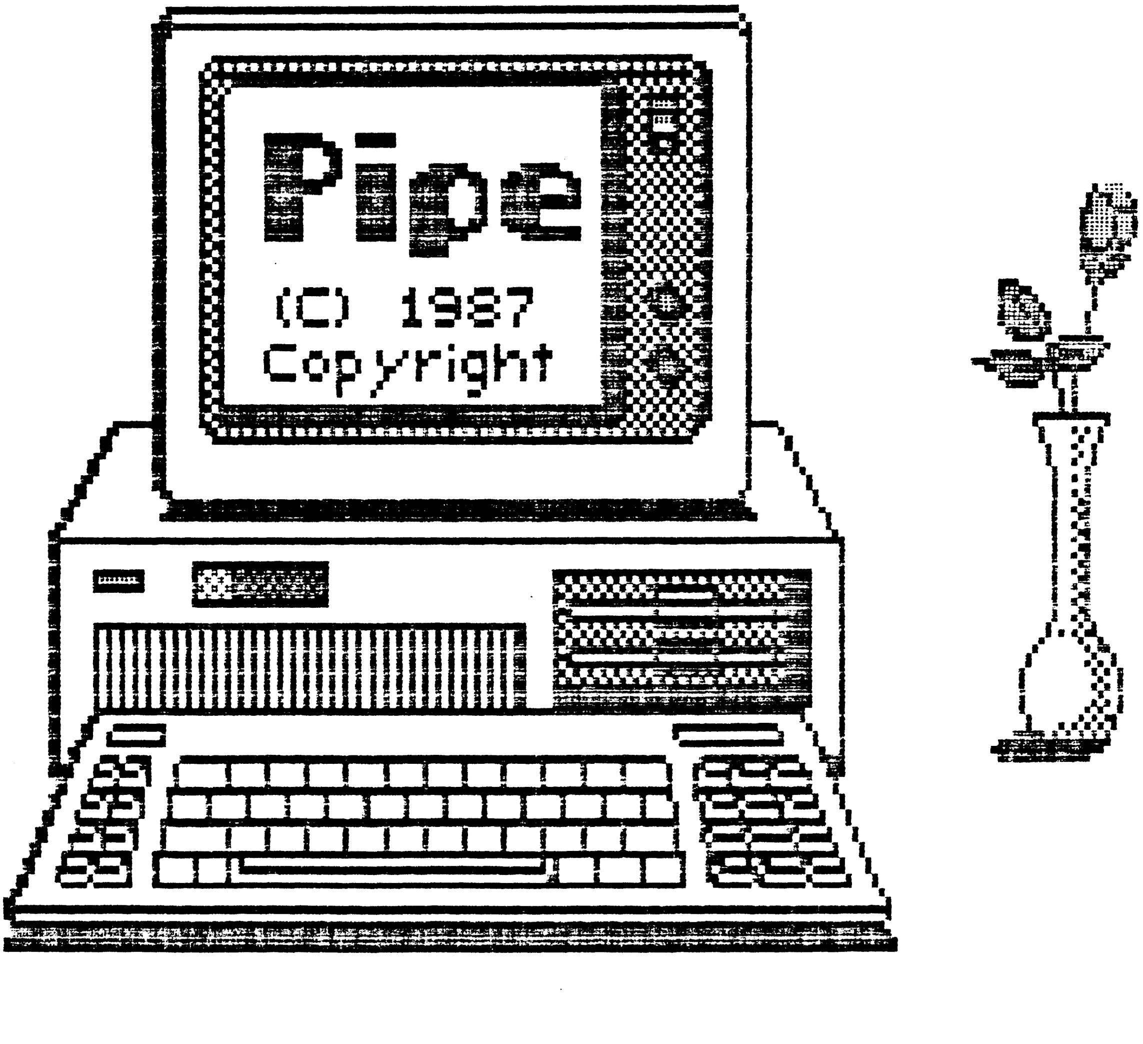
Nomograph - a graph, usually containing three parallel scales graduated for different variables so that when a straight line connects values of any two, the related value may be read dirrectly from the third at the point intersected by the line.

Design storm - The return period, in years, for a particular intensity of storm

Appendix

E

Proposal



A computer program in hydrology for Landscape Architecture.

A Proposal - toward the completion of the
requirements for the Degree of
Bachelor of Landscape Architecture.

By:
Timothy R. Quillen

Blacksburg, VA. 24060

October 17 1986

Approved By: ______________________________________
Mr. Mike Deeter ADVISOR

____
Mr. Benjamin C. Johnson INSTRUCTOR

____
Dr. Patrick A. Miller CHAIRMAN

BACKGROUND

Storm water management is a problem every land planner has to deal with. The problem is I don't have to do it very often; therefore I have to relearn the process every time I do it because things are forgotten.

Being interested in computers and computer applications, I have decided to write a program that will assist the designer with storm water calculations; this program will be referred to as "Pipe". Pipe will be a user - friendly program and will take the user through the entire process of storm water calculations, via a variety of menus and questions. Pipe will be written in Turbo Pascal, to allow it to run on a wide variety of machines.

THESIS STATEMENT

To write a workable program in Turbo Pascal for the purpose of doing the entire process of storm drainage, using the Virginia Erosion and Sediment Control Handbook's chapter five, as the model. And to apply it in a former hydrology project.

SPECIFIC TASKS AND METHODOLOGY

Generally speaking, the tasks to be undertaken by Tim are as follows:

1> Research

2> Write the program "Pipe", using Turbo Pascal.

3> Apply "Pipe" to a Landscape Architecture type project.

4> Synthesis all work into a final form.

PRODUCTS

1> A self bootable computer disk and manual.

2> Computer drawn Landscape plan and documented use of "Pipe".

3> The findings of the application.

PROJECT GOALS

A. To create a useful tool for the LAR profession,"Pipe".

B. To combine my knowledge of computers, and landscape architecture.

PERSONAL GOALS

A. To learn more about computer applications in order that I may be more valuable in an office situation.

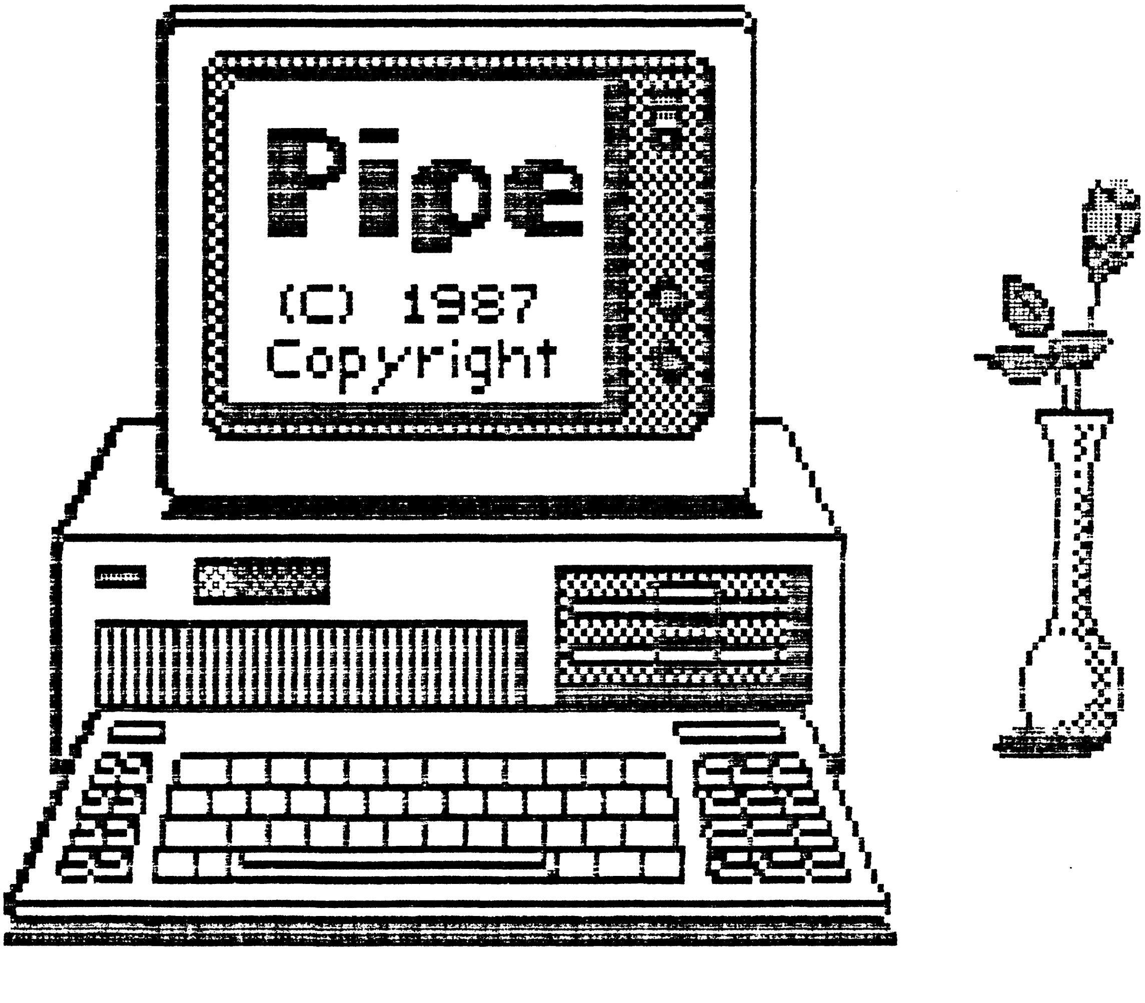
B. To create a product that will aid me in finding a job.

C. To become very knowledgeable in computer applications and processes that deal in Landscape Architecture.

Appendix

F

Original charts for manual technique



1980

Rational Method

The Rational Formula is the most commonly used method of determining peak discharges from small drainage areas. This method is traditionally used to size storm sewers, channels and other drainage structures which handle runoff from drainage areas less than 200 acres.

The Rational Formula is $Q = CiA$ in which:

Q = Peak Rate of runoff in cubic feet per second.

C = Runoff coefficient, an empirical coefficient representing a relationship between rainfall and runoff.

i = Average intensity of rainfall in inches per hour for the time of concentration (T_C) for a selected frequency of occurrence or return period.

T_C = *Time of concentration, the estimated time required for runoff* to flow from the most remote part of the area under consideration to the point under consideration. It consists of the total time for overland sheet flow, open channel flow and pipe flow.

A = Area drained in acres.

This is an empirical formula and its accuracy is highly dependent upon the judgement and experience of the user.

The general procedure for determining peak discharge using the Rational Formula is as follows:

Step 1 - Determine the drainage area (in acres).

Step 2 - Determine the runoff coefficient (C) for the type of soil/cover in the drainage area. (Table 5-2)

If the land use and soil cover is homogeneous over the drainage area, a (C) value can be determined directly from Table 5-2. If there are multiple soil cover conditions, a weighted average must be performed (see example 5-1).

Step 3 - Determine the time of concentration (T_C) for the drainage area. (i.e., the time of flow from the most remote point in the basin to the design point, in minutes).

The overland flow portion of flow time may be determined from Plate 5-1. The flow time (in minutes) in the channel sections can be estimated by calculating the average velocity How in a section of channel, and dividing the length of the section (in feet) by 60 times the average velocity (in feet per second).

The chart contained in Plate 5-2 may be used to check the time of concentration (T_c) for small drainage basins. The most conservative value (shortest time) should be used.

Step 4 - Determine the Rainfall Intensity Factor (i) for the selected design storm.

This is done by using the Rainfall Intensity - Frequency - Duration charts (Plates 5-3 through 5-7). Select the chart for the locality closest to your location. Enter the "Duration" axis of the chart with the calculated time of concentration (T_c). Move vertically until you intersect the curve of the appropriate design storm, then move horizontally to read the Rainfall Intensity Factor (i) in inches per hour.

Step 5 - Determine the peak discharge (Q - in cubic feet per second) by multiplying the previously determined factors.

$$Q = C(i)(A)$$

EXAMPLE 5-1

Given: Drainage Area: 80 acres
30% - Rooftops (24 acres)
10% - Streets & Driveways (8 acres)
20% - Lawns @5% slope (16 acres) on sandy soil
40% - Woodland (32 acres)

Time of Concentration (T_c) = 15 min.
Location: Lynchburg, Va.

Find: Peak runoff rate from 10-year frequency storm.

Solution:

1. Drainage Area = 80 acres (given)

2. Determine runoff coefficient (c)

Perform Weighted Average

	Area	X	C from Table 5-2	
Rooftops	24	X	.9	= 21.6
Streets	8	X	.9	= 7.2
Lawns	16	X	.15	= 2.4
Woodland	32	X	.10	= 3.2
	80			34.4

$$C = \frac{34.4}{80} = .43$$

3. Time of Concentration (T_C) = 15 min. (given)

4. Determine Rainfall Intensity Factor (i)
(i) = 4.9 in/hr - (from Plate 5-3)

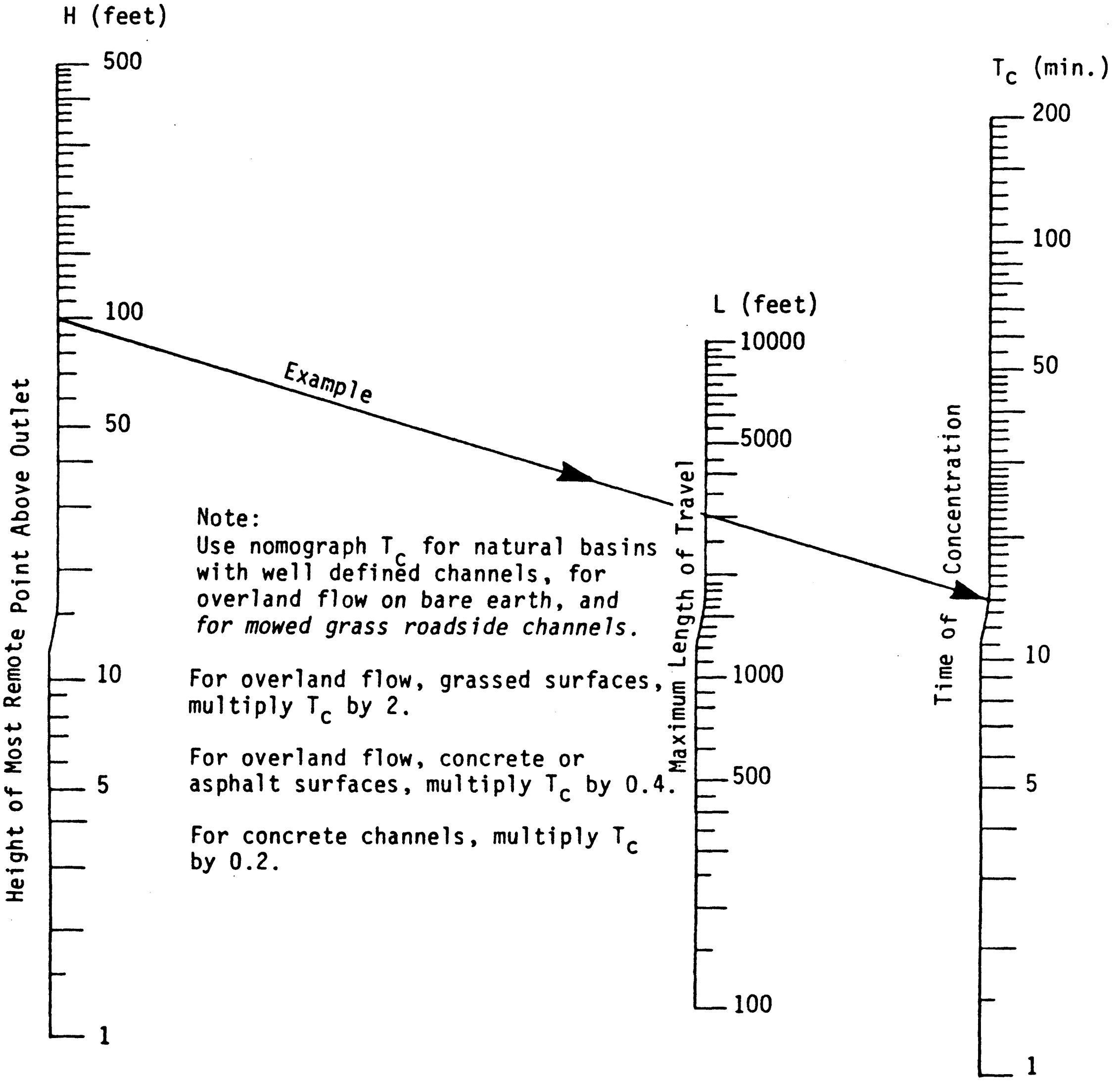
5. $Q = C(i)(A)$
$Q = .43(4.9)(80) = \underline{168.56 \text{ cfs}}$

Table 5-2
VALUES OF RUNOFF COEFFICIENT (C) FOR RATIONAL FORMULA

Land use	C	Land use	C
Business:		Lawns:	
Downtown areas	0.70-0.95	Sandy soil, flat, 2%	0.05-0.10
Neighborhood areas	0.50-0.70	Sandy soil, average, 2-7%	0.10-0.15
		Sandy soil, steep, 7%	0.15-0.20
Residential:		Heavy soil, flat, 2%	0.13-0.17
Single-family areas	0.30-0.50	Heavy soil, average, 2-7%	0.18-0.22
Multi units, detached	0.40-0.60	Heavy soil, steep, 7 %	0.25-0.35
Multi units, attached	0.60-0.75		
Suburban	0.25-0.40	Agricultural land:	
		Bare packed soil	
Industrial:		Smooth	0.30-0.60
Light areas	0.50-0.80	Rough	0.20-0.50
Heavy areas	0.60-0.90	Cultivated rows	
		Heavy soil no crop	0.30-0.60
Parks, cemeteries	0.10-0.25	Heavy soil with crop	0.20-0.50
		Sandy soil no crop	0.20-0.40
Playgrounds	0.20-0.35	Sandy soil with crop	0.10-0.25
		Pasture	
Railroad yard areas	0.20-0.40	Heavy soil	0.15-0.45
		Sandy soil	0.05-0.25
Unimproved areas	0.10-0.30	Woodlands	0.05-0.25
Streets:			
Asphaltic	0.70-0.95		
Concrete	0.80-0.95		
Brick	0.70-0.85		
Drives and walks	0.75-0.85		
Roofs	0.75-0.95		

Note: The designer must use judgement to select the appropriate C value within the range. Generally, larger areas with permeable soils, flat slopes and dense vegetation should have lowest (C) values. Smaller areas with dense soils, moderate to steep slopes, and sparce vegetation should be assigned highest (C) values.

Source: American Society of Civil Engineers



Note:
Use nomograph T_c for natural basins with well defined channels, for overland flow on bare earth, and *for mowed grass roadside channels.*

For overland flow, grassed surfaces, multiply T_c by 2.

For overland flow, concrete or asphalt surfaces, multiply T_c by 0.4.

For concrete channels, multiply T_c by 0.2.

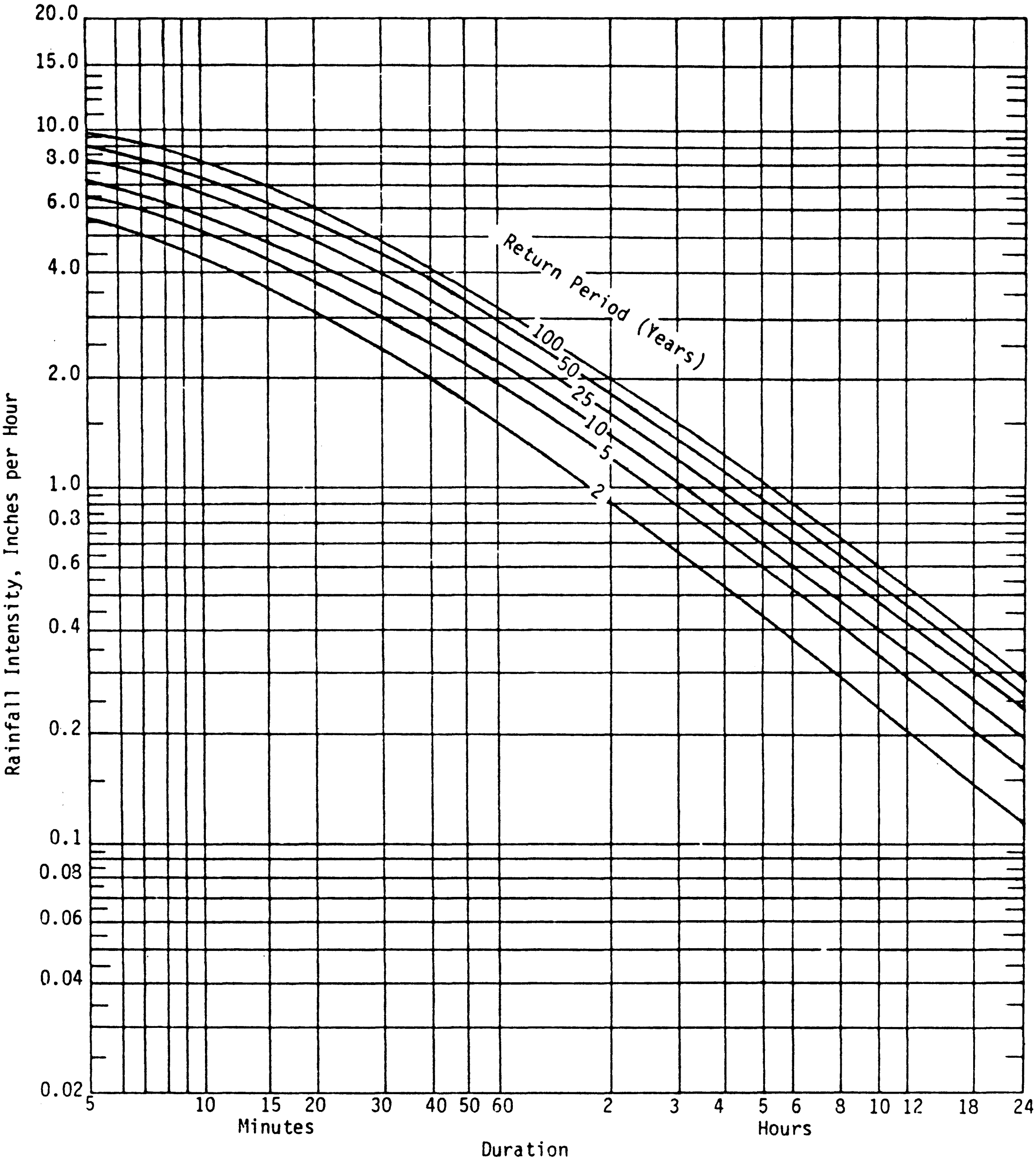
TIME OF CONCENTRATION OF SMALL DRAINAGE BASINS

Source: VDH&T

Plate 5-2

RAINFALL FREQUENCY-INTENSITY-DURATION CHART

LYNCHBURG, VA.

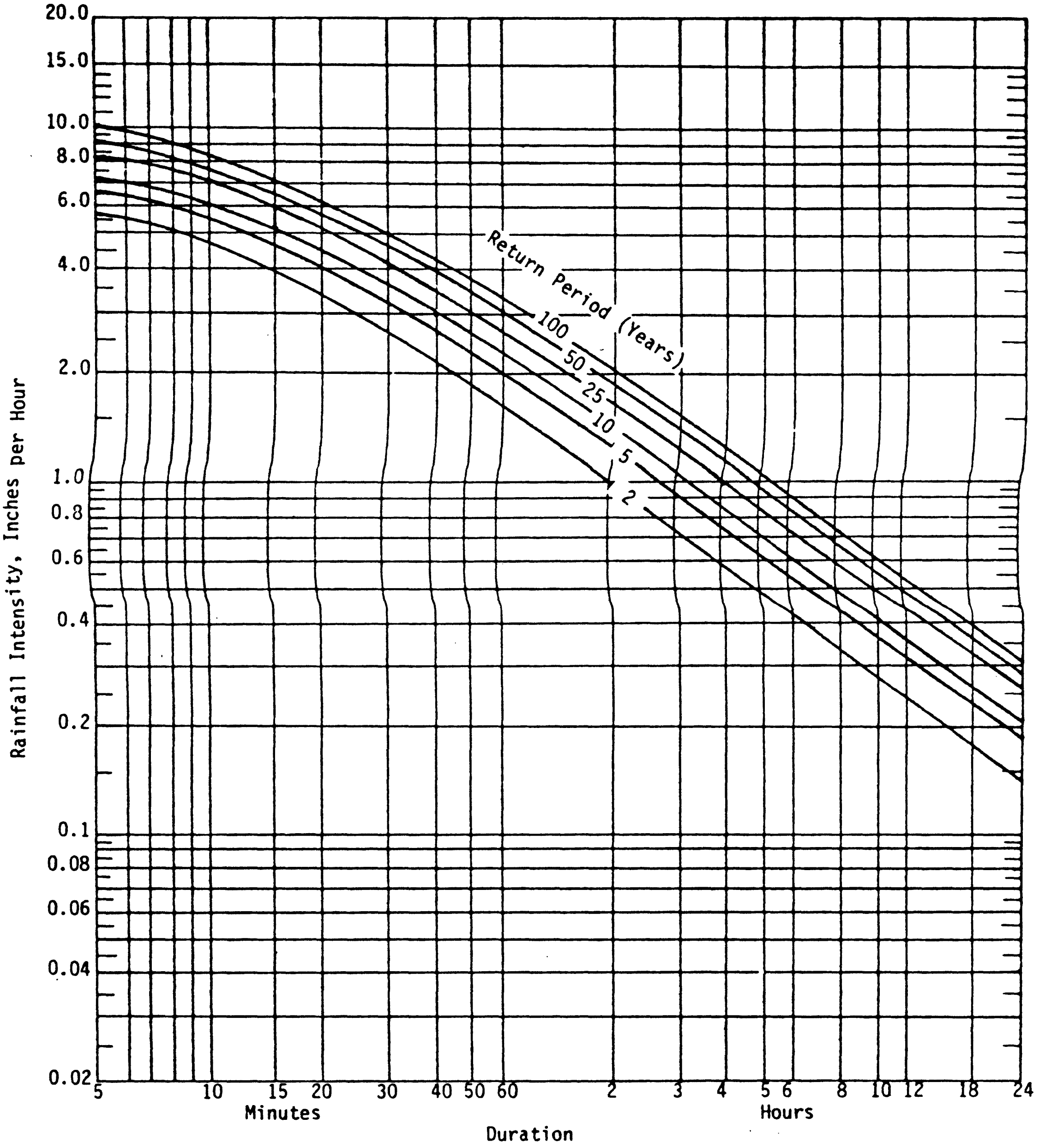


Source: VDH&T

Plate 5-3

RAINFALL FREQUENCY-INTENSITY-DURATION CHART

RICHMOND, VA.



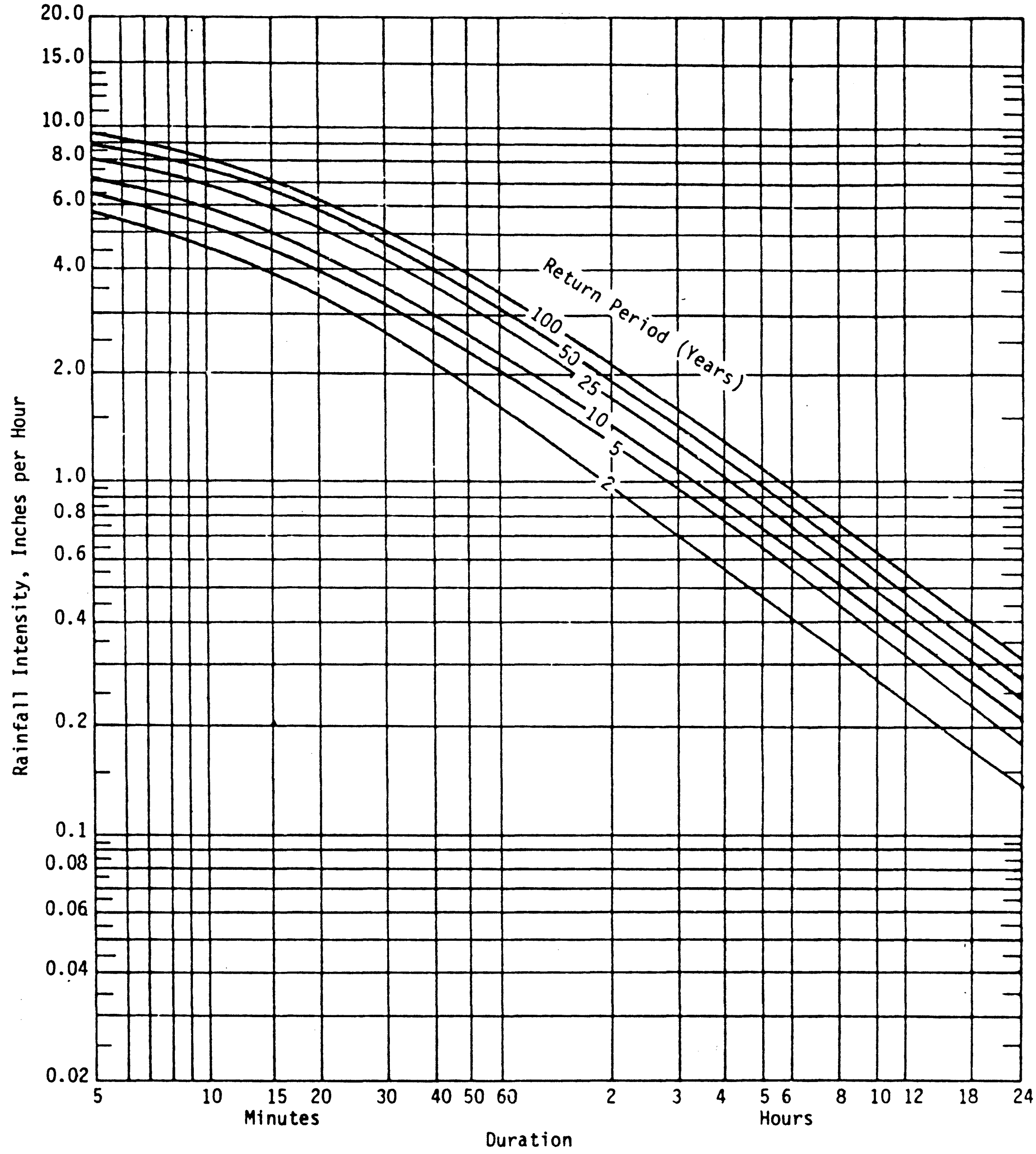
Source: VDH&T

Plate 5-4

1980

RAINFALL FREQUENCY-INTENSITY-DURATION CHART

NORFOLK, VA.



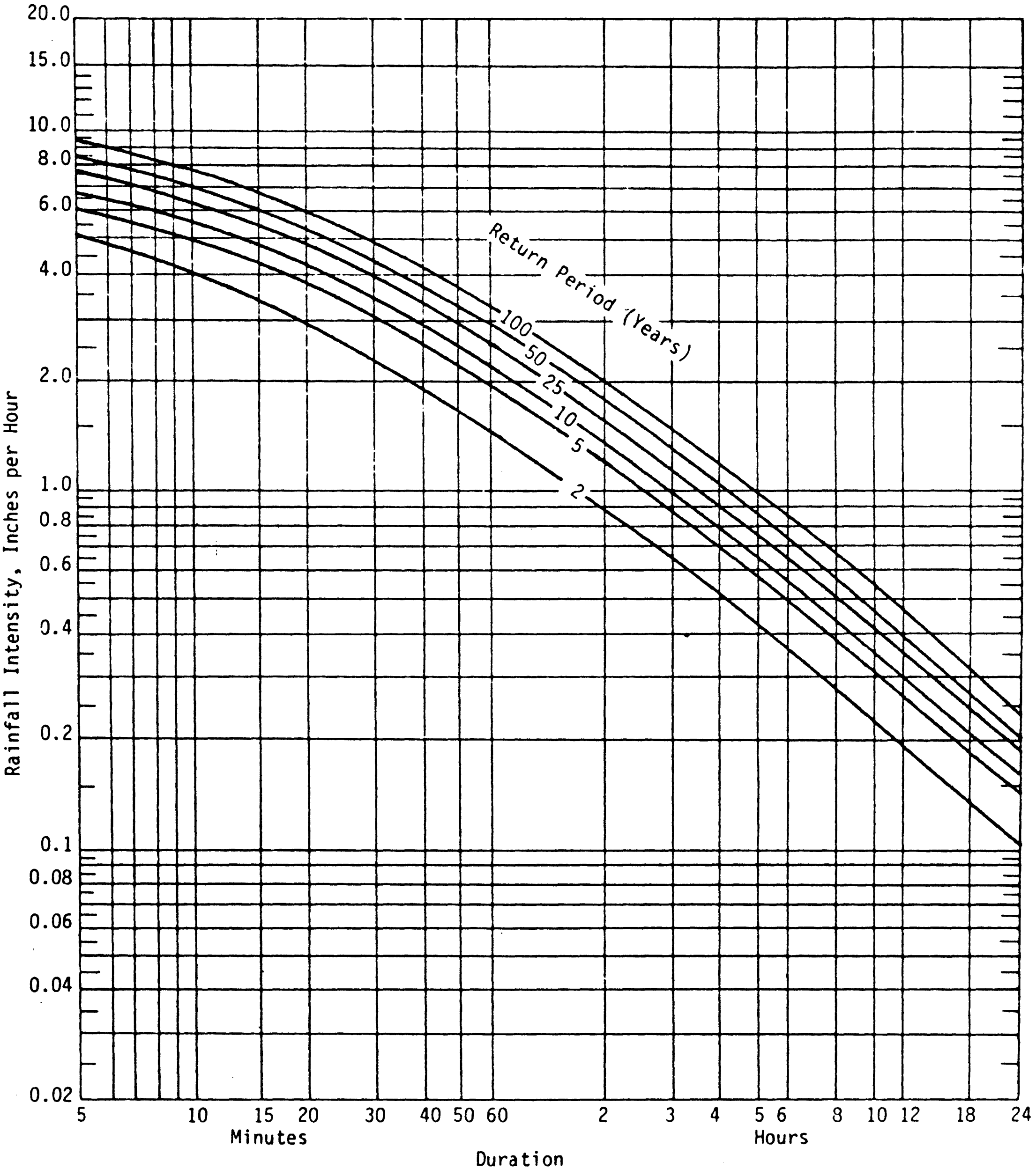
Source: VDH&T

Plate 5-5

1980

RAINFALL FREQUENCY-INTENSITY-DURATION CHART

WYTHEVILLE, VA.

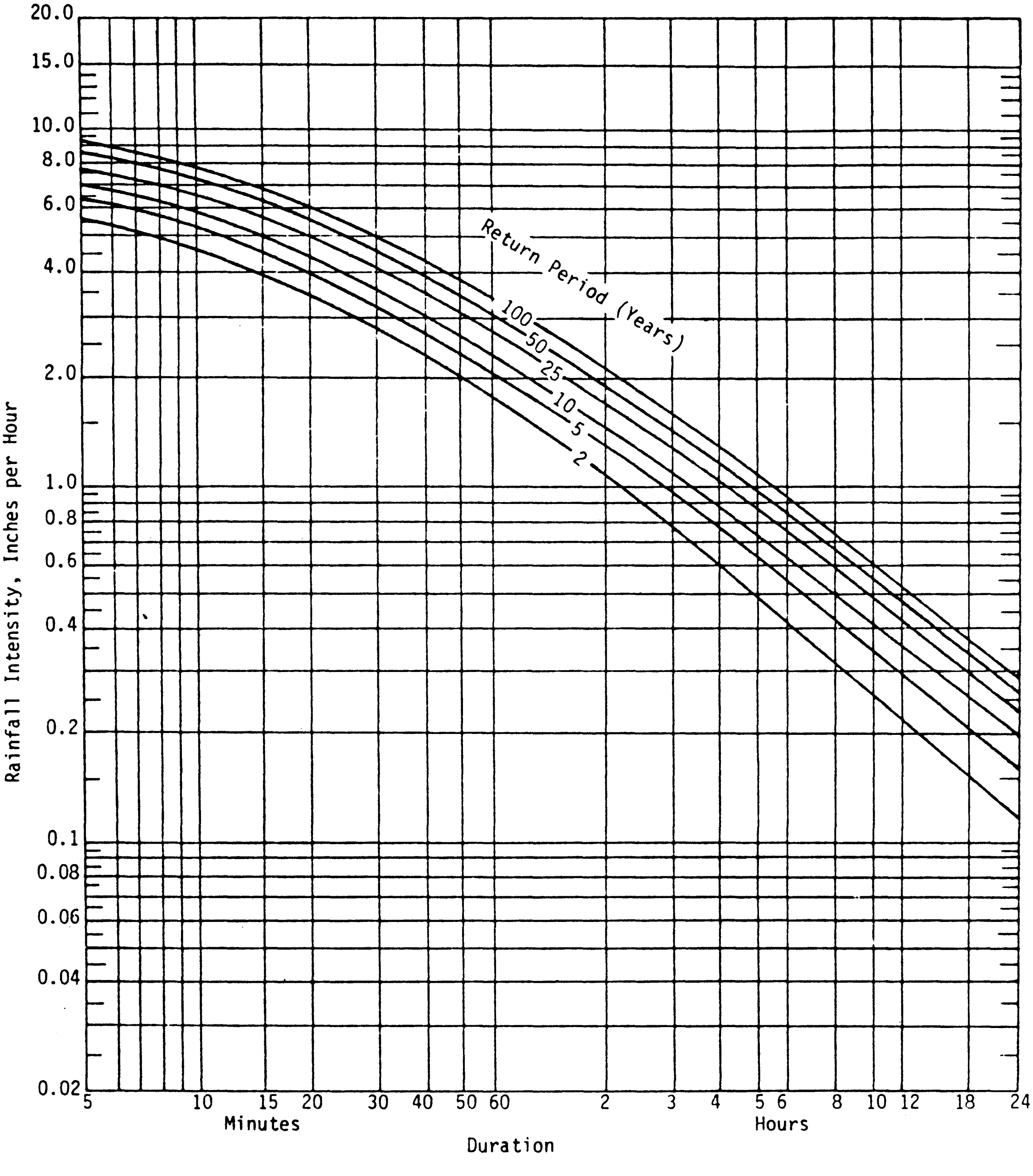


Source: VDH&T

Plate 5-6

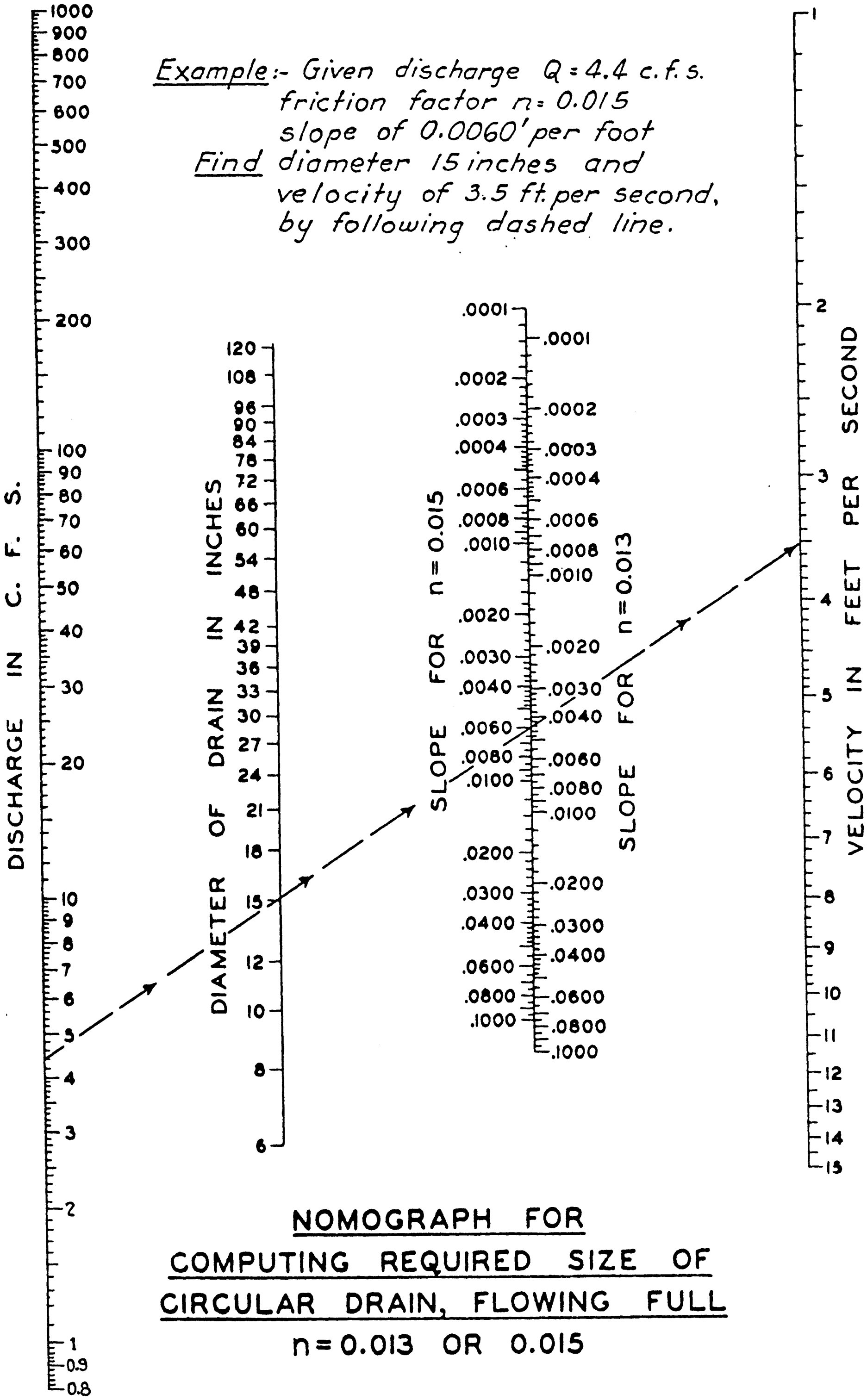
RAINFALL FREQUENCY-INTENSITY-DURATION CHART

CAPE HENRY, VA.



Source: VDH&T

Plate 5-7



Example:- Given discharge Q = 4.4 c.f.s.
friction factor n = 0.015
slope of 0.0060' per foot
Find diameter 15 inches and
velocity of 3.5 ft. per second,
by following dashed line.

NOMOGRAPH FOR
COMPUTING REQUIRED SIZE OF
CIRCULAR DRAIN, FLOWING FULL
n = 0.013 OR 0.015

Adapted from Engineering Manual, War Department, Corps of Engineers, Part XIII, Chap. 1, June 1955.