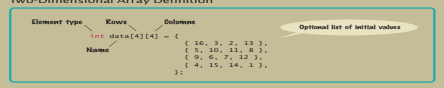
COMPUTER SCIENCE 1: STARTING COMPUTING CSCI 1300

The image part with relationship ID rId3 was not found in the file.

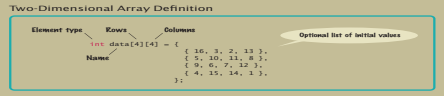
Ioana Fleming / Vipra Gupta
Spring 2018
Lecture 19

Two-Dimensional Array Definition



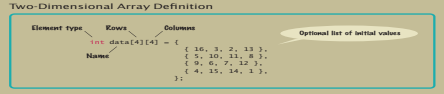
Announcements

- Rec 8 due on 3/10
- Hmwk 6 due on 3/11
- **Practicum 2: March 14th**, 2018
  - Loops: while, for
  - Strings
  - Arrays
  - File I/O



Agenda

- Today:
  - Reading from files and writing to files: how to
  - Applications



Reading and Writing Files – how to

1. Understand the processing task.
2. Determine which files you need to read and write.
3. Choose a method for obtaining thele names.
   - hardcode the file names
   - ask the user for input
4. Choose between line, word, and character-based input.

   4a. if we choose line, might need processing
5. Place repeatedly occurring tasks into functions.
6. If required, use manipulators to format the output.

Examples: cloud9

Example 1: reverse each line, write them in a new file

Example 2: reverse the lines, write them in a new file

Example 3: reverse each line, overwrite the original file

File I/O summary

Develop programs that read and write files.

- To read or write files, you use variables of type `fstream`, `ifstream`, or `ofstream`.
- When opening a file stream, you supply the name of the file stored on disk.
- Read from a file stream with the same operations that you use with `cin`.
- Write to a file stream with the same operations that you use with `cout`.
- Always use a reference parameter for a stream.

Be able to process text in files.

- When reading a string with the >> operator, the white space between words is consumed.
- You can get individual characters from a stream and unget the last one.
- You can read a line of input with the `getline` function and then process it further.

Write programs that neatly format their output.

- Use the `setw` manipulator to set the width of the next output.
- Use the `fixed` and `setprecision` manipulators to format floating-point numbers with a fixed number of digits after the decimal point.

More examples - applications

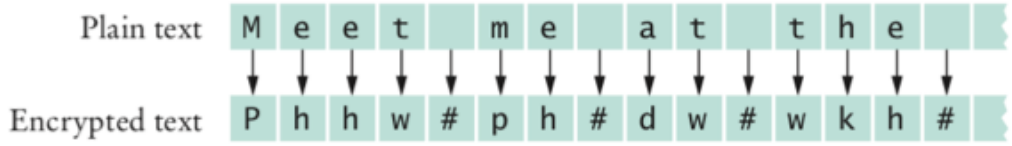


Figure 2 Caesar Cipher

Let's write a program that *encrypts* a file—that is, scrambles it so that it is unreadable except to those who know the decryption method. Ignoring 2,000 years of progress in the field of encryption, we will use a method familiar to Julius Caesar, replacing an A with a D, a B with an E, and so on. That is, each character c is replaced with $c + 3$ (see Figure 2).

How hard is it to decrypt it?

The emperor Julius Caesar used a simple scheme to encrypt messages.

More encryption

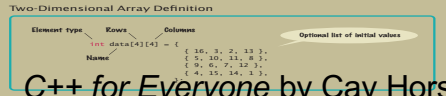
Random monoalphabet cipher. The Caesar cipher, which shifts all letters by a fixed amount, is far too easy to crack. Here is a better idea. As the key, don't use numbers but words. Suppose the key word is FEATHER. Then first remove duplicate letters, yielding FEATHR, and append the other letters of the alphabet in reverse order:

F	E	A	T	H	R	Z	Y	X	W	V	U	S	Q	P	O	N	M	L	K	J	I	G	D	C	B

Now encrypt the letters as follows:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
F	E	A	T	H	R	Z	Y	X	W	V	U	S	Q	P	O	N	M	L	K	J	I	G	D	C	B

How hard is it to decrypt it?



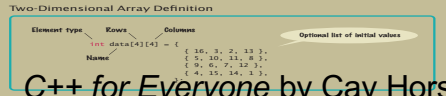
More encryption

Letter frequencies. If you encrypt a file using the cipher of Exercise P8.17, it will have all of its letters jumbled up, and will look as if there is no hope of decrypting it without knowing the keyword. Guessing the keyword seems hopeless too. There are just too many possible keywords. However, someone who is trained in decryption will be able to break this cipher in no time at all. The average letter frequencies of English letters are well known. The most common letter is E, which occurs about 13 percent of the time. Here are the average frequencies of the letters.

A	8%	H	4%	O	7%	U	3%
B	<1%	I	7%	P	3%	V	<1%
C	3%	J	<1%	Q	<1%	W	2%
D	4%	K	<1%	R	8%	X	<1%
E	13%	L	4%	S	6%	Y	2%
F	3%	M	3%	T	9%	Z	<1%
G	2%	N	8%				

Write a program that reads an input file and displays the letter frequencies in that file. Such a tool will help a code breaker. If the most frequent letters in an encrypted file are H and K, then there is an excellent chance that they are the encryptions of E and T.

Show the result in a table such as the one above, and make sure the columns line up.



More encryption

Vigenère cipher. In order to defeat a simple letter frequency analysis, the Vigenère cipher encodes a letter into one of several cipher letters, depending on its position in

the input document. Choose a keyword, for example `TIGER`. Then encode the first letter of the input text like this:

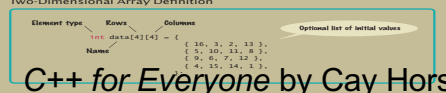
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S

The encoded alphabet is just the regular alphabet shifted to start at `T`, the first letter of the keyword `TIGER`. The second letter is encrypted according to the following map.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H

The third, fourth, and fifth letters in the input text are encrypted using the alphabet sequences beginning with characters `G`, `E`, and `R`, and so on. Because the key is only five letters long, the sixth letter of the input text is encrypted in the same way as the first.

Write a program that encrypts or decrypts an input text according to this cipher.



More encryption

Junk mail. Write a program that reads in two files: a *template* and a *database*. The template file contains text and tags. The tags have the form |1| |2| |3|... and need to be replaced with the first, second, third, ... field in the current database record.

A typical database looks like this:

```
Mr.|Harry|Morgan|1105 Torre Ave.|Cupertino|CA|95014
Dr.|John|Lee|702 Ninth Street Apt. 4|San Jose|CA|95109
Miss|Evelyn|Garcia|1101 S. University Place|Ann Arbor|MI|48105
```

And here is a typical form letter:

```
To:
|1| |2| |3|
|4|
|5|, |6| |7|

Dear |1| |3|:

You and the |3| family may be the lucky winners of $10,000,000 in the C++ compiler
clearinghouse sweepstakes! ...
```