This is CS 50.

Harvard College's Introduction to Computer Science I

COMPUTER SCIENCE 50

WEEK 2

DAVID J. MALAN '99
malan@post.harvard.edu

Or fher gb qevax lbhe binygvar!

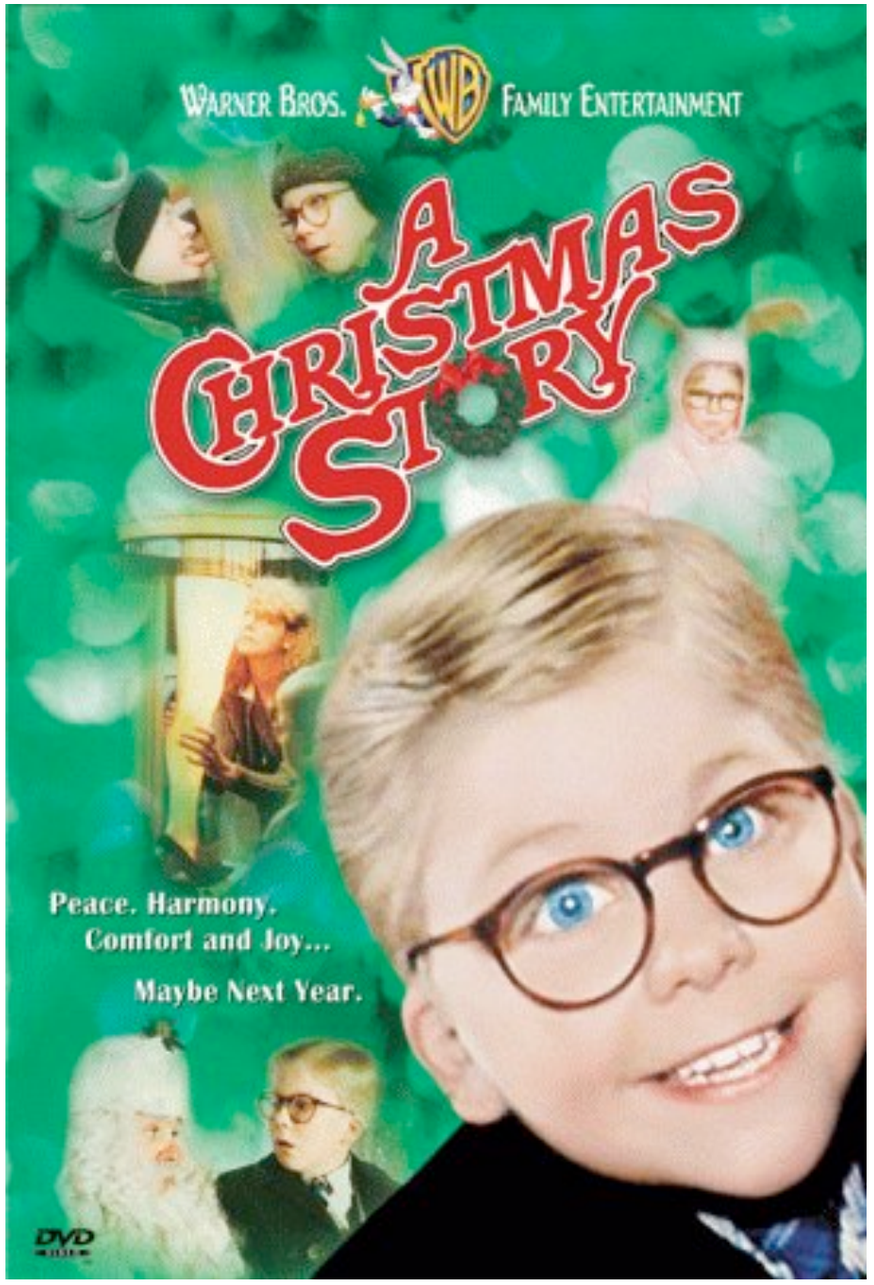


Image from http://www.questexperiences.com/quest2/movieadventures/default.asp.

ROT13

"Double ROT13 is pretty good, but for extra security, quadruple ROT13 is available. It is probably pretty computationally expensive to pass it through the cipher than many times, but security is worth it."

http://channel9.msdn.com/forums/Coffeehouse/221675-Can-someone-translate-this-please/

Academic Honesty

1 2 3 4 5

6 7 8 9 10

11 12 13 14 15

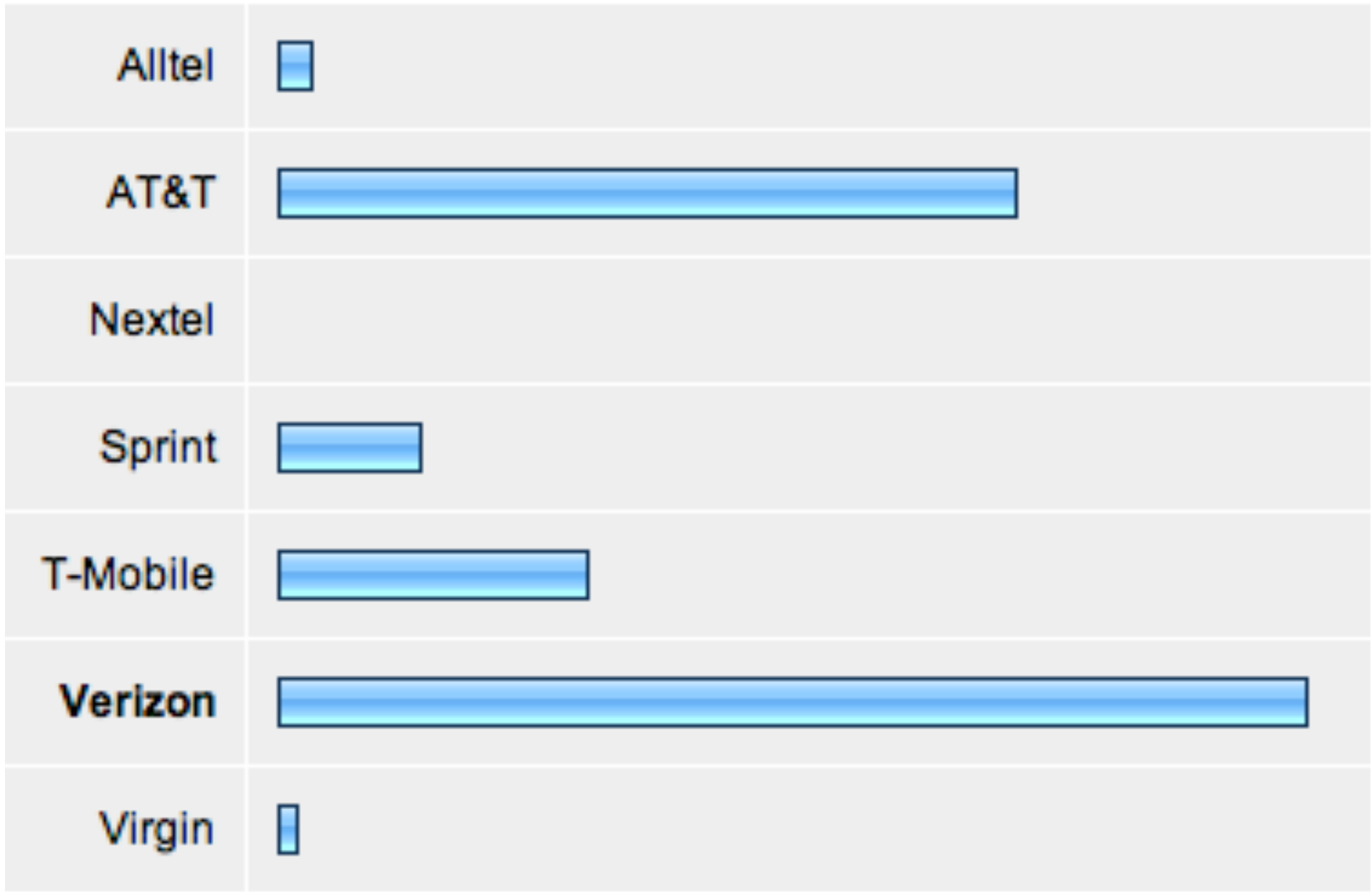
16 17 18 19 20

21 22 23 24 25

Mobile Phones

BlackBerry	
iPhone	
Pre	
Treo	
Some other "smart" phone	
Just a "normal" mobile phone	
None	

Mobile Carriers



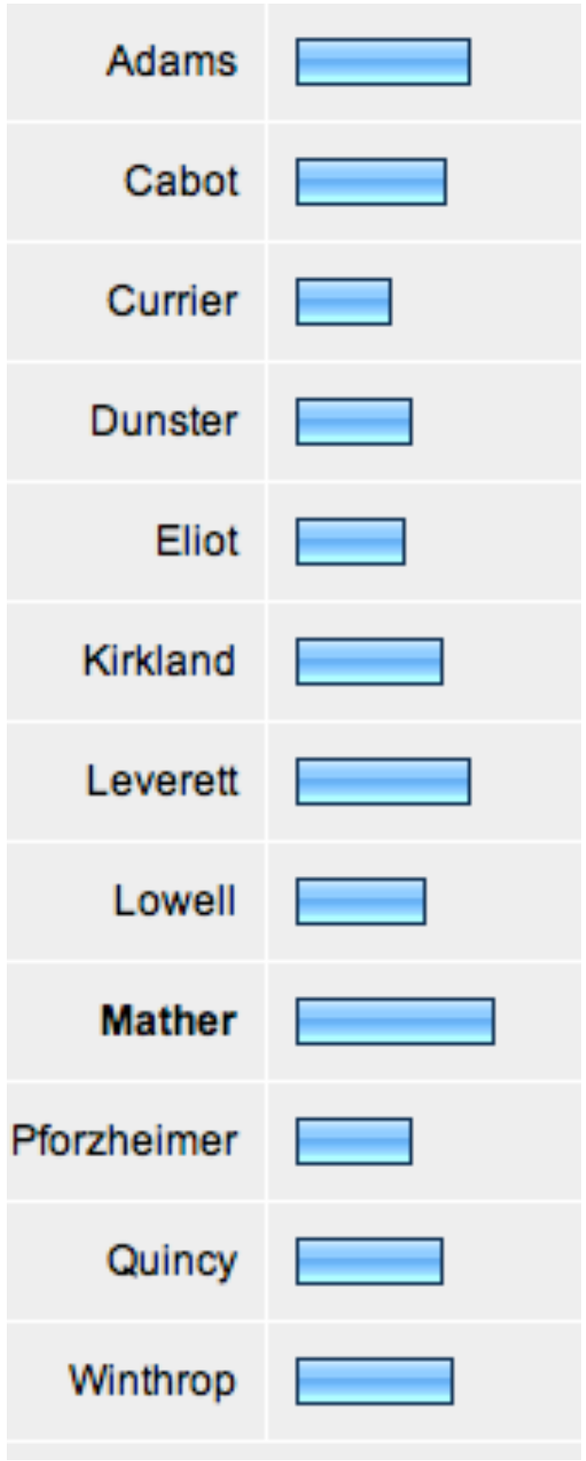
And you are...

Freshman	
Sophomore	
Junior	
Senior	
Grad Student at GSAS	
Extension Student	

Dorms

Apley Court		Matthews	
Canaday		Mower	
Grays		Pennypacker	
Greenough		Stoughton	
Hollis		Straus	
Holworthy		Thayer	
Hurlbut		Weld	
Lionel		Wigglesworth	
Mass Hall		---	

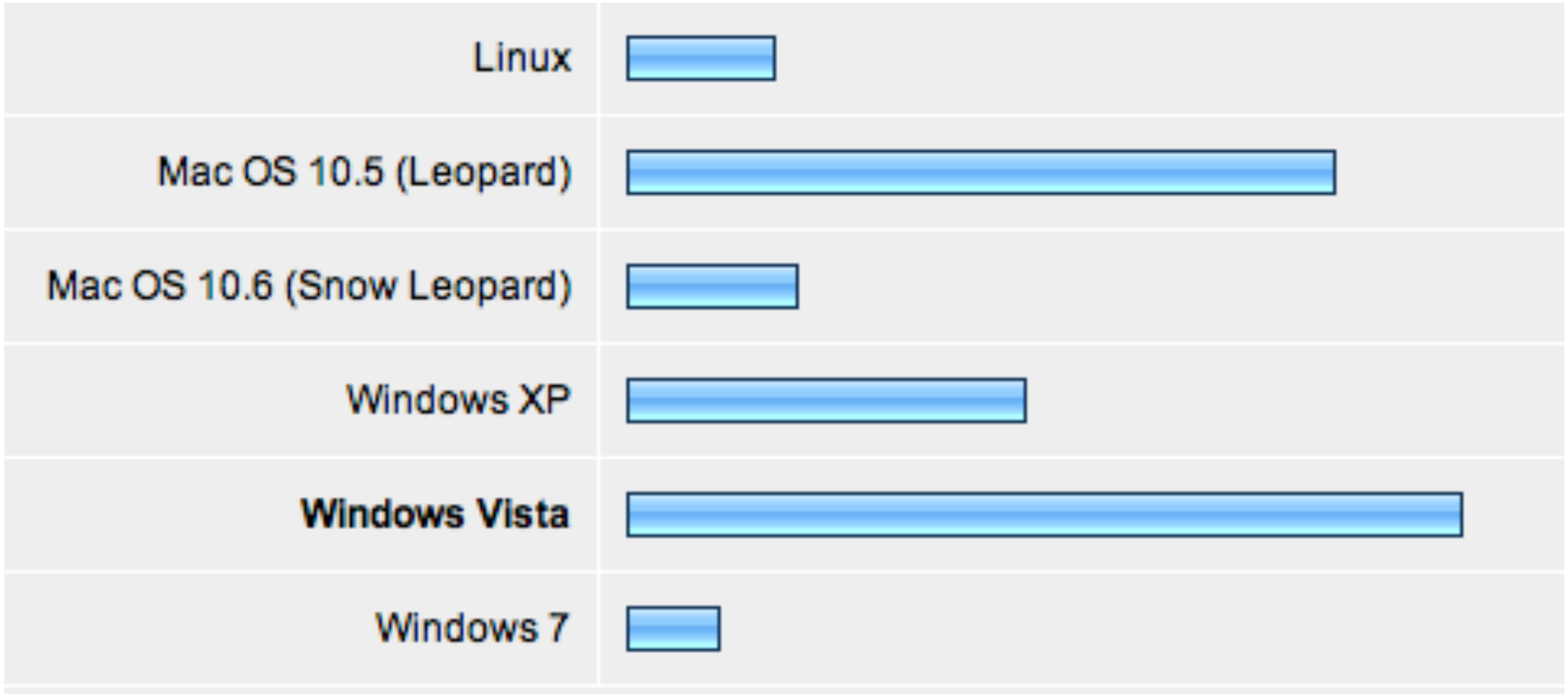
Houses


Adams
Cabot
Currier
Dunster
Eliot
Kirkland
Leverett
Lowell
Mather
Pforzheimer
Quincy
Winthrop

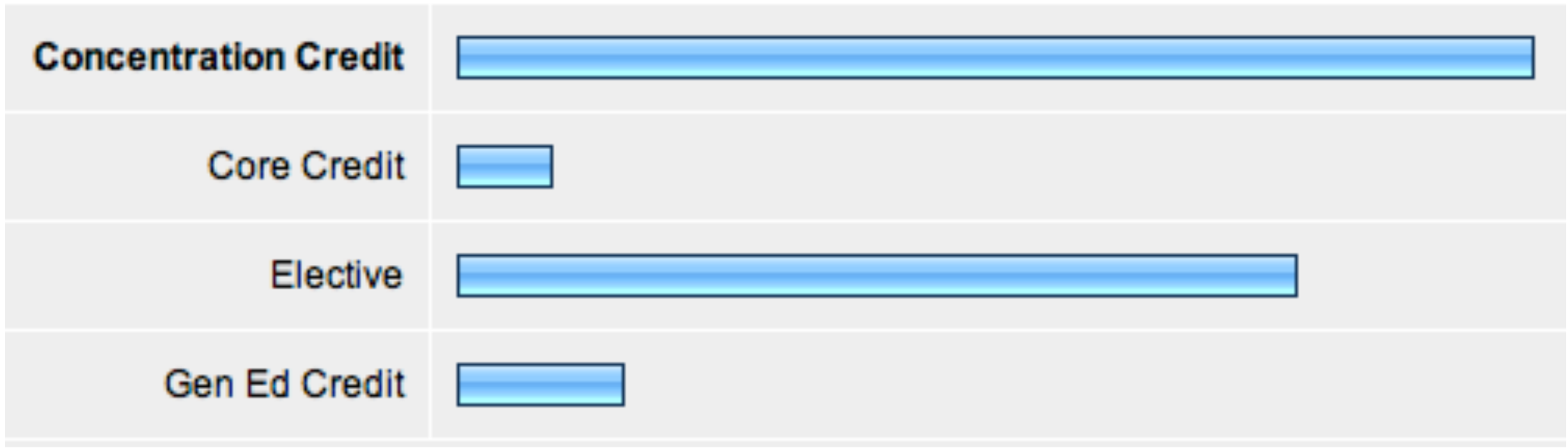
Best House

Adams	
Cabot	
Currier	
Dunster	
Eliot	
Kirkland	
Leverett	
Lowell	
Mather	
Pforzheimer	
Quincy	
Winthrop	

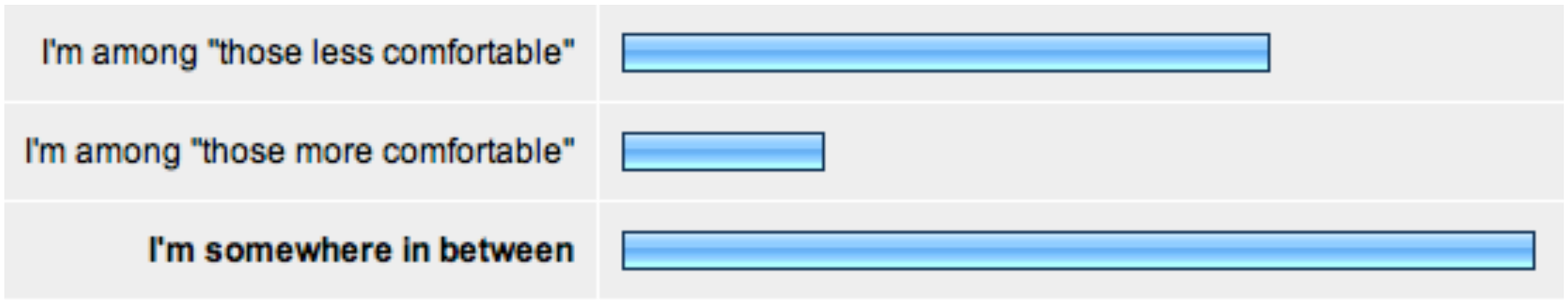
Operating Systems



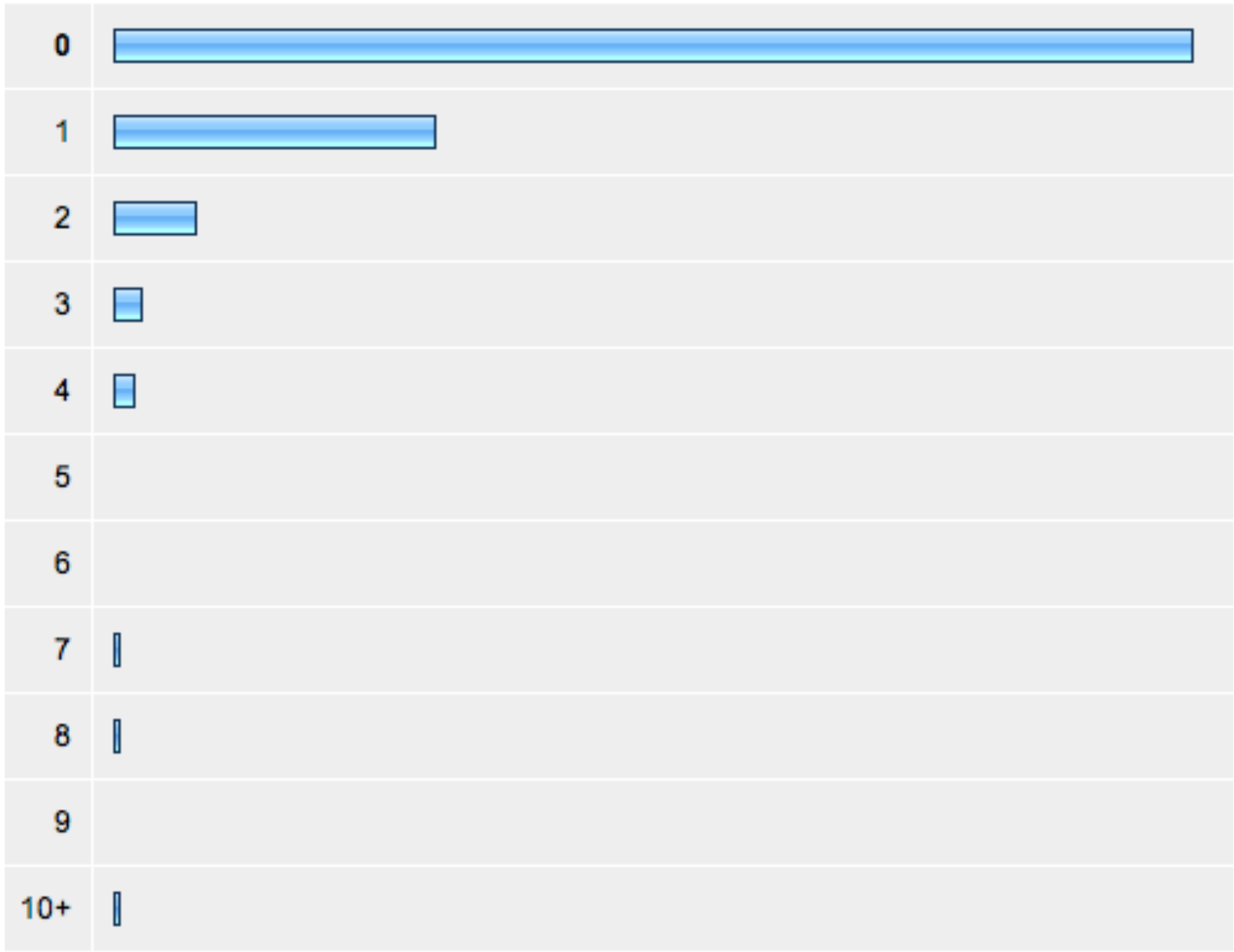
Why 50?



Comfort Levels



Prior Courses

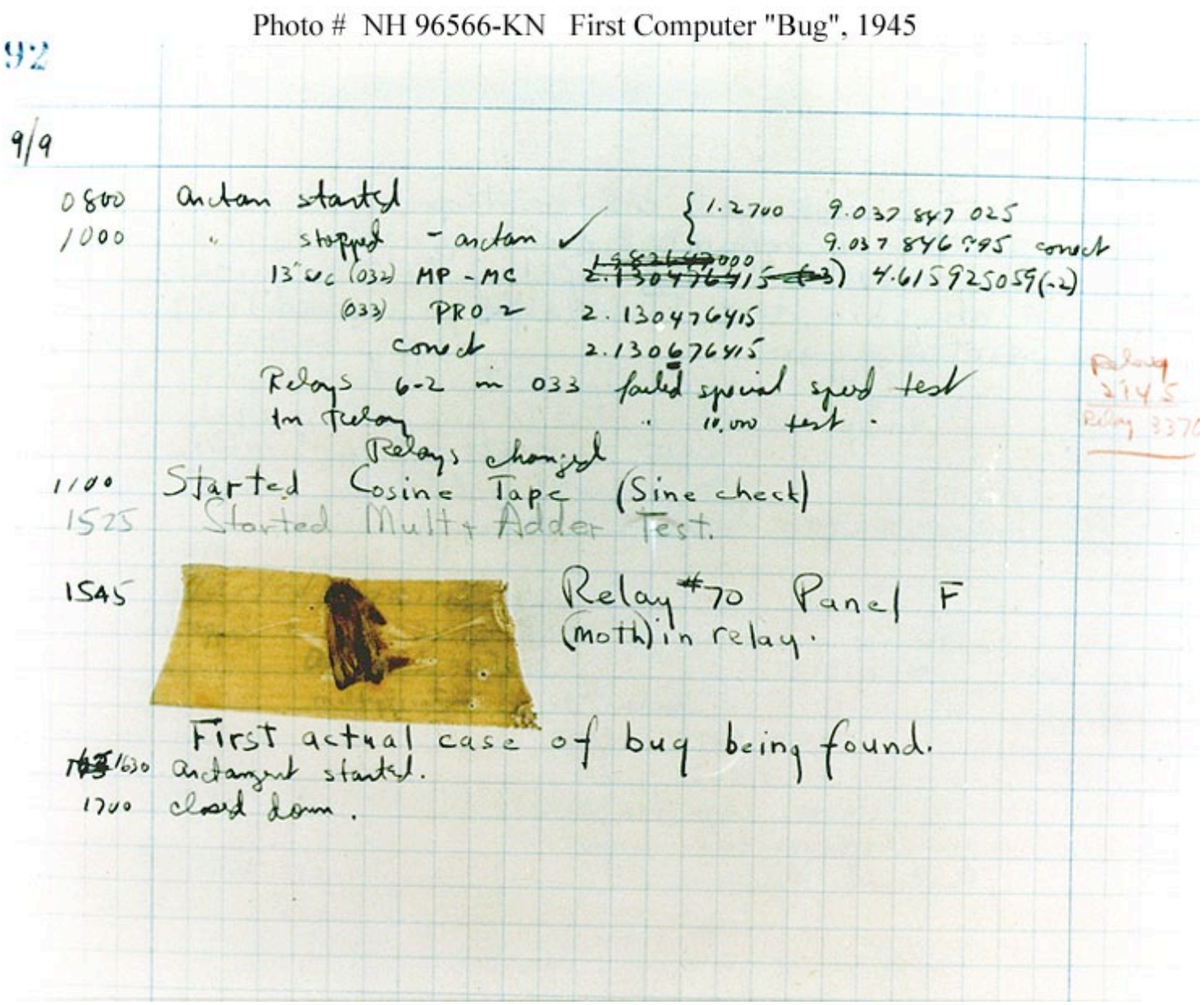


How to Write a Program in...

- C++
- Java
- LISP
- Perl
- PHP
- ...

see
hai.{cc,lisp,php,pl}, Hai.java

Bugs



see
buggy{1,2}.c

Image from http://www.history.navy.mil/.

Casting

```
int i = (int) 'A';
char c = (char) 65;
```

see
ascii{1,2,3}.c, battleship.c

Functions

Parameters and Arguments

99 bottles of beer on the wall,
99 bottles of beer,
Take one down, pass it around,
98 bottles of beer on the wall.

see
beer{1,2,3,4}.c

Image from http://z.about.com/d/tvcomedies/1/7/n/5/-/-/homer_simpson.jpg.

Functions

Scope, Local Variables, Temporary Variables

```
void
swap(int a, int b)
{
    int tmp;

    tmp = a;
    a = b;
    b = tmp;
}
```

see
buggy3.c

Functions

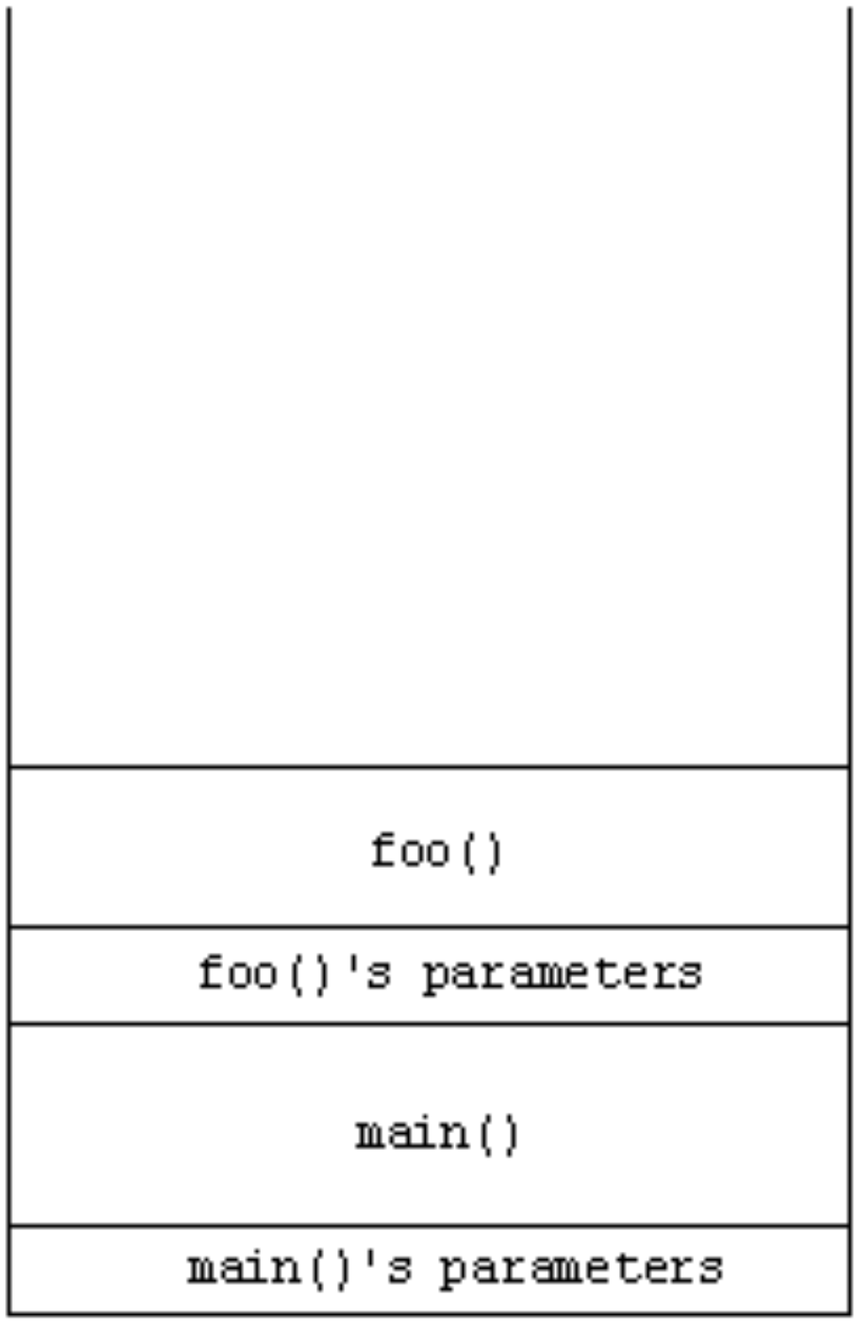
Scope, Global Variables

```
void
increment()
{
    x++;
}
```

see
buggy4.c, global.c, buggy5.c

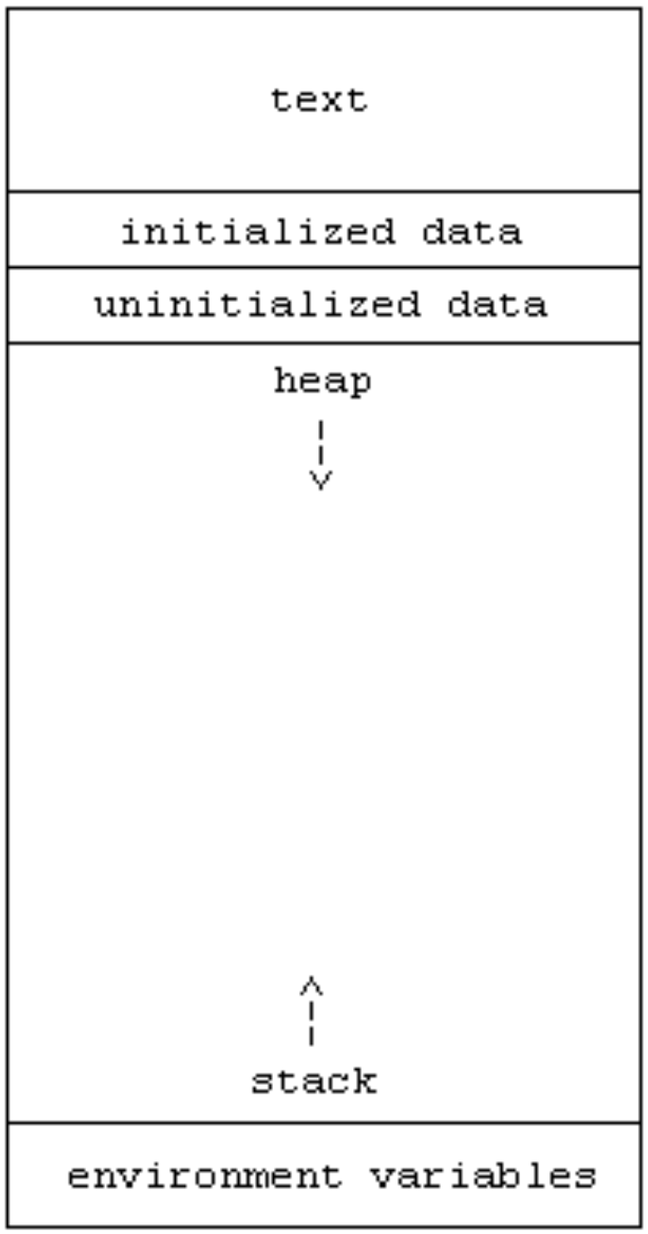
The Stack

Frames



Memory Management

Sneak Preview



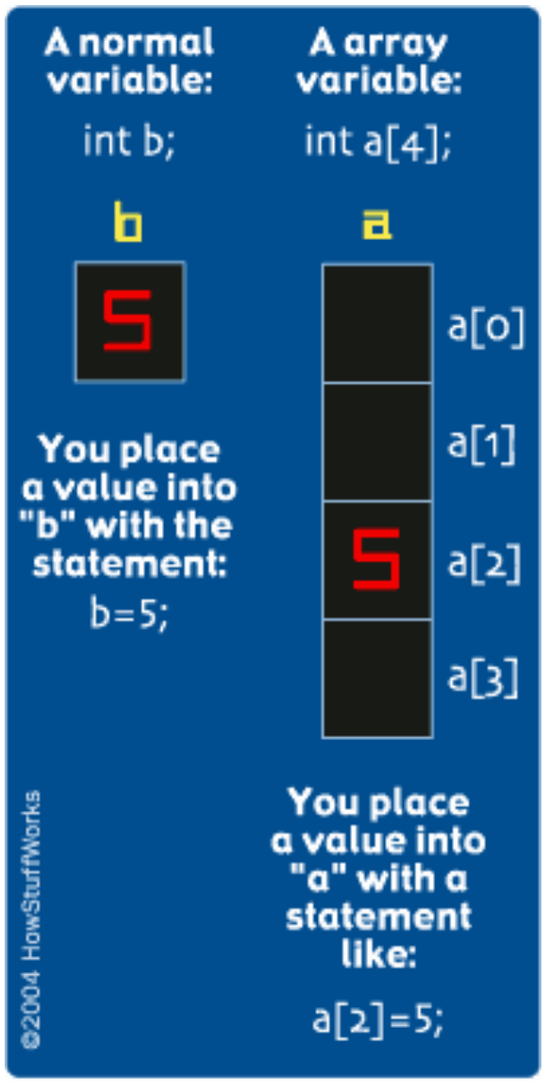
Functions

Return Values

```
int
cube(int a)
{
    return a * a * a;
}
```

see
return{1,2}.c

Arrays



see

array.c, buggy6.c, string{1,2}.c, capitalize.c

Image from http://computer.howstuffworks.com/c10.htm.

Free Resources

- http://www.howstuffworks.com/c.htm
- http://www.cs50.net/resources/cppreference.com/

Command-Line Arguments

argc, argv

```
int main(int argc, char *argv[]);
```

see
argv{1,2}.c

CS 50's Library

(Memory Leaks)

- `bool`
- `string`
- `char GetChar();`
- `double GetDouble();`
- `float GetFloat();`
- `int GetInt();`
- `long long GetLongLong();`
- `string GetString();`

see
http://www.cs50.net/pub/releases/cs50/

Cryptography

Or fher gb qevax lbhe binygvar!

Image from http://www.radioarchives.org/annie/.

Cryptography

Enigma Machine

Image from http://en.wikipedia.org/wiki/Enigma_machine.

Cryptography

Secret (Symmetric) Keys

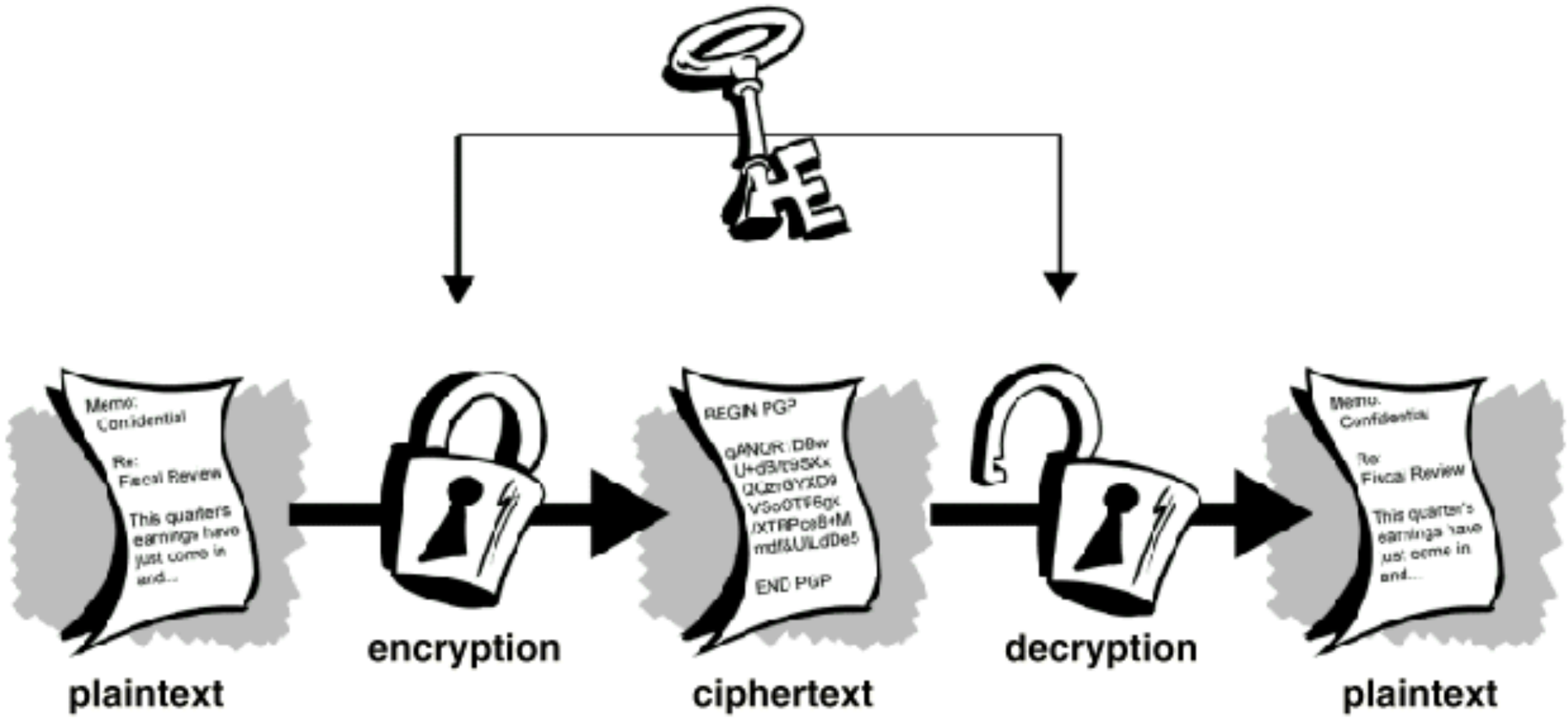


Image from http://www.nuitari.de/crypto.html.

Cryptography

Caesar Cipher

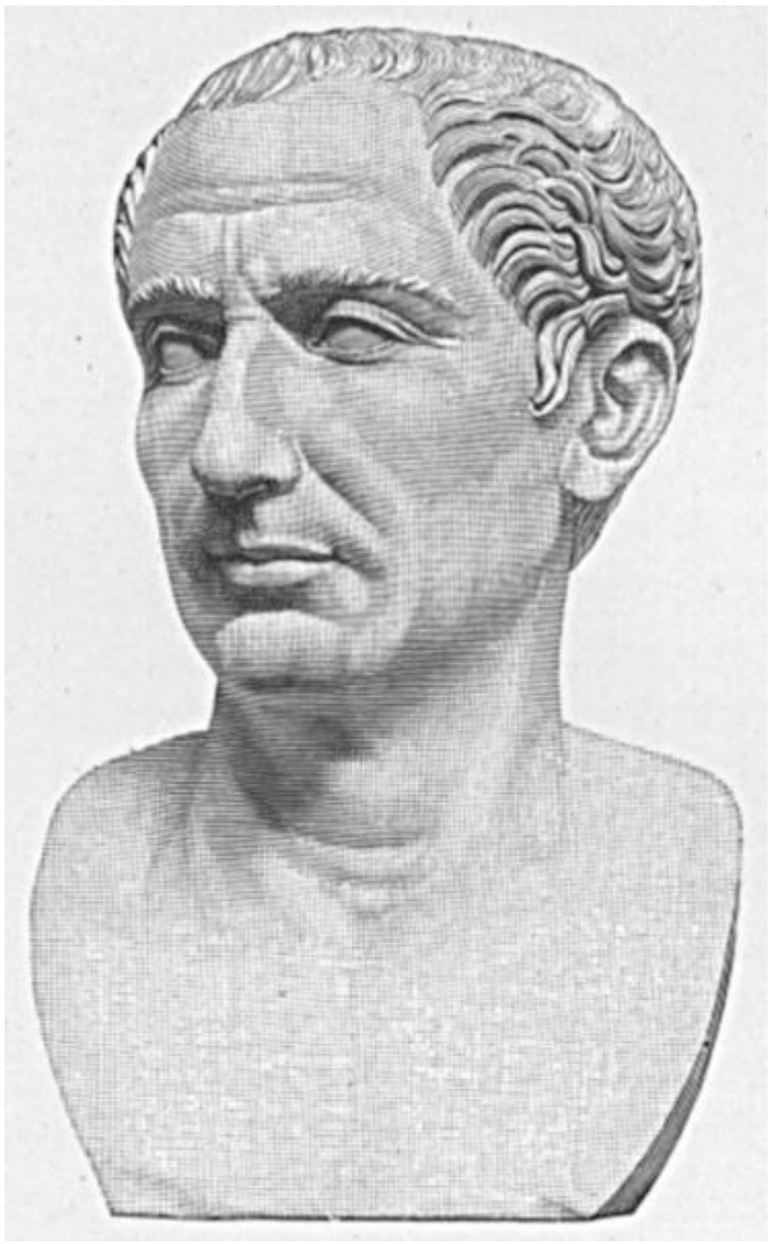


Image from http://commons.wikimedia.org/wiki/Image:Hw-caesar.jpg.

Cryptography

Caesar Cipher

$$c_i = (p_i + k) \% 26$$

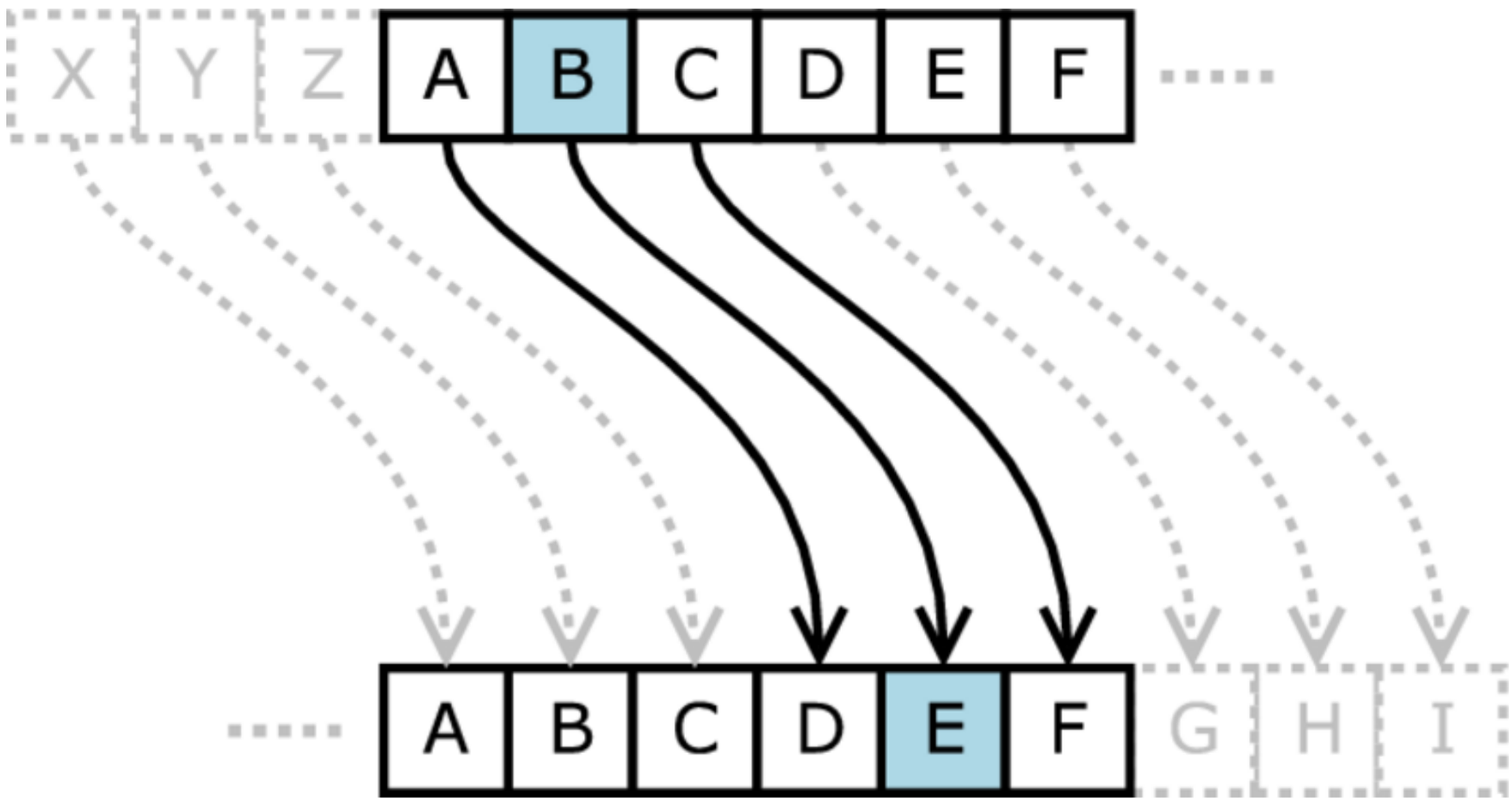


Image from http://en.wikipedia.org/wiki/Caesar_cipher.

Cryptography

Vigenère Cipher

$$c_i = (p_i + k_i) \% 26$$

p	H	E	L	L	O	,	W	O	R	L	D	
	+	+	+	+	+		+	+	+	+	+	
k	F	O	O	B	A		R	F	O	O	B	
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
c	M	S	Z	M	O	,	N	T	F	Z	E	

Cryptography

DES

72,057,594,037,927,936

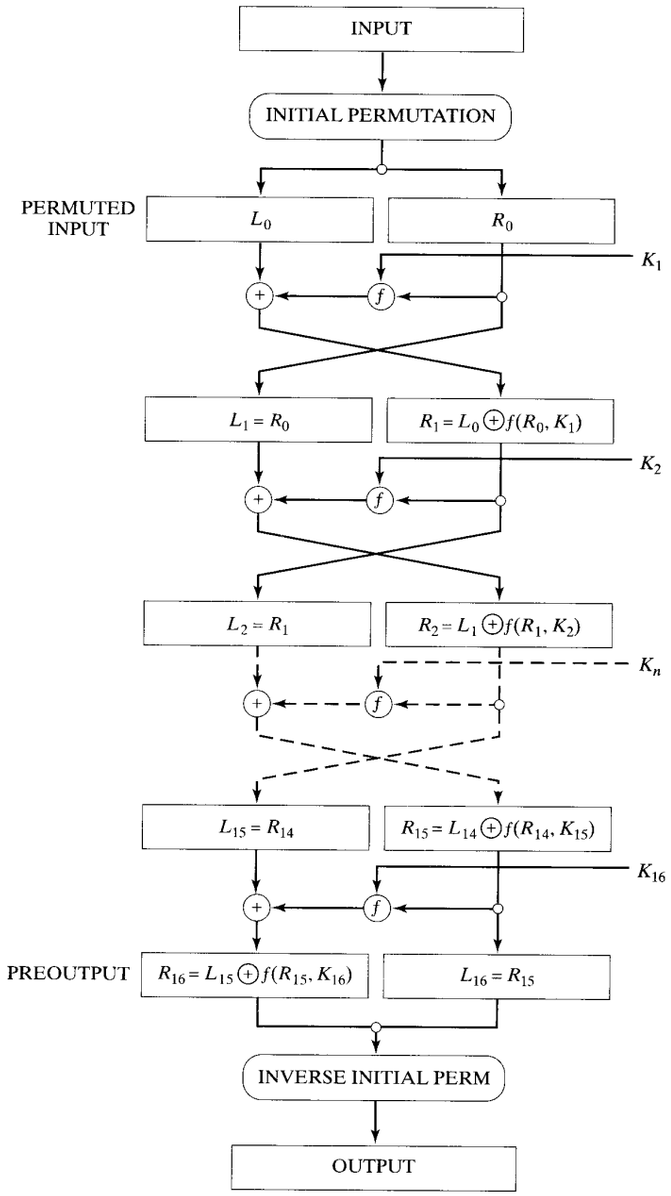


FIGURE 3–14 DES.

Figure from Larry Nyhoff's C++: An Introduction to Data Structures

Cryptography

Public and Private (Asymmetric) Keys

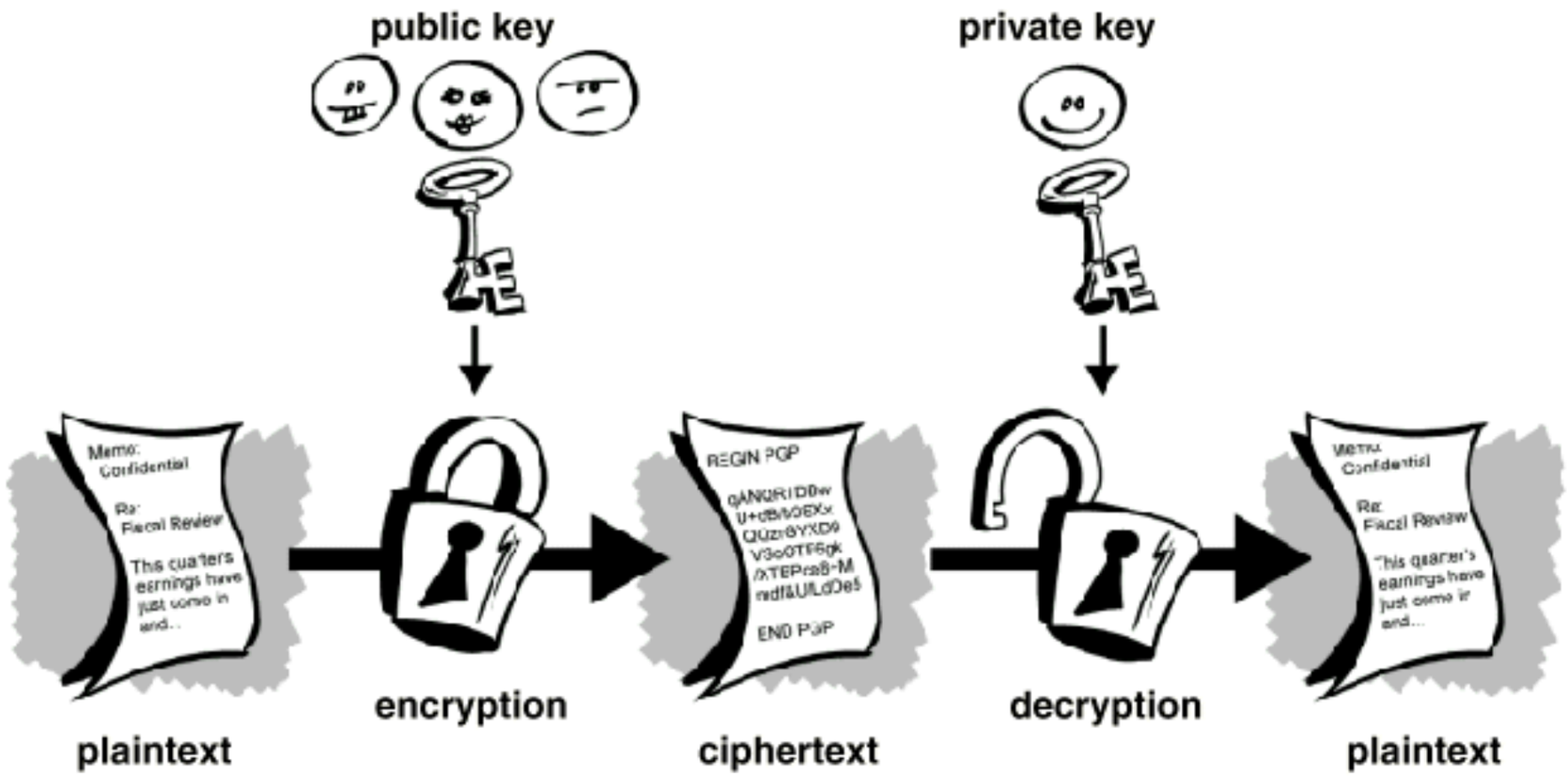


Figure from http://www.nuitari.de/crypto.html.

Cryptography

PGP

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: PGP Key Server 0.9.6

mQGiBEFMqoURBADaifCKsYkPBmVSMvspdLHLIAsb0xe4cyOieCA5LQCUZi9Z+Yxu
OSMkbQ+VSAvtP31/7o9pNf6LsLU4ADA5knZVB+GfZZpiGnEd62qKDKNpjVo20NRH
Xcd/4RpxE6aJWUWe2tPqLSCi3NFLPEhnfo5v9WyLRHjqdIQKc6vGAT41BQCg/5/1
vNpKhyA6VrFDIuozNWqKpAUD/1lAHxPxfxLyn8K/Gv/wl0y97dRDq0vsRqkh57IT
YKy/Xjv4qzNWZ/dSXI7Fa/6xULRuYK6tcr5aI6bFVLBl4fIXn5tapcCdYLAMo2ap
Uf/+PRJgSNUg4j50F5GjjiKco7FYldaF3oy6DVQzjEtSHN2TFczVOMHJUax1Ip/U
DRjIA/9vOO/MZ7FspAW1ZOdC13CxVSnig4oALGbNf76RViFGOl0bByVLVlBxMiIl
v8wxSbydqxsvokPZ/uCOfSqedO+l9xmIEp/Luq4k2owKfyAB2U33+HkfzS8RM4zJ
Wy1i8jXNzEfyFsqmJ0RKfrzJe7jXX34ZMfbPc3r39eR4w9lo+bQmUm9uYWxkIEwg
Uml2ZXN0IDxyaXZlc3RAY3NhaWwubWl0LmVkdT6JAEYEEBECAAYFAkFMquoACgkQ
5DGedS1eYN7UGACgzEZmCLhzzVz2kc3/5curi183AiMAn3NOJx6SJOL3n2fNAAar
7B5M0z9ZiQBGBBARAgAGBQJBTK1BAAoJEKXuoAZz/b3Wi9oAoPYpdchyMLydUjzh
GxiwYxQEZS8uAJ91BLfY5FIIGYLgHz/QkcUS+Ps2N4kAVAQQEQIAFAUCQUyqhQUJ
AmpPgAQLAwIBAhkBAAoJEIdenepUv6CUvGQAoKNCajxfdNc1/Lf73xvQLq//YBRt
AKCl5mvYi3D+w+4NikeXcA+tQe9korkEDQRBTKqFEBAA+RigfloGYXpDkJXcBWyH
huxh7M1FHw7Y4KN5xsncegus5D/jRpS2MEpT13wCFkiAtRXlKZmpnwd00//jocWW
IE6YZbjYDe4QXau2FxxR2FDKIldDKb6V6FYrOHhcC9v4TE3V46pGzPvOF+gqnRRh
44SpT9GDhKh5tu+Pp0NGCMbMHXdXJDhK4sTw6I4TZ5dOkhNh9tvrJQ4X/faY98h8
ebByHTh1+/bBc8SDESYrQ2DD4+jWCv2hKCYLrqmus2UPogBTAaB81qujEh76DyrO
H3SET8rzF/OkQOnX0ne2Qi0CNsEmy2henXyYCQqNfi3t5F159dSST5sYjvwqp0t8
MvZCV7cIfwgXcqK61qlC8wXo+VMROU+28W65Szgg2gGnVqMU6Y9AVfPQB8bLQ6mU
rfdMZIZJ+AyDvWXpF9Sh01D49Vlf3HZSTz09jdvOmeFXklnN/biudE/F/Ha8g8VH
MGHOfMlm/xX5u/2RXscBqtNbno2gpXI61Brwv0YAWCvl9Ij9WE5J280gtJ3kkQc2
azNsOA1FHQ98iLMcfFstjvbzySPAQ/ClWxiNjrtVjLhdONM0/XwXV0OjHRhs3jMh
LLUq/zzhsSlAGBGNfISnCnLWhsQDGcgHKXrKlQzZlp+r0ApQmwJG0wg9ZqRdQZ+c
fL2JSyIZJrqrol7DVes91hcAAgIP/0zPNiJsHsHyJL56YFfTmOTm2016zXaGErmM
b6Ej2VhXQEjjUAoV+HZ3odm2rVa0XR9F4RU7AFaIUedGmbET/Zp5uIT9CAuwODRq
wIaPdxXaS5HfEsDdwPC4rUigIg3wU7unWq8zKGy8gx+I0XgPvkUmdwb+vCZ0Zrl0
LC/SvyXyPNb87RAN1ttuDspFQ4/puUoxz/ICurJbBWx09oc29yyXiGX8YHfF6NFa
UCSJH5WlfS9uIQEdip6dmFB7Q2qvOYHLF5nAg2zXvg8LzWI3dcxH0OXHVy2KkG1E
bndUtq8cI8yz1+I6Pdfqb0DWvmIVVSHJMLtuZBUY1D8vsoZ2K0//PcNMuqHU8ZfH
CAXwmrJAfzYhU8TP6P4YKqa/W4Cxwy897yaaZHoR3iqhdDakMHrnDPAw4isGJ20j
PEXpzQ5H4i7PEqk+phVxiEhbLZbddz1y0ZK/5dub5ci5mCwGZBVb9XTecZruwOe7
ptWIvBvYhGBltUUfSF4wEwvoaxcC6EzFRpEqBRm+tgcgcfwU1V9oywoMhLQwB9LD
VjnNkRoNuaEa2o8CnheehNU05NSASsSo4z2WWbkRGErZZaWiafLe+XhDC+hImWwO
dL5ZatkQ5qJp3GuFW0FldqaYJLY1KNn9P+cpLhPEq5Hq27vcULDa1L5AMnKIbusS
SrRP9MhWiQBMBBgRAgAMBQJBTKqFBQkCak+AAAoJEIdenepUv6CUbCkAn13adk2J
HcZLgEhuNLZPTye4iNgRAKCtq+gBowVJ761YhVK2NMBi+8B3sw==
=j2zM
-----END PGP PUBLIC KEY BLOCK-----
```

Ron Rivest's public key from http://pgp.mit.edu/.

Cryptography

RSA

- Public Key: (e, n)
- Private Key: (d, n)
- To Encrypt:
  - $c = p^e \pmod n$
- To Decrypt:
  - $p = c^d \pmod n$

Ron Rivest's public key from http://pgp.mit.edu/.