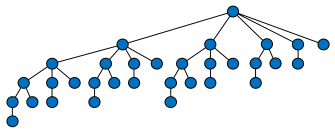
Lecture 11. Abstraction, Recursion, Induction, and How to Live Like a Computer Scientist

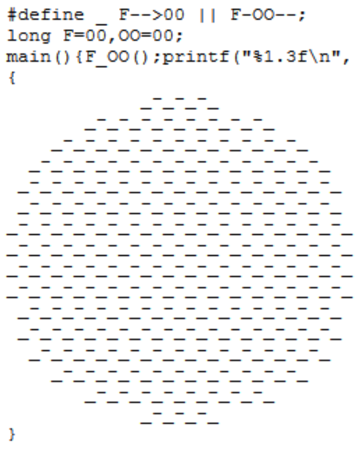
CpSc 2120: Algorithms and Data Structures
Brian C. Dean



School of Computing
Clemson University
Fall, 2020

Trivia / Show & Tell / Interesting Factoids

- The International Obfuscated C Code Contest



```
#define _ F-->00 || F-OO--;
long F=00,OO=00;
main(){F_OO();printf("%1.3f\n", 4.*-F/OO/OO);}F_OO()
{
            _-_-_-_
       _-_-_-_-_-_-_-_-_
    _-_-_-_-_-_-_-_-_-_-_-_
  _-_-_-_-_-_-_-_-_-_-_-_-_-_
 _-_-_-_-_-_-_-_-_-_-_-_-_-_-_
 _-_-_-_-_-_-_-_-_-_-_-_-_-_-_
_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_
_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_
_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_
_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_
 _-_-_-_-_-_-_-_-_-_-_-_-_-_-_
 _-_-_-_-_-_-_-_-_-_-_-_-_-_-_
  _-_-_-_-_-_-_-_-_-_-_-_-_-_
    _-_-_-_-_-_-_-_-_-_-_-_
       _-_-_-_-_-_-_-_-_
            _-_-_-_
}
```

Trivia / Show & Tell / Interesting Factoids

- The International Obfuscated C Code Contest

```
#define O(b,f,u,s,c,a)b(){int o=f();switch(*p++){X u:_ o s b();X c:_ o a b();default:p--;_ o;}}
#define t(e,d,_,C)X e:f=fopen(B+d,_);C;fclose(f)
#define U(y,z)while(p=Q(s,y))*p++=z,*p=' '
#define N for(i=0;i<11*R;i++)m[i]&&
#define I "%d %s\n",i,m[i]
#define X ;break;case
#define _ return
#define R 999
typedef char*A;int*C,E[R],L[R],M[R],P[R],l,i,j;char B[R],F[2];A m[12*R],malloc
(),p,q,x,y,z,s,d,f,fopen();A Q(s,o)A s,o;{for(x=s;*x;x++){for(y=x,z=o;*z&&*y==
*z;y++)z++;if(z>o&&!*z)_ x;}_   0;}main(){m[11*R]="E";while(puts("Ok"),gets(B)
)switch(*B){X'R':C=E;l=1;for(i=0;i<R;P[i++]=0);while(l){while(!(s=m[l]))l++;if
(!Q(s,"\"")){U("<>",'#');U("<=",'$');U(">=",'!');}d=B;while(*F=*s){*s=='"'&&j
++;if(j&1||!Q(" \t",F))*d++=*s;s++;}*d--=j=0;if(B[1]!='=')switch(*B){X'E':l=-1
X'R':B[2]!='M'&&(l=*--C)X'I':B[1]=='N'?gets(p=B),P[*d]=S():(*(q=Q(B,"TH"))=0,p
=B+2,S()&&(p=q+4,l=S()-1))X'P':B[5]=='"'?*d=0,puts(B+6):(p=B+5,printf("%d\n",S
()))X'G':p=B+4,B[2]=='S'&&(*C++=l,p++),l=S()-1 X'F':*(q=Q(B,"TO"))=0;p=B+5;P[i
=B[3]]=S();p=q+2;M[i]=S();L[i]=l X'N':++P[*d]<=M[*d]&&(l=L[*d]);}else p=B+2,P[
*B]=S();l++;}X'L':N printf(I)X'N':N free(m[i]),m[i]=0   X'B':_ 0 t('S',5,"w",N
fprintf(f,I))t('O',4,"r",while(fgets(B,R,f))(*Q(B,"\n")=0,G()))X 0:default:G()
;}_ 0;}G(){l=atoi(B);m[l]&&free(m[l]);(p=Q(B," "))?strcpy(m[l]=malloc(strlen(p
)),p+1):(m[l]=0,0);}O(S,J,'=',==,'#',!=)O(J,K,'<',<,'>',>)O(K,V,'$',<=,'!',>=)
O(V,W,'+',+,'-',-)O(W,Y,'*',*,'/',/)Y(){int o;_*p=='-'?p++,-Y():*p>='0'&&*p<=
'9'?strtol(p,&p,0):*p=='('?p++,o=S(),p++,o:P[*p++];}
```

Trivia / Factoids

```
                               char rahc
                                  [ ]
                                   =
                                 "\n/"
                                   ,
                               redivider
                                  [ ]
                                   =
                        "Able was I ere I saw elbA"
                                   ,
                                   *
                            deliver,reviled
                                   =
                                  1+1
                                   ,
                              niam ; main
                                  ( )
                                 {/*\}
                                  \*/
                                int tni
                                   =
                                  0x0
                                   ,
                            rahctup,putchar
                                  ( )
                          ,LACEDx0 = 0xDECAL,
                               rof ; for
                            (;(int) (tni);)
                              (int) (tni)
                         = reviled ; deliver =
                               redivider
                                   ;
for ((int)(tni)++,++reviled;reviled* *deliver;deliver++,++(int)(tni)) rof
                                   =
                            (int) -1- (tni)
                         ;reviled--;--deliver;
                            (tni)  =  (int)
                         - 0xDECAL + LACEDx0 -
                               rof ; for
       (reviled--,(int)--(tni);(int) (tni);(int)--(tni),--deliver)
                          rahctup = putchar
                          (reviled* *deliver)
                                   ;
                          rahctup * putchar
                           ((char) * (rahc))
                                   ;
                                  /*\
                                 {\*/}
```

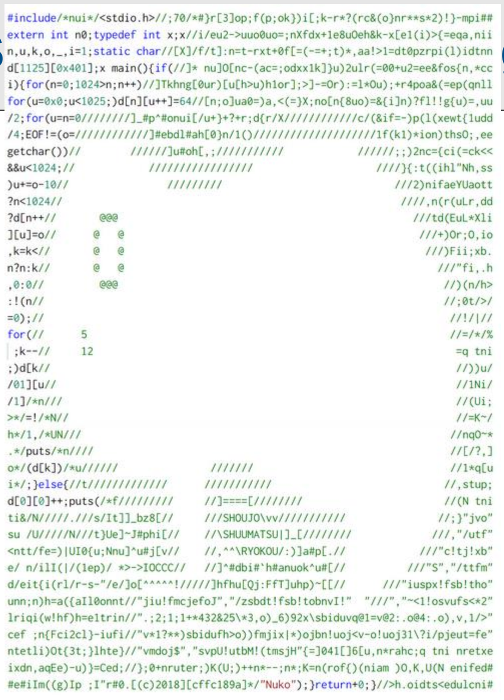
Trivia / Show & Tell / Interesting Factoids

```
#define DIT     (
#define DAH     )
#define __DAH   ++
#define DITDAH  *
#define DAHDIT  for
#define DIT_DAH malloc
#define DAH_DIT gets
#define _DAHDIT char
_DAHDIT _DAH_[]="ETIANMSURWDKGOHVFaLaPJBXCYZQb54a3d2f16g7c8a901?e'b.s;i,d:"
;main                   DIT                     DAH{_DAHDIT
DITDAH                  _DIT,DITDAH             DAH_,DITDAH DIT_,
DITDAH                  _DIT_,DITDAH            DIT_DAH DIT
DAH,DITDAH              DAH_DIT DIT             DAH;DAHDIT
DIT _DIT=DIT_DAH        DIT 81                  DAH,DIT_=_DIT
__DAH;_DIT==DAH_DIT     DIT _DIT                DAH;__DIT
DIT'\n'DAH DAH          DAHDIT DIT              DAH_=_DIT;DITDAH
DAH_;__DIT              DIT                     DITDAH
_DIT_?_DAH DIT          DITDAH                  DIT_ DAH:'?'DAH,__DIT
DIT' 'DAH,DAH_ __DAH    DAH DAHDIT              DIT
DITDAH                  DIT_=2,_DIT_=_DAH_;     DITDAH _DIT_&&DIT
DITDAH _DIT_!=DIT       DITDAH DAH_>='a'?       DITDAH
DAH_&223:DITDAH         DAH_ DAH DAH;           DIT
DITDAH                  DIT_ DAH __DAH,_DIT_    __DAH DAH
DITDAH DIT_+=           DIT DITDAH _DIT_>='a'?  DITDAH _DIT_-'a':0
DAH;}_DAH DIT DIT_      DAH{                    __DIT DIT
DIT_>3?_DAH             DIT                      DIT_>>1 DAH:'\0'DAH;return
DIT_&1?'-':'.';}__DIT DIT                       DIT_ DAH _DAHDIT
DIT_;{DIT void DAH write DIT                    1,&DIT_,1 DAH;}
************************************************
```

5

5

Trivia / S g Factoids



```
#include/*nui*/<stdio.h>//;70/*#}r[3]op;f(p;ok})i[;k-r*?(rc&(o}nr**s*2)!}-mpi##
extern int n0;typedef int x;x//i/eu2->uuo0uo=;nXfdx+1e8uOeh&k-x[e1(i)>{=eqa,nii
n,u,k,o,_,i=1;static char//[X]/f/t]:n=t-rxt+0f[=(-=+;t)*,aa!>1=dt0pzrpi(l)idtnn
d[1125][0x401];x main(){if(//]* nu]O[nc-(ac=;odxx1k]}u)2ulr(=00+u2=ee&fos{n,*cc
i){for(n=0;1024>n;n++)//]Tkhng[0ur)[u[h>u)h1or];>]-=Or):=1*Ou);+r4poa&(=ep(qnll
for(u=0x0;u<1025;)d[n][u++]=64//[n;o]ua0=)a,<(=}X;no[n{8uo)=&{i]n)?fl!!g{u)=,uu
/2;for(u=n=0////////]_#p^#onui[/u+}+?+r;d{r/X////////////c/(&if=-)p(l(xewt{1udd
/4;EOF!=(o=////////////]#ebdl#ah[0}n/1()/////////////////////1f(k1)*ion)thsO;,ee
getchar())//        //////]u#oh[,;////////////           //////;;)2nc={ci(=ck<<
&&u<1024;//            /////////////////                    ////}{:t((ihl"Nh,ss
)u+=o-10//                /////////                            ///2)nifaeYUaott
?n<1024//                                                       ////,n(r(uLr,dd
?d[n++//       @@@                                                ///td(EuL*Xli
][u]=o//      @   @                                                ///+)Or;O,io
,k=k<//       @   @                                                 ///)Fii;xb.
n?n:k//       @   @                                                   ///"fi,.h
,0:0//         @@@                                                     //)(n/h>
:!(n//                                                                 //;0t/>/
=0);//                                                                  //!/|//
for(//      5                                                           //=/*/%
 ;k--//     12                                                          =q tni
;)d[k//                                                                  //))u/
/01][u//                                                                 //1Ni/
/1]/*n///                                                                //(Ui;
>*/=!/*N//                                                               //=K~/
h*/1,/*UN///                                                            //nqO~*
.*/puts/*n////                                                          //[/?,]
o*/(d[k])/*u//////               ///////                                //1*q[u
i*/;}else{//t/////////////      ///////////                            //,stup;
d[0][0]++;puts(/*f/////////     //]====[////////                       //(N tni
ti&/N/////.///s/It]]_bz8[//     ///SHOUJO\vv///////////               //;}"jvo"
su /U/////N///t}Ue]~J#phi[//    //\SHUUMATSU|]_[////////             ///,"/utf"
<ntt/fe=)|UI0{u;Nnu]^u#j[v//    //,^^\RYOKOU/:)]a#p[.//            ///"c!tj!xb"
e/ n/ilI(|/(1ep)/ *>->IOCCC//   //]^#dbi#`h#anuok^u#[//         ///"S","/ttfm"
d/eit{i(rl/r-s-"/e/]o[^^^^^!/////]hfhu[Qj:FfT]uhp)~[[//      ///"iuspx!fsb!tho"
unn;n)h=a({aIl0onnt//"jiu!fmcjefoJ","/zsbdt!fsb!tobnvI!"  "///","~<1!osvufs<*2"
lriqi(w!hf)h=e1trin//".;2;1;1+*432&25\*3,o)_6)92x\sbiduvq@1=v@2:.o@4:.o),v,1/>"
cef ;n{Fci2cl}-iufi//"v*1?**)sbidufh>o))fmjix|*)ojbn!uoj<v-o!uoj31\?i/pjeut=fe"
ntetli)Ot{3t;}lhte}//"vmdoj$","svpU!utbM!(tmsjH"{=]041[]6[u,n*rahc;q tni nretxe
ixdn,aqEe)-u)}=Ced;//};0+nruter;)K(U;)++n*--;n*;K=n(rof{)(niam )O,K,U(N enifed#
#e#iIm((g)Ip ;I"r#0.[(c)2018][cffc189a]*/"Nuko");}return+0;}//>h.oidts<edulcni#
```

6

6

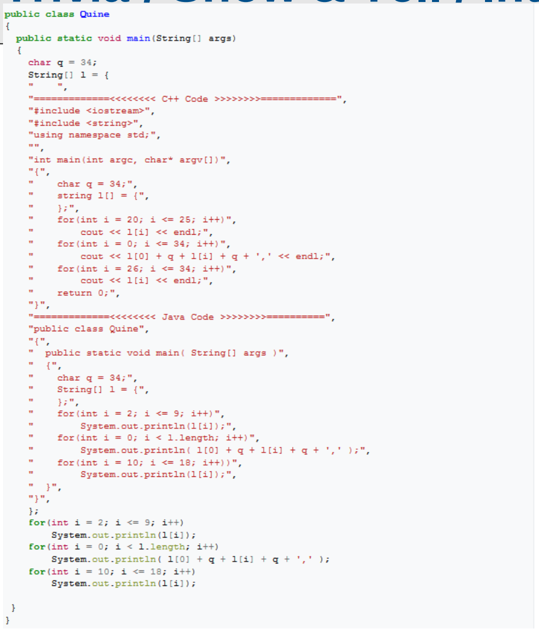
Trivia / Show & Tell / Interesting Factoids

- Smallest self-reproducing program:

7

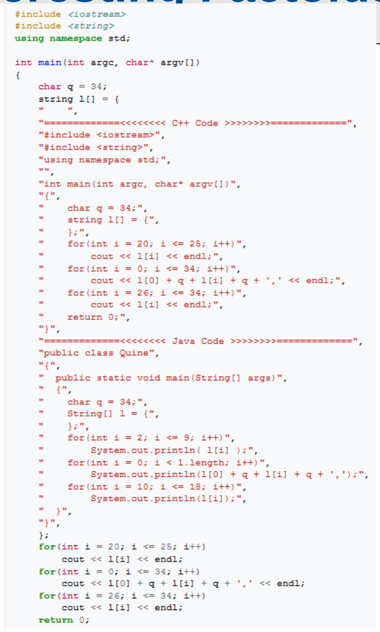
7

Trivia / Show & Tell / Interesting Factoids



```java
public class Quine
{
  public static void main(String[] args)
  {
    char q = 34;
    String[] l = {
    "    ",
    "=============<<<<<<<< C++ Code >>>>>>>>=============",
    "#include <iostream>",
    "#include <string>",
    "using namespace std;",
    "",
    "int main(int argc, char* argv[])",
    "{",
    "    char q = 34;",
    "    string l[] = {",
    "    };",
    "    for(int i = 20; i <= 25; i++)",
    "        cout << l[i] << endl;",
    "    for(int i = 0; i <= 34; i++)",
    "        cout << l[0] + q + l[i] + q + ',' << endl;",
    "    for(int i = 26; i <= 34; i++)",
    "        cout << l[i] << endl;",
    "    return 0;",
    "}",
    "=============<<<<<<<< Java Code >>>>>>>>==========",
    "public class Quine",
    "{",
    "  public static void main( String[] args )",
    "  {",
    "    char q = 34;",
    "    String[] l = {",
    "    };",
    "    for(int i = 2; i <= 9; i++)",
    "        System.out.println(l[i]);",
    "    for(int i = 0; i < l.length; i++)",
    "        System.out.println( l[0] + q + l[i] + q + ',' );",
    "    for(int i = 10; i <= 18; i++))",
    "        System.out.println(l[i]);",
    "  }",
    "}",
    };
    for(int i = 2; i <= 9; i++)
        System.out.println(l[i]);
    for(int i = 0; i < l.length; i++)
        System.out.println( l[0] + q + l[i] + q + ',' );
    for(int i = 10; i <= 18; i++)
        System.out.println(l[i]);

 }
}
```



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

int main(int argc, char* argv[])
{
    char q = 34;
    string l[] = {
    "    ",
    "=============<<<<<<<< C++ Code >>>>>>>>=============",
    "#include <iostream>",
    "#include <string>",
    "using namespace std;",
    "",
    "int main(int argc, char* argv[])",
    "{",
    "    char q = 34;",
    "    string l[] = {",
    "    };",
    "    for(int i = 20; i <= 25; i++)",
    "        cout << l[i] << endl;",
    "    for(int i = 0; i <= 34; i++)",
    "        cout << l[0] + q + l[i] + q + ',' << endl;",
    "    for(int i = 26; i <= 34; i++)",
    "        cout << l[i] << endl;",
    "    return 0;",
    "}",
    "=============<<<<<<<< Java Code >>>>>>>>=============",
    "public class Quine",
    "{",
    "  public static void main(String[] args)",
    "  {",
    "    char q = 34;",
    "    String[] l = {",
    "    };",
    "    for(int i = 2; i <= 9; i++)",
    "        System.out.println( l[i] );",
    "    for(int i = 0; i < l.length; i++)",
    "        System.out.println(l[0] + q + l[i] + q + ',');",
    "    for(int i = 10; i <= 18; i++)",
    "        System.out.println(l[i]);",
    "  }",
    "}",
    };
    for(int i = 20; i <= 25; i++)
        cout << l[i] << endl;
    for(int i = 0; i <= 34; i++)
        cout << l[0] + q + l[i] + q + ',' << endl;
    for(int i = 26; i <= 34; i++)
        cout << l[i] << endl;
    return 0;
```

8

8

Trivia / Show & Tell / Interesting Factoids

```
char*lie;

        double time, me= !0XFACE,

        not; int rested,   get, out;

        main(ly, die) char ly, **die ;{

            signed char lotte,


dear; (char)lotte--;

        for(get= !me;; not){

        1 -  out & out ;lie;{

        char lotte, my= dear,

        **let= !!me *!not+ ++die;

            (char*)(lie=
```

9

9

Abstraction

10

10

An Important Distinction…

Specification of a data structure in terms of the operations it needs to support.

(sometimes called an *abstract data type)*

A concrete approach for **implementation** of the data structure that fulfills these requirements.

Example: Queues

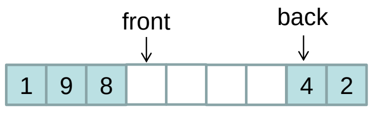
Abstract data type: queue

Must support these operations:

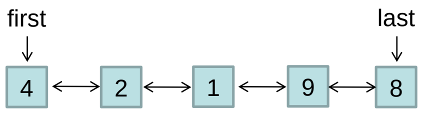
- *Insert* a new key *k* into the structure.
- *Remove* the least-recently-inserted key from the structure.

(so FIFO behavior)

Choices for concrete implementation:



Circular array

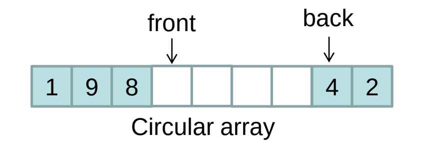


(Doubly) linked list

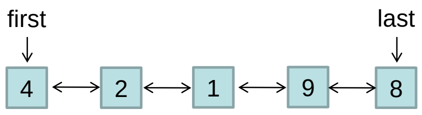
Which Implementation is Best?
(Right Answer is Often "It Depends…")

What factors would influence your decision of which underlying representation to use? (e.g., efficiency)

Choices for concrete implementation:



Circular array



(Doubly) linked list

13

Enforcing Abstraction in Code

Abstract data type: queue

Concrete implementation: queue.cpp

```
queue.h:

class Queue {
  private:
     int *A;
     int front, back, N;

  public:
     Queue();
     ~Queue();
     void insert(int key);
     int remove(void);
};
```

14

Enforcing Abstraction in Code

Abstract data type:
queue

Concrete implementation:
queue.cpp

```
queue.h:

class Queue {
private:
   int *A;
   int front, back, N;

public:
   Queue();
   ~Queue();
   void insert(i
   int remove(vo
};
```

```
Queue q;
q.insert(6);
x = q.remove();
```

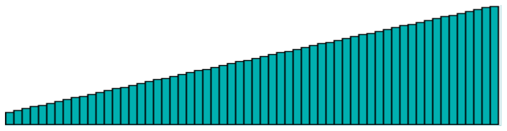
```
int Queue::remove(void)
{
   int result = A[back];
   back = (back+1) % N;
   return result;
}
```

Abstraction: Example

- Given N numbers, do two of them sum to 42?

Abstraction: Example

- Given N numbers, do two of them sum to 42?
- Step 1: Sort the numbers (we'll soon see how this can be done in O(N log N) time).



- Step 2: For each number X in the array, use binary search to see if 42 – X is also present in the array. Total time: N × O(log n) = O(N log N).

17

17

Abstraction: Example

```
main()
{
  xxxxx
  xxxxx
  xxxxx
  xxxxx

  yyyyy
  yyyyy
  yyyyy
  yyyyy
}
```

$\longrightarrow$

```
read_input()
{
  xxxxx
  xxxxx
  xxxxx
  xxxxx
}

write_output()
{
  yyyyy
  yyyyy
  yyyyy
  yyyyy
}

main()
{
  read_input();
  write_output();
}
```

18

18

Abstraction: Example

```
main()
{
 xxxxx
 xxxxx
 xxxxx
 xxxxx

 xxxxx
 xxxxx
 xxxxx
 xxxxx
}
```

$\longrightarrow$

```
do_something_complicated()
{
 xxxxx
 xxxxx
 xxxxx
 xxxxx
}

main()
{
 do_something_complicated();
 do_something_complicated();
}
```

Abstraction

- Don’t think about everything at once.
- Don’t worry about irrelevant low-level details when thinking about a high-level idea.

Abstraction

- Don’t think about everything at once.
- Don’t worry about irrelevant low-level details when thinking about a high-level idea.

- However, this doesn’t mean you can ignore the low-level details -- these may have a large impact on performance, and can sometimes cause unexpected behavior.
- It often helps to know as much of the technology “stack” as possible…

21

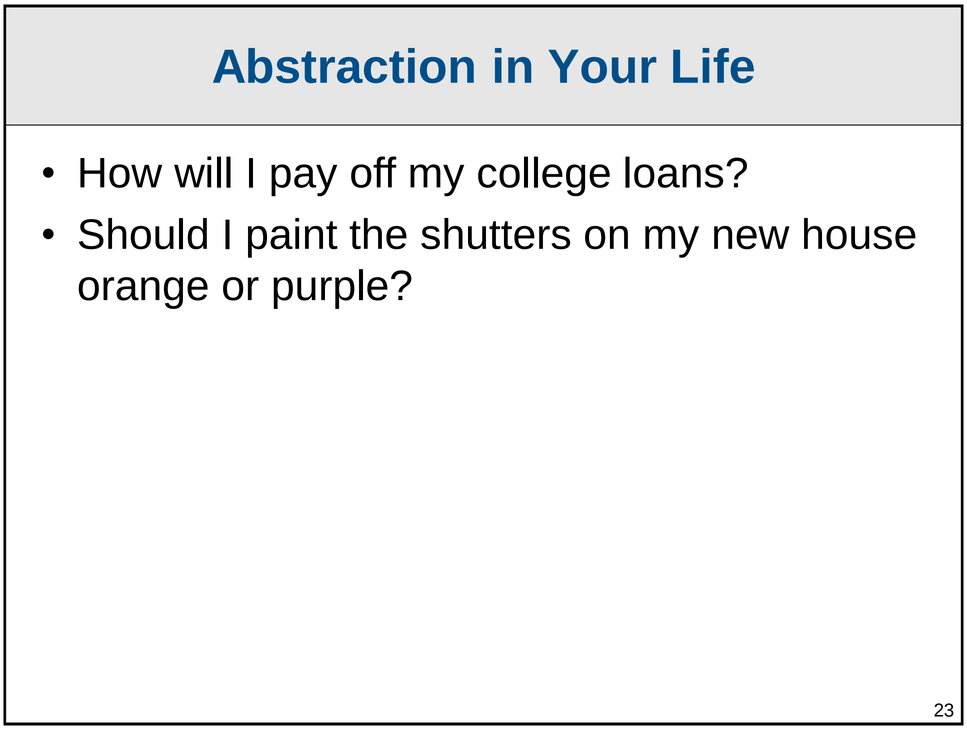
21

Abstraction in Your Life

- How will I pay off my college loans?

22

22

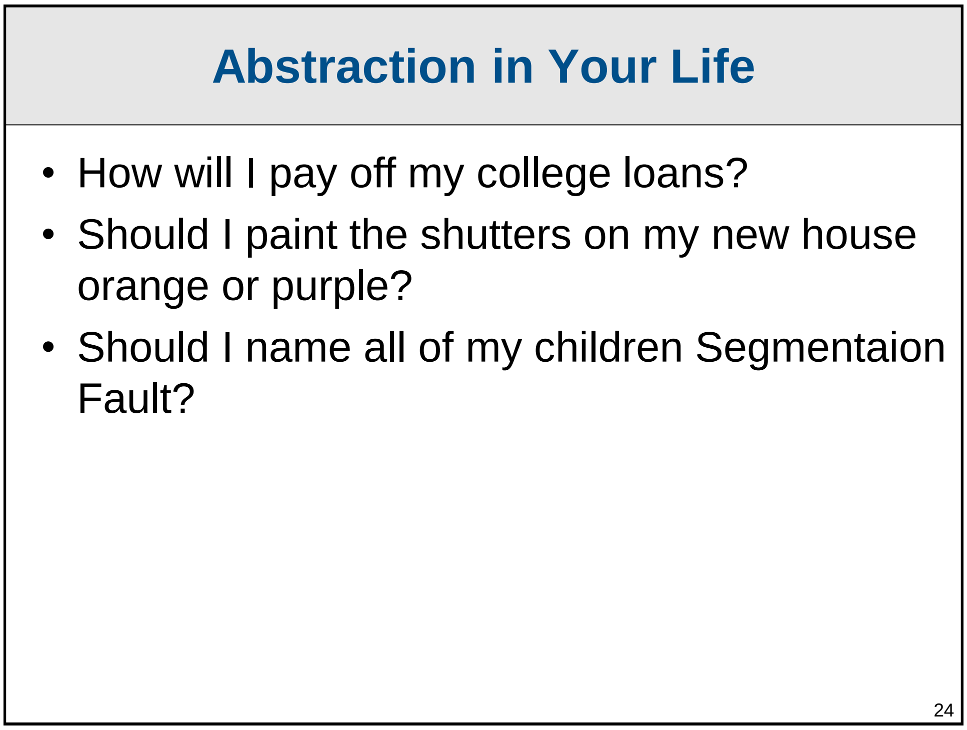


Abstraction in Your Life

- How will I pay off my college loans?
- Should I paint the shutters on my new house orange or purple?

23

23



Abstraction in Your Life

- How will I pay off my college loans?
- Should I paint the shutters on my new house orange or purple?
- Should I name all of my children Segmentaion Fault?

24

24

Revisiting our Interview Question…

- Given N strings, find all the duplicates.

25

25

Revisiting our Interview Question…

- Given N strings, find all the duplicates.
- Using set abstraction makes solution and running time analysis much easier!

```
Stringset entire_set, duplicates;
string s;
while (cin >> s) {
  if (entire_set.find(s) && !duplicates.find(s))
     duplicates.insert(s);
  entire_set.insert(s);
}
duplicates.print_contents();
```

26

26

Revisiting our Interview Question…

- Given N strings, find all the duplicates.
- The code below makes N…2N = O(N) calls to *find* and N…2N = O(N) calls to *insert*…

```
Stringset entire_set, duplicates;
string s;
while (cin >> s) {
  if (entire_set.find(s) && !duplicates.find(s))
     duplicates.insert(s);
  entire_set.insert(s);
}
duplicates.print_contents();
```

27

27

Recursion Fits in this Discussion Let's Practice Some Recursion…

- Add the contents of a linked list
- Copy a linked list
- Print a linked list backwards
- Insert into a linked list
  - Remember that Inserting a new node at the front of a linked list is super-easy (1 line of code!)
  - What about inserting a new node at the end of the list?
  - What about inserting so as to keep the list sorted?

28

28