

More C and Some C++

CS 1025 Computer Science Fundamentals I

Stephen M. Watt

University of Western Ontario

Exercise

- `man -s3c qsort`
- `#include <stdlib.h>`

```
void qsort( void    *base,
            size_t  nel,
            size_t  width,
            int (*compar)(const void *, const void *) );
```

- Write a C program that sorts `argv`

Note: `int strcmp(const char *s1, char *s2);`

A Solution

```
#include <stdlib.h>
#include <string.h>
```

```
int pstrcmp(const void *ps1, const void *ps2) {
    return strcmp(* (char **)ps1, * (char **)ps2);
}
```

```
int main(int argc, char **argv)
{
    int i;

    qsort(argv, argc, sizeof(*argv), pstrcmp);

    for (i = 0; i < argc; i++)
        printf("%d. %s\n", i, argv[i]);
    return EXIT_SUCCESS;
}
```

Good C/Bad C

- You have been encouraged to write clear C programs with **good** style.
- *The International Obfuscated C Code Contest* does the **opposite**!

International Obfuscated C Code Contest

Obfuscate: tr.v. -cated, -cating, -cates.

1. To render obscure. To darken.

2. To confuse: his emotions obfuscated his judgment.

[LLat. obfuscare, to darken : ob(intensive) +

Lat. fuscare, to darken < fuscus, dark.]

- obfuscation n. obfuscatory adj

Goals

- To write the most Obscure/Obfuscated C program under the rules.
- To show the importance of programming style, in an ironic way.
- To stress C compilers with unusual code.
- To illustrate some of the subtleties of the C language.
- To provide a safe forum for poor C code. :-)

Rules

1. Your entry must be a complete program.
2. The size of your program source must be ≤ 4096 bytes in length.
The number of characters excluding whitespace (tab, space, newline, formfeed, return), and excluding any ; { or } immediately followed by whitespace or end of file, must be ≤ 2048 .
6. Your program source must be able to be compiled cleanly by an ANSI C compiler, or if there are any compile errors, they must be documented in the "remarks" section of your submission.
12. Legal abuse of the rules is somewhat encouraged.
An entry that, in the opinion of the judges, violates the rules will be disqualified. Entries that attempt to abuse the rules must try to justify why their rule abuse is legal in the submitted "remarks".

Example of Entry

```
int i;main(){for(;i["]<i;++i){--i;}"];read('-'-'-'-'-,i+++ "hell\
o, world!\n",'/'/'/'/')));}read(j,i,p){write(j/p+p,i---j,i/i);}
```

Anonymous 1984

This prints “hello world!”

Understanding the program

```
int i;

main() {
    for( ; i["]<i;++i){--i;}";
        read('-'-'-'-' , i+++ "hello, world!\n", '/'/'/'/'')
    )
    ;
}

read(j,i,p) {
    write(j/p+p,i---j,i/i);
}
```

Understanding the program

```
int i;

main() {
    for( ; "]<i;++i){--i;}" [i];
        read('-'-'-', i+++ "hello, world!\n", '/'/'/'/'')
    )
    ;
}

read(j,i,p) {
    write(j/p+p,i---j,i/i);
}
```

Understanding the program

```
int i;

main() {
    for( ; "]<i;++i){--i;}" [i];
        read(0, i+++ "hello, world!\n", '/'/'/'/'')
    )
    ;
}

read(j,i,p) {
    write(j/p+p,i---j,i/i);
}
```

Understanding the program

```
int i;

main() {
    for( ; "]<i;++i){--i;}" [i];
        read(0, i+++ "hello, world!\n", 1)
    )
    ;
}

read(j,i,p) {
    write(j/p+p,i---j,i/i);
}
```

Understanding the program

```
int i;
```

```
main() {
```

```
    for( ; "]<i;++i){--i;}" [i]; )
```

```
        read(0, i+++ "hello, world!\n", 1);
```

```
}
```

```
read(j,i,p) {
```

```
    write(j/p+p,i---j,i/i);
```

```
}
```

Understanding the program

```
int i;
```

```
main() {
```

```
    for(i=0 ; "]<i;++i){--i;}" [i]; i++)
```

```
        read(0, i + "hello, world!\n",1);
```

```
}
```

```
read(j,i,p) {
```

```
    write(j/p+p,i---j,i/i);
```

```
}
```

Understanding the program

```
int i;
```

```
main() {
```

```
    for(i=0 ; "hello, world!\n"[i]; i++)
```

```
        read(0, i + "hello, world!\n",1);
```

```
}
```

```
read(j,i,p) {
```

```
    write(j/p+p,i---j,i/i);
```

```
}
```

Understanding the program

```
int i;  
char *s = "hello, world!\n";  
main() {  
    for(i=0 ; s[i]; i++)  
        read(0, i + s, 1);  
}  
  
read(j,i,p) {  
    write(j/p+p,i---j,i/i);  
}
```

Understanding the program

```
int i;  
char *s = "hello, world!\n";  
main() {  
    for(i=0 ; s[i]; i++)  
        myfun(0, i + s, 1);  
}  
  
myfun(zero, str, one) {  
    write(zero/one+one, str---zero, str/str) ;  
}
```

Understanding the program

```
int i;  
char *s = "hello, world!\n";  
main() {  
    for(i=0 ; s[i]; i++)  
        myfun(0, i + s,1);  
}  
  
myfun(zero, str, one) {  
    write(1, str--,1);  
}
```

Understanding the program

```
int i;  
char *s = "hello, world!\n";  
main() {  
    for(i=0 ; s[i]; i++)  
        myfun(0, i + s, 1);  
}  
  
myfun(zero, str, one) {  
    write(1, str, 1);  
}
```

Understanding the program

```
int i;  
char *s = "hello, world!\n";  
  
main() {  
    for(i=0 ; s[i]; i++)  
        write(1, s+i,1);  
}
```

An Othello Game

```
#define Y return
#define R for
#define e while
#define I printf
#define l int
#define W if
#define C y=v+111;H(x,v)*y++= *x
#define H(a,b)R(a=b+11;a<b+89;a++)
#define s(a)t=scanf("%d",&a)
#define U Z I
#define Z I("123\
45678\n");H(x,V){putchar(".XO"[*x]);W((x-V)%10==8){x+=2;I("%d\n",(x-V)/10-1);}}
l V[1600],u,r[]={-1,-11,-10,-9,1,11,10,9},h[]={11,18,81,88},ih[]={22,27,72,77},
bz,lv=60,*x,*y,m,t;S(d,v,f,_,a,b)l*v;{l c=0,*n=v+100,j=d<u-1?a:-9000,w,z,i,g,q=
3-f;W(d>u){R(w=i=0;i<4;i++)w+=(m=v[h[i]])==f?300:m==q?-300:(t=v[ih[i]])==f?-50:
t==q?50:0;Y w;}H(z,0){W(E(v,z,f,100)){c++;w=-S(d+1,n,q,0,-b,-j);W(w>j){g=bz=z;
j=w;W(w>=b||w>=8003)Y w;}}}W(!c){g=0;W(_){H(x,v)c+= *x==f?1:*x==3-f?-1:0;Y c>0?
8000+c:c-8000;}}C;j=-S(d+1,n,q,1,-b,-j);bz=g;Y d>=u-1?j+(c<<3):j;}main(){R(;t<
1600;t+=100)R(m=0;m<100;m++)V[t+m]=m<11||m>88||(m+1)%10<2?3:0;I("Level:");V[44]
=V[55]=1;V[45]=V[54]=2;s(u);e(lv>0){Z do{I("You:");s(m);}e(!E(V,m,2,0)&&m!=99);
W(m!=99)lv--;W(lv<15&&u<10)u+=2;U("Wait\n");I("Value:%d\n",S(0,V,1,0,-9000,9000
));I("move: %d\n",(lv-=E(V,bz,1,0),bz));}}E(v,z,f,o)l*v;{l*j,q=3-f,g=0,i,w,*k=v
+z;W(*k==0)R(i=7;i>=0;i--){j=k+(w=r[i]);e(*j==q)j+=w;W(*j==f&&j-w!=k){W(!g){g=1
;C;}e(j!=k)*(j-=w)+o=f;}}Y g;}
```

Roemer B. Lievaart 1987

An Amazing Program

Nick Johnson 2004

```
#include <ncurses.h>/*****/
int m[256] [ 256 ], a
, b ;;; WINDOW*w; char*l="" "\176qx1" "q" "q" "w\
xm" "x" "t" "j" "v" "u" "n" ,Q[
]= "Z" "pt!ftd\" "qdc!\eu" "dq!$c!nnwf"/** *** */"t\040\t";c(
int u , int v){ v?m [u] [v-
1] |=2,m[u] [v-1] & 48?W [v-1] & 15]]):0:0;u?m[u -1] [v] |=1 ,m[
u- 1] [ v]& 48? W-1 ] [v ]&
15] ]):0:0;v< 255 ?m[ u] [v+1] |=8,m[u] [v+1]& 48? W [ v+1]&15]
):0 :0; u < 255 ?m[ u+1 ] [v ] |=
4,m[u+1] [ v]&48?W+1] [v]&15]]):0:0;W [ v]& 15] ]);}cu(char*q){ return
*q ?cu (q+ 1)& 1?q [0] ++:
q[0 ]-- :1; }d( int u , int/**/v, int y){ int
Y=y -v, X=x -u; int S,s ;Y< 0?Y =-Y ,s,
s=- 1:( s=1);X<0?X=-X,S =-1 :(S= 1); Y<= 1;X<=1; if(X>Y){
int f=Y -(X >>1 );; while(u!= x){
f>= 0?v+=s,f-=X:0;u +=S ;f+= Y;m[u] [v] |=32;mvwaddch(w,v ,u, m[u
] [ v]& 64? 46) ;if (m[ u] [
v]&16){c(u,v);; ;;; ;;; return;}} }else{int f=X -(Y>>1);; while
(v !=y ) {f >=0 ?u +=S, f-= Y:0
;v +=s ;f+=X;m[u] [v] |= 32;mvwaddch(w,v ,u,m[u] [v]&64?60:46);if(m[u
] [ v]& 16) {c( u,v );
; return;};}}Z( int/**/a, int b){ }e( int/**/y,int/**/ x){
int i ; for (i= a;i <=a
+S;i++)d(y,x,i,b),d(y,x,i,b+L);for(i=b;i<=b+L;i++)d(y,x,a,i),d(y,x,a+ S,i
); ;;; ;;; ;;; ;;; ;
mvwaddch(w,x,y,64); ;;; ;;; prefresh( w,b,a,0,0 ,L- 1,S-1
);} main( int V , char *C[
] ) {FILE*f= fopen(V==1?"arachnid.c"/**/ :C[ 1],"r");int/**/x,y,c,
v=0 ;;; initscr (); Z(Z (raw
() ,Z( curs_set(0),Z(1 ,noecho()))),keypad( stdscr,TRUE));w =newpad
( 300, 300 ) ; for (x= 255 ; x >=0 ;x--
) for (y= 255 ;y>=0;y-- )m[ x] [ y]= 0;x=y=0;refresh( );while
( (c= fgetc (f) )+1) {if(
0||c==10|| x== 256){x=0;y++;if(y==256 )break;}} else{m[x] [y]=(c ==
'~' ?64 : c ==32 ?0: 16) ;;;x ++;
}}for(x=0 ;x< 256;x++)m [x] [0]=16 ,m[ x] [ 255]=16;for(y=0
;y< 256 ; y ++ ) m[0 ] [y ] = 16,
m[255] [y] =16 ;a=b=c=0; x=y =1; do{v++;mvwaddch (w, y,x ,m[
x] [ y]& 32? m[x ] [y ] & 16?
0| acs_map[1[m[x] [y]&15]]:46 : 32);c==0163&&!(m[x] [y+1]&16)?y++: 0;c
== 119 &&!( m[
y- 1]& 16) ?y--:0;c ==97 &&!(m[x-1] [y]&16)?x--:0;c==100&&!(m[x+1
] [ y]& 16) ? x ++:0 ;if(
3- 1+1 ) {endwin( );; return(0) ;}x -a<5?a>S- 5?a-=S-5:(a=0):
0;x -a> S-5?a<255 -S* 2?a +=S
-5:(a=256-S):0; y-b<5?b>L-5?b-=L-5:(b =0) :0; y-b>L-5?b<255-L *2?
b+= L-5 : (b =256
-L) :0;e(x,y);if(m[x] [y]&64)break;}while( (c=getch())!=-1);endwin();cu(Q);
printf(Q,v);}
```

For More...

- <http://www.ioccc.org>

A Preview of C++

- Started as ``C with classes''
- Started in 1983 by Bjarne Stroustrup
<http://www.research.att.com/~bs/homepage.html>
- First commercial release 1985
- Standardized 1997
- Non standard extensions: g++, MS Visual C++

Could Put Function Pointers in C Structs

- ```
typedef struct rect {
 int x, y;
 int (*area) (struct rect *);
} rect_t;
```
- ```
int rect_area(rect_t *pr) {  
    return (pr->x) * (pr->y);  
}
```
- ```
rect_t *rect_new(int x, int y) {
 rect_t *pr = (rect_t *) malloc(sizeof(rect_t));
 pr->x = x; pr->y = y;
 pr->area = rect_area;
 return pr;
}
```

# Then Use As...

```
#include <stdio.h>
```

```
#include "rect.h"
```

```
int main() {
```

```
 rect_t *pr = rect_new(10,20);
```

```
 printf("The area is %d\n", pr->area(pr));
```

```
 return 0;
```

```
}
```

# In C++

```
#include <stdio.h>
```

```
struct rect_t {
 int x, y;
 int area() { return x*y; }
 rect_t(int x0, int y0) { x = x0; y = y0; }
};
```

```
int main() {
 rect_t *pr = new rect_t(10, 20);
 printf("The area is %d\n", pr.area());
 return 0;
}
```

# In Better C++

```
#include <iostream.h>
```

```
struct rect_t {
 int x, y;
 int area() { return x*y; }
 rect_t(int x0, int y0) { x = x0; y = y0; }
};
```

```
int main() {
 rect_t *pr = new rect_t(10, 20);
 cout << "The area is " << pr.area() << endl;
 return 0;
}
```

# Structs vs Classes

- Fields can be declared **private** or **public**.
- The fields of structs are visible unless declared private.
- For classes it's the other way around.

```
class Rectangle {
 private:
 int w, a;
 public:
 Rectangle(int w0, int h0) { w=w0; a=w0*h0; }

 int width() { return w; }
 int height() { return a/w; }
};
```

# And so on

- Inheritance, overloading, exceptions, templates, ...