Burt Rosenberg

Best Game

Chris Mills
Wyse Technology
3471 N. 1st St.
San Jose CA 95134

Judges notes:

```
Compile notes:

    cc -DM=500 -DN=52 -DX="srand((int)time(0L))" -DV=void -o cmills cmills.c
```

```
-DM=<max bet allowed>
-DN=<52 * number of decks used>
-DX=<how to seed the rand() generator>
-DV=<void ot int>
```

```
Usage: cmills [starting_cash]
```

Selected notes from the author:

This is a blackjack program. It does splits, doubling down, insurance, etc.

As best I can figure, it should be ANSI-compliant, as well as portable to BSD and sysV (as well as the irritating compiler I was using which assumes `x=*p++` means `x*=p++` !). It is reasonably robust in its error handling—it won't let you split, double, etc. unless you actually have the cash, etc. To terminate the program, hit EOF or type 0 or a negative number for you wager.

The way it deals with splits is particularly cool (IMHO). You can split many times, and it will show all hands as they are played.

```
#include<stdio.h>
#include<time.h>
#define S(q)B(*e=0),q,p);
#define W(w)if((w)<22)v= -v;else v-=10;else
#define F for(r=d;r<d+N;
#define H(v,g)v=h(v,*g++= *r++),*g=0;
#define B(n)b(l,n),(V)printf(l
V
exit();char*g,c[M],d[N],q[N],p[N],*e,*r=d+N,l[M],*t="??",*k,
*m="DEATHISDSev2oinpohae3ttsluuln4a?uibfe 5l\0rtlfrb6 ?a?el:e"
"7$!n\0?e t8%ccp\0.%s9deelc.s T.@?-t.\t<J /\0=a\nP=Q Sex \0l KW Sin"
" a$\0ane-lay% ge#-slediefuk ar  r$via-:o ?+}:r? n \0:) ee%lon"
"e 1-esy666!-~v\n.!^‘’~@#\0\np~===:=q";
b(o,n)char*o;{for(k=n+m;*o++=*k;k+=9);}int
y=M*2,v,x,s,f,j,o;u(s){B(s));if(!gets(l))exit(0);return(o=
*l)==’y’||o==’Y’||o!=’n’&&o!=’N’&&u(s);}h(v,m){for(k=c;*k!=’J’;)
if(m==*k++)break;m=k-c;if(v<0)W(v=m-v)
if(m==1)W(v+=11)
v+=m;return
v;}main(w,z)char**z;{b(c,2)*X;for(--w&&(y=atoi(1[z]));y>1;)
{if(r-d>N*3/4){B(8));F++r)*r=c[(r-d)%13];F)w=
*(g=d+rand()%N),*g=
*r,*r++=w;r=d;}for(;;){B(3),y);if(!gets(l)||(w=atoi(l))<1)exit(
0);if(w&1||w>M||w>y)B(1),y<M?y:M);else
break;}y-=w,s=f=j=x=v=0,g=q,e=p;H(x,g)H(x,g)H(v,e)H(v,e)*t=
*q;S(t)*q==’A’&&y>=w/2&&u(5)&&(y+=(3*(h(0,1[q])==10)-1)*w/2);if(x==-21)goto
_;if(v==-21){y+=w/2;goto
_;}while(x>-17&&x<17)H(x,g)while((v==20||*p==-1[e])&&y>=w&&u(6
)){y-=w;++s;for(g=e++;2[g]=
*g,g!=p;)--g;*g++=
*e;*g=’ ’;i:v=h(h(0,*p),*e++=
*r++);S(t)if(*p==’A’&&-1[e]!=
*p)goto
_;}if(f=y>=w&&u(7))y-=w,H(v,e)while(!f&&v<22&&u(4)){H(v
,e)if(v<22)S(t)}_:x<0&&(x=
-x);v<0&&(v=
-v);if(v<22)if(v==x)j+=w*++f;else
if(x>21||v>x)j+=w*2*++f;if(s--){*e++=’ ’;*e++=
*p;*e=0;for(e=p;*e=2[e];)e++;goto
i;}y+=j;S(q)};}
```

Best Language Tool

Diomidis Spinellis
Imperial College, University of London
Myrsinis 1
GR-145 62 Kifissia
Greece

Judges’ comments:

Compile and run. Give the following lines as input:

```
OLD LANDER.BAS
RUN
```

Be sure not to run out of fuel!

Selected notes from the author:

This is the DDS-BASIC Interpreter (Version 1.00).
Immediate commands:

```
RUN LIST NEW OLD filename
BYE SAVE filename
```

Program commands:

```
variable names a to z variables initialized to 0 on RUN
FOR var = exp TO exp NEXT variable
GOSUB exp RETURN
GOTO exp IF exp THEN exp
INPUT variable PRINT string
PRINT exp var = exp
REM any text END
```

Expressions (ranked by precedence):

```
Bracketed expressions
number (leading 0 for octal, 0x for hex, else decimal), variable
Unary -
* /
+ -
= <>
> <
<= >=
(* and + are also used for boolean AND and boolean OR)
Boolean expressions evaluate to 0 for false and 1 for true
```

Editing:

- Line editor using line re-entry.
- A line number with nothing following it, deletes the line.

Input format:

- Free format positioning of tokens on the line.
- No space is allowed before the line number.
- Exactly one space is needed between the OLD or SAVE command and the filename.
- ALL INPUT MUST BE UPPERCASE.

Limits:

- Line numbers 1-10000
- Line length 999 characters
- FOR nesting 26
- GOSUB 999 levels
- Program Dynamically allocated
- Expressions -32768 - 32767 for 16 bit machines, -2147483648 - 2147483647 for 32 bit machines

Error checking / error reports:

- No error checking is performed.
- The message ‘core dumped’ signifies a syntax or semantic error.

Hosting Environment:

- ANSI C, traditional K&R C or OSF ANDF
- ASCII or EBCIDIC character set
- 48K bytes memory

The Speed of DDS-BASIC Interpreter (Version 1.00). relative to Microsoft Advanced BASIC 3.31 is Approximately 60

The code size could be further reduced by doing ugly things like not declaring the return type of functions, not freeing memory, #defining define, and joining lines. In its present 1536 character form the program is reasonably portable (it may fail to run in a tagged object architecture) and nicely formatted (it fits in an 80*25 screen).

```
#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--
#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++];}
```

Most Unusual Data Structure

Peter J Ruczynski.
Pyramid Technology Ltd.
Pyramid House,
Solatron Road
Farnborough. Hants GU14 7PL
England, U.K.

Judges' comments:

This program prints out a string. Can you figure out how it is done?

By use of pointers to functions returning a struct of pointers to functions!

Note that some compilers will be unable to compile the expression 'X=g()...' in main due to lack of temporary value space. One might want to try replacing main with:

```
main(){X=s().v().o().o().l().S().d().l().i().o().w().N();}
```

if this is a problem.

```
#include <stdio.h>
#define A(a) G a();
#define B(a) G (*a)();
#define C(a,b) G a() { printf(b); return X; }
typedef struct F G;A(a)A(b)A(c)A(d)A(e)A(f)A(g)A(h)A(i)A(j)A(k)A(l)A(m)A(n)A(
o)A(p)A(q)A(r)A(s)A(t)A(u)A(v)A(w)A(x)A(y)A(z)A(S)A(N)void Q();struct F{B(a)B
(b)B(c)B(d)B(e)B(f)B(g)B(h)B(i)B(j)B(k)B(l)B(m)B(n)B(o)B(p)B(q)B(r)B(s)B(t)B(
u)B(v)B(w)B(x)B(y)B(z)B(S)B(N)void(*Q)();}X={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,S,N,Q};C(a,"z")C(b,"y")C(c,"x")C(d,"w")C(e,"v")C(f,"u")C(
g,"t")C(h,"s")C(i,"r")C(j,"q")C(k,"p")C(l,"o")C(m,"n")C(n,"m")C(o,"l")C(p,"k"
)C(q,"j")C(r,"i")C(s,"h")C(t,"g")C(u,"f")C(v,"e")C(w,"d")C(x,"c")C(y,"b")C(z,
"a")C(S," ")C(N,"\n") void Q(){}main(){X=g().s().v().S().j().f().r().x().p().
S().y().i().l().d().m().S().u().l().c().S().q().f().n().k().v().w().S().l().e
().v().i().S().g().s().v().S().o().z().a().b().S().w().l().t().N();}
```

ANSI Committee's Worst Abuse of C

Larry Jones
SDRC
2000 Eastman Dr
Milford, OH 45150

Judges' comments:

The addition of trigraphs has made it easier to make programs hard to read. It was a dark and stormy night for C when the ANSI C committee added trigraphs to the standard. Too bad that they didn't just simply require a more reasonable ISO character set.

The common.mk (non-ANSI) makefile uses sed to form scjones.alt.c. The scjones.alt.c is similar to the original scjones.c source, but with all the trigraphs converted. You might want to try:

```
make -f common.mk scjones.alt.c
```

even if you only have an ANSI compiler, just to see how the trigrapsh are converted.

WARNING: In the future, we will be passing ANSI programs through a trigraph filter before reading. Trigraph-ifacation will not be a major consideration in future contests. People should exploit more interesting features of ANSI C.

Selected notes from the author:

Since this year's contest is dedicated to ANSI C, here is a strictly conforming entry. In accordance with the ANSI definition of a strictly conforming program it does not produce output dependent on any unspecified, undefined, or implementation defined behavior, and it does not exceed any of the minimum implementation limits. It adheres to all of the syntactic and semantic constraints of the standard, #includes the relevant headers for the functions it uses, and uses the EXIT_SUCCESS and EXIT_FAILURE macros to return a correct success/fail status to the invoking program. It also uses trigraphs for characters which are not in the Invariant Code Set of ISO 646:1983, so it will be portable to machines with deficient (from C's standpoint) character sets. In short, it should be very widely portable. As an extra added attraction, all of the lines are under 72 characters long, which would allow for sequence numbers to be punched in columns 73-80 :-).

[[The Author sent us this note after he was informed that he won]]

Although I am quite grateful for the ”ANSI Committee’s Worst Abuse of C” title (were you aware that I am a member of the committee or is the double meaning unintentional?), I was really hoping for something more along the lines of ”Closest Resemblance to Line Noise.” :-)

[[... it was unintentional]]

```
char*a??(??)=??<
"??=include<stdio.h>",
"??=include<stdlib.h>",
"??=define o stdout",
"??=define b break;case",
"??=define s(p)fputs(p,o);",
"??=define c(p)fputc(p,o);",
"void t(p,f)char*p;??<f&&c(’??/"’)",
"for(;;p++)??<switch(*p)??<case 0:f&&",
"s(??/"??/??/??/",??/")c(’??/??/n’)return;case",
"’??=’:s(??/"???/??/?=??/")b’??<’:s(??/"???/??/?<??/")",
"b’??>’:s(??/"???/??/?>??/")b’??(’:s(??/"???/??/?(??/")b’??)’",
":s(??/"???/??/?)??/")b’??/??/??/??/’:f&&s(??/"???/??/?/??/")",
"s(??/"???/??/?/??/")b’??/??/n’:if(f)s(??/"???/??/?/n??/")",
"else case’??/"’:if(f)s(??/"???/??/?/??/??/??/"??/")",
"else default:c(*p)??>??>??>main()??<char**p",
";t(??/"char*a??(??)=??<??/",0);for(p=a;*p",
";p++)t(*p,1);t(??/"0??>;??/",0);for(p=a",
";*p;p++)t(*p,0);exit(!ferror(o)&&",
"!fclose(o)?EXIT_SUCCESS",
":EXIT_FAILURE);",
"/*NOTREACHED*/",
"??>",
0??>;
??=include<stdio.h>
??=include<stdlib.h>
??=define o stdout
??=define b break;case
??=define s(p)fputs(p,o);
??=define c(p)fputc(p,o);
void t(p,f)char*p;??<f&&c(’"’)
for(;;p++)??<switch(*p)??<case 0:f&&
s("??/",")c(’??/n’)return;case
’??=’:s("???/?=")b’??<’:s("???/?<")
b’??>’:s("???/?>")b’??(’:s("???/?(")b’??)’
:s("???/?)")b’??/??/’:f&&s("???/?/")
s("???/?/")b’??/n’:if(f)s("???/?/n")
```

```
else case’"’:if(f)s("???/?/??/"")
else default:c(*p)??>??>??>main()??<char**p
;t("char*a??(??)=??<",0);for(p=a;*p
;p++)t(*p,1);t("0??>;",0);for(p=a
;*p;p++)t(*p,0);exit(!ferror(o)&&
!fclose(o)?EXIT_SUCCESS
:EXIT_FAILURE);
/*NOTREACHED*/
??>
```

Best Utility

Byron Rakitzis
Sean Dorward
Princeton University

Judges’ comments:

This program implements a subject a well known Un*x utility whose original source was considered to be extremely obfuscated by many people, excluding its author. In fact, this utility one a major inspiration for the formation of this contest.

The author supplied us with a slightly smaller unformatted version of the program which we include below:

```
    #define D ,close(
    char*c,q[512],m[256],*v[99],**u,*i[3];int f[2],p;main(){for(m[m[60]=m[62]=
    32]=m[*m=124[m]=9]=6;e(-8),gets(1+(c=q))||exit(0);r(0,0))for(;*++c;);}
    r(t,o){*i=i[2]=0;for(u=v+98;m[*--c]^9;m[*c]&32?i[*c&2]=
    *u,u-v^98&&++u:3)if(!m[*c]){for(*++c=0;!m[*--c];);*--u=
    ++c;}u-v^98?strcmp(*u,"cd")?*c?pipe(f),o=f[1]:1,(p=fork())?e(p),o?
    r(o,0)D o)D*f):4,wait(0):(o?dup2(*f,0)D*f)D o):*i?1 D
    0),e(open(*i,0)):5,t?dup2(t,1)D t):i[2]?9 D
    1),e(creat(i[2],438)):2,e(execvp(*u,u))):e(chdir(u[1])*2):6;}
    e(x){x<0?write(2,"?\n$ "-x/4,2),x+1||exit(1):5;}
```

```
#define D ,close(

char              *c,q              [512              ],m[              256
],*v[           99], **u,        *i[3];int         f[2],p;main       (){for
 (m[m        [60]=   m[62      ]=32   ]=m[*      m=124   [m]=       9]=6;
  e(-8)     ,gets      (1+(    c=q)     )||      exit      (0);     r(0,0)
   )for(    ;*++        c;);  }r(t,      o){    *i=i        [2]=    0;for
```

```
      (u=v  +98              ;m[*--c]            ^9;m [*c]             &32  ?i[*c
        &2]=                   *u,u-                v^98                 &&++u:

3 )if(!m[*c]){for(*++c=0;!m[*--c];);
* --u= ++c;}u-v^98?strcmp(*u,"cd")?*c?pipe(f),o=f[
1 ]:
4 ,(p=fork())?e(p),o?r(o,0)D o)D*f):
1 ,wait(0):(o?dup2(*f,0)D*f)D o):*i?
5 D 0),e(open(*i,0)):
9 ,t?dup2(t,1)D t):i[
2 ]?
6 D 1),e(creat(i[2],438)):
5 ,e(execvp(*u,u))):e(chdir(u[1])*2):
3 ;}e(x){x<0?write(2,"?\n$ "-x/4,2),x+1||exit(1):
5 ;}
```

Best Layout

Brian Westley (Merlyn LeRoy on usenet)
DigiBoard, Inc.
1026 Blair Ave.
St. Paul, MN 55104

Judges' comments:

usage: westley ¡number¿

If you would rather "Daisy" someone other than Westley, rename the program as needed. :-)

Read each block of code as if it were a piece of correspondence. For example, the first block of code would read:

charlie, doubletime me, OXFACE! not interested, get out mainly die, charly, *die* signed charlotte

The original source had control-L's after each code block. To make it easier on news readers, we converted each control-L to a blank line.

Some ANSI compilers will not accept '1s' as a short integer - for these compilers replace the '1s' with '1'.

Selected notes from the author:

This is a "Picking the Daisy" simulation. Now, instead of mangling a daisy, simply run this program with the number of petals desired as the argument.

This is a good counter-example to peoples' complaints that C doesn't have an "English-like" syntax.

Lint complains about everything - null effect, xxx may be used before set, statement not reached, return(e) and return. Lint dumps core on some systems.

My personal favorite lint complaint is
”warning: eroticism unused in function main”.

Also obviously, (char)lotte and (char*)lie are incompatible types...

```
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=
"The gloves are OFF this time, I detest you, snot\n\0sed GEEK!");
do {not= *lie++ & 0xF00L* !me;

#define love (char*)lie -
love 1s *!(not= atoi(let
[get -me?
    (char)lotte-

(char)lotte: my- *love -
’I’  -  *love -  ’U’ -
’I’  -  (long)  - 4 - ’U’ ])- !!
(time  =out=  ’a’));} while( my - dear
&& ’I’-1l  -get-  ’a’); break;}}
    (char)*lie++;

(char)*lie++, (char)*lie++; hell:0, (char)*lie;
get *out* (short)ly   -0-’R’-  get- ’a’^rested;
do {auto*eroticism,
that; puts(*( out
    - ’c’
-(’P’-’S’) +die+ -2 ));}while(!"you’re at it");

for (*((char*)&lotte)^=
(char)lotte; (love ly) [(char)++lotte+
!!0xBABE];){ if (’I’ -lie[ 2 +(char)lotte]){ ’I’-1l ***die; }
else{ if (’I’ * get *out* (’I’-1l **die[ 2 ])) *((char*)&lotte) -=
’4’ - (’I’-1l); not; for(get=!
get; !out; (char)*lie  &  0xD0- !not) return!!
(char)lotte;}

(char)lotte;
do{ not* putchar(lie [out
*!not* !!me +(char)lotte]);
not; for(;!'a';);}while(
    love (char*)lie);{

register this; switch( (char)lie
[(char)lotte] -1s *!out) {
char*les, get= 0xFF, my; case' ':
*((char*)&lotte) += 15; !not +(char)*lie*'s';
this +1s+ not; default: 0xF +(char*)lie;}}}
get - !out;

if (not--)
goto hell;
    exit( (char)lotte);}
```