

## Fixed Points

The following programs compute themselves. They are fixed points of the function which takes text, compiles it to machine code and runs it.

### alderequia.c

Submitted by Alfredo Alderequia. All on one line.

```
char ticky=0x22;char *first="main(){printf(";char *second="char ticky=0x22;char *first=%c%c%c;char *second=%c%c%c;char *third=%c%c%c;%s%c%s%c%s";char *third=",ticky,first,ticky,ticky,second,ticky,ticky,third,ticky,first,ticky,second,ticky,third);}";main(){printf("char ticky=0x22;char *first=%c%c%c;char *second=%c%c%c;char *third=%c%c%c;%s%c%s%c%s",ticky,first,ticky,ticky,second,ticky,ticky,third,ticky,first,ticky,second,ticky,third);}
```

### boyd.c

Submitted by Al Boyd, in C++. Except for comments, it prints itself. Its output prints itself.

```
/* Alan Boyd   Math 688   Recursive program . prints itself   */
/* 0x5c = /    */
/* 0x27 = '    */
/* 0x22 = "    */
/* 0x3B = ;    */
/* 0x7D = }    */

#include<stdio.h>
void prtitt(char *s)
{
    int i,l = strlen(s);
    for(i=0;i<l;i++){
        if(s[i]=='!'){
            if(s[i-1]==0x27) putchar(s[i]);
            else putchar('\n');
        }
    }
}
```

```

        else if(s[i]==0x5c)
            putchar(0x5c);
        else if(s[i]=='\n'){
            putchar(0x5c); putchar('\n');
        }
        else putchar(s[i]);
    }
    putchar(0x22);
    for(i=0;i<l;i++){
        if(s[i]=='\n'){
            putchar(0x5c); putchar('\n');
        }
        else putchar(s[i]);
        if(s[i]=='!'){
            if(s[i-1]==0x27);
            else{
                putchar(0x5c);
                putchar('\n');
            }
        }
    }
    putchar(0x22); putchar(0x3B); putchar('\n');
    putchar(0x70); putchar(0x72); putchar(0x74);
    putchar(0x69); putchar(0x74); putchar(0x28);
    putchar(0x73); putchar(0x29); putchar(0x3B);
    putchar('\n'); putchar(0x7D);
}
main(){
char s[2000] = " !\
#include<stdio.h> !\
void prtit(char *s) !\
{ !\
    int i,l = strlen(s); !\
    for(i=0;i<l;i++){ !\
        if(s[i]=='!'){ !\
            if(s[i-1]==0x27) putchar(s[i]); !\
            else putchar('\n'); !\

```

```

    }
    else if(s[i]==0x5c)
        putchar(0x5c);
    else if(s[i]=='\n'){
        putchar(0x5c); putchar('\n');
    }
    else putchar(s[i]);
}
putchar(0x22);
for(i=0;i<l;i++){
    if(s[i]=='\n'){
        putchar(0x5c); putchar('\n');
    }
    else putchar(s[i]);
    if(s[i]=='!'){
        if(s[i-1]==0x27);
        else{
            putchar(0x5c);
            putchar('\n');
        }
    }
}
putchar(0x22); putchar(0x3B); putchar('\n');
putchar(0x70); putchar(0x72); putchar(0x74);
putchar(0x69); putchar(0x74); putchar(0x28);
putchar(0x73); putchar(0x29); putchar(0x3B);
putchar('\n'); putchar(0x7D);
}
main(){
char s[2000] = ";
prtit(s);
}

```

## **burt.pas**

Submitted by Burt Rosenberg. All this is on one line.

```
program pm(input,output);const a='program pm(input,output);const a=
;begin write(substr(a,1,33)+chr(39)+a+chr(39)+substr(a,34,66))end.'
;begin write(substr(a,1,33)+chr(39)+a+chr(39)+substr(a,34,66))end.
```

## **hargrave.c**

Submitted by Bentley Hargrave. The first newline is intentional. The second is not.

```
#include<stdio.h>
char *prog="#include<stdio.h>%cchar *prog=%c%s%c; void main() {prin
tf(prog,10,34,prog,34);}";void main() {printf(prog,10,34,prog,34);}
```

## **robinson.c**

Submitted by Robert Robinson. Only the newlines within the quoted string aa[] are unintentional.

```
#include<string.h>
char a1[]="/*This is great*/";
char aa[]="#include<string.h>%cchar a1[]={%c/*This is great*/%c;
%cchar aa[]={%c%s%c;%cmain()%c{%cprintf(aa,10,34,34,10,34,aa,34,
10,10,10,10,10,10)}%cprintf(a1);%c}%c";
main()
{
printf(aa,10,34,34,10,34,aa,34,10,10,10,10,10,10);
printf(a1);
}
/*This is great*/
```

## **Others**

Other correct solutions were submitted by Marilee Betor, William Piazza and C. Rutherford.