1. **Intro.** After a SAT solver has solved a problem set up with the SAT-TOMOGRAPHY programs, we want to see the answer in a convenient form. This program accepts the result (one line per solution) and converts the literals of the form $dxd$ into the rectangular “dots” format of periods and asterisks.

Input and output go from stdin to stdout.

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

char pix[101][101];

(Subroutine 2);

main()
{
    register int c, i, j, bit, maxi = 0, maxj = 0;
    while (1) {
        if (feof(stdin)) break;
            (Process the next line of input 3);
    }
}

2. (Subroutine 2) \(\equiv\)
   
   int nextchar(void)
   {
      register int c = fgetc(stdin);
      if (c \(!\!\! = EOF\)) return c;
      exit(-1);
   }

This code is used in section 1.

3. (Process the next line of input 3) \(\equiv\)
   
   for (c = nextchar(); c \(!\!\! = '\n'\); ) {
      (Process a literal 4);
   }
   (Output the pixels found 6);

This code is used in section 1.
4. \{ Process a literal 4 \} \equiv
c = \text{nextchar}();
if (c \neq '^-') bit = 1;
else {
  bit = 0;
  c = \text{nextchar}();
}
for (i = 0; c \geq '0' \land c \leq '9'; c = \text{nextchar}()) i = 10 \times i + c - '0';
if (i \geq 100) \{ 
  \text{fprintf} (\text{stderr}, \text{"Eh? I found a number of more than two digits! \n")};
  \text{exit} (-2);
}\}
if (c \neq 'x') \text{goto litdone};
c = \text{nextchar}();
for (j = 0; c \geq '0' \land c \leq '9'; c = \text{nextchar}()) j = 10 \times j + c - '0';
if (j \geq 100) \{ 
  \text{fprintf} (\text{stderr}, \text{"Eh? I found a number of more than two digits! \n")};
  \text{exit} (-2);
\}
if (c \neq '_.' \land c \neq '\n') \text{goto litdone};
(Record the pixel value \((i, j)\) 5);
litdone: while (c \neq '_.' \land c \neq '\n') c = \text{nextchar}();
This code is used in section 3.

5. \{ Record the pixel value \((i, j)\) 5 \} \equiv
if (i > maxi) maxi = i;
if (j > maxj) maxj = j;
pix[i][j] = bit;
This code is used in section 4.

6. \{ Output the pixels found 6 \} \equiv
for (i = 1; i \leq maxi; i++) {
  for (j = 1; j \leq maxj; j++) \text{putchar} (pix[i][j] ? '*': '.');
  \text{putchar} ('\n');
} 
\text{putchar} ('\n');
This code is used in section 3.
7. Index.

bit: 1, 4, 5.
c: 1, 2.
EOF: 2.
exit: 2, 4.
feof: 1.
fgetc: 2.
fprintf: 4.
i: 1.
j: 1.
litdone: 4.
main: 1.
maxi: 1, 5, 6.
maxj: 1, 5, 6.
nexchar: 2, 3, 4.
pix: 1, 5, 6.
putchar: 6.
stderr: 4.
stdin: 1, 2.
stdout: 1.
(Output the pixels found 6) Used in section 3.
(Process a literal 4) Used in section 3.
(Process the next line of input 3) Used in section 1.
(Record the pixel value \((i, j)\) 5) Used in section 4.
(Subroutine 2) Used in section 1.