$\S1$ QUEENS DATA FOR DANCING 1

(See https://cs.stanford.edu/~knuth/programs.html for date.)

1. Data for dancing. This program creates data suitable for the DANCE routine, solving the famous "n queens problem." The value of n is a command-line parameter.

```
#include <stdio.h>
#include <stdlib.h>
   (Global variables 3)
   \langle \text{Subroutines 5} \rangle;
   main(argc, argv)
         \mathbf{int}\ \mathit{argc};
         \mathbf{char} * argv[];
      register int j, k, n, nn, t;
      \langle \text{Read the command line 2} \rangle;
      \langle \text{ Output the column names 4} \rangle;
      \langle \text{ Output the possible queen moves } 6 \rangle;
2. \langle \text{Read the command line } 2 \rangle \equiv
  if (argc \neq 2 \lor sscanf(argv[1], "%d", \& param) \neq 1) {
      fprintf(stderr, "Usage: \_\%s_n\n", argv[0]);
      exit(-1);
  n = param;
  nn = n + n - 2;
This code is used in section 1.
3. \langle \text{Global variables 3} \rangle \equiv
  int param;
This code is used in section 1.
```

 $\langle \text{ Output the column names 4} \rangle \equiv$

This code is used in section 1.

4. We process the cells of the board in "organ pipe order," on the assumption that—all other things being equal—a move near the center yields more constraints on the subsequent search.

```
for (j = 0; j < n; j + +) {
	t = (j \& 1 ? n - 1 - j : n + j) \gg 1;
	printf("r\%c \Box c\%c \Box ", encode(t), encode(t));
}

printf("|");

for (j = 1; j < nn; j + +) printf(" \Box a\%c \Box b\%c", encode(j), encode(j));
printf(" \ ");

This code is used in section 1.

5. \langle \text{Subroutines 5} \rangle \equiv \text{char } encode(x)
	int x;
{
	if (x < 10) return '0' + x;
	return 'a' - 10 + x;
}
```

2 data for dancing queens $\S 6$

```
 \begin{aligned} \textbf{6.} & \text{$\langle$ Output the possible queen moves 6$\rangle$} \equiv \\ \textbf{for } & (j=0; \ j < n; \ j++) \\ & \textbf{for } & (k=0; \ k < n; \ k++) \ \{\\ & printf(\texttt{"r%c\_c%c"}, encode(j), encode(k)); \\ & t = j + k; \\ & \textbf{if } & (t \land (t < nn)) \ printf(\texttt{"\_a%c"}, encode(t)); \\ & t = n - 1 - j + k; \\ & \textbf{if } & (t \land (t < nn)) \ printf(\texttt{"\_b%c"}, encode(t)); \\ & printf(\texttt{"} \ "); \\ \} \end{aligned}
```

This code is used in section 1.

 $\S 7$ Queens index 3

7. Index.

```
\begin{array}{llll} argc: & \underline{1}, & 2. \\ argv: & \underline{1}, & 2. \\ encode: & 4, & \underline{5}, & 6. \\ exit: & 2. \\ fprintf: & 2. \\ j: & \underline{1}. \\ k: & \underline{1}. \\ main: & \underline{1}. \\ nr: & \underline{1}. \\ nn: & \underline{1}, & 2, & 4, & 6. \\ param: & 2, & \underline{3}. \\ printf: & 4, & 6. \\ sscanf: & 2. \\ stderr: & 2. \\ t: & \underline{1}. \\ x: & \underline{5}. \end{array}
```

4 NAMES OF THE SECTIONS QUEENS

```
 \begin{array}{lll} \left\langle \mbox{Global variables 3} \right\rangle & \mbox{Used in section 1.} \\ \left\langle \mbox{Output the column names 4} \right\rangle & \mbox{Used in section 1.} \\ \left\langle \mbox{Output the possible queen moves 6} \right\rangle & \mbox{Used in section 1.} \\ \left\langle \mbox{Read the command line 2} \right\rangle & \mbox{Used in section 1.} \\ \left\langle \mbox{Subroutines 5} \right\rangle & \mbox{Used in section 1.} \end{array}
```

QUEENS

	Section	Page
Data for dancing	1	1
Index	7	