

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

1. Intro. Pepperdine's method for topswops. [Reference: *Mathematical Gazette* **73** (1989), 131–133.]

```
#define n 15      /* degree of perms; should be less than 16 */
#define maxl 1000 /* max level */
#include <stdio.h>
typedef struct {
    char p[16];
} perm; /* this speeds it up by better than 2 */
perm a[maxl];
perm b[maxl];
char x[maxl];

main()
{
    register int j, k, l, m;
    a[0].p[0] = 1;
    m = l = 0;
tryit: k = ++x[l];
    if (k < n) {
        if (a[l].p[k] == 0) {
            if (b[l].p[k + 1] != 0) goto tryit;
        } else if (a[l].p[k] != k + 1) goto tryit;
        a[l + 1] = a[l];
        for (j = 1; j ≤ k; j++) a[l + 1].p[j] = a[l].p[k - j];
        b[l + 1] = b[l];
        a[l + 1].p[0] = k + 1, b[l + 1].p[k + 1] = 1;
        if (l ≥ m) {
            m = l;
            printf("%d:", m + 1);
            for (j = 0; j < n; j++) printf("␣%d", a[l + 1].p[j]);
            printf("\n");
        }
        l++;
        x[l] = 0;
        goto tryit;
    }
    l--;
    if (l ≥ 0) goto tryit;
}
```

2. Index.*a*: [1](#).*b*: [1](#).*j*: [1](#).*k*: [1](#).*l*: [1](#).*m*: [1](#).*main*: [1](#).*maxl*: [1](#).*n*: [1](#).*p*: [1](#).**perm**: [1](#).*printf*: [1](#).*tryit*: [1](#).*x*: [1](#).

TOPSWOPS

	Section	Page
Intro	1	1
Index	2	2