

Execvp "library" call

- Same process -- new program
- `int execvp (const char *file, char *const argv[])`
- `vp =>` search environment PATH variable for "file"
- file is also `argv[0]`

- Environment
 - `printenv`
 - PATH variable
 - Csh
 - `set path=(/bin /usr/bin /usr/pkg/bin)`
 - sh/bash
 - `PATH=/bin:/usr/bin:/usr/pkg/bin:....:`

Command line argument list

`emacs +367 -nw file.c`

□ `emacs` - program name (argument 0)

□ `+367` - argument 1

□ `-nw` - argument 2

□ `file.c` - argument 3

Program to print command line arguments

```
int main (int argc, char **argv)
{
    int ix;

    /* Args */
    for (ix=0; ix < argc; ix++)
        printf ("Arg %d: '%s'\n", ix, argv[ix]);

    return 0;
}
```

execvp again

- `int execvp (const char *file, char *const argv[])`
- file is the name of the executable file.
- argv is the argument list to new program.
- What you pass in as argv ... can be printed!

