

2o48.hackable.c
16D0

Generated by Doxygen 1.8.10

Wed Oct 28 2015 15:15:45

Contents

1	2o48.hackable.c	1
2	Data Structure Index	3
2.1	Data Structures	3
3	File Index	5
3.1	File List	5
4	Data Structure Documentation	7
4.1	_win Struct Reference	7
4.1.1	Detailed Description	8
4.1.2	Field Documentation	8
4.1.2.1	_attrs	8
4.1.2.2	_begx	8
4.1.2.3	_begy	8
4.1.2.4	_bkgd	8
4.1.2.5	_bmarg	8
4.1.2.6	_clear	8
4.1.2.7	_curx	8
4.1.2.8	_cury	8
4.1.2.9	_delayms	8
4.1.2.10	_firstch	9
4.1.2.11	_flags	9
4.1.2.12	_immed	9
4.1.2.13	_lastch	9
4.1.2.14	_leaveit	9
4.1.2.15	_maxx	9
4.1.2.16	_maxy	9
4.1.2.17	_nodelay	9
4.1.2.18	_parent	9
4.1.2.19	_parx	9
4.1.2.20	_pary	9

4.1.2.21	<code>_scroll</code>	9
4.1.2.22	<code>_sync</code>	10
4.1.2.23	<code>_tmarg</code>	10
4.1.2.24	<code>_use_keypad</code>	10
4.1.2.25	<code>_y</code>	10
4.2	MEVENT Struct Reference	10
4.2.1	Detailed Description	10
4.2.2	Field Documentation	10
4.2.2.1	<code>bstate</code>	10
4.2.2.2	<code>id</code>	10
4.2.2.3	<code>x</code>	10
4.2.2.4	<code>y</code>	11
4.2.2.5	<code>z</code>	11
4.3	MOUSE_STATUS Struct Reference	11
4.3.1	Detailed Description	11
4.3.2	Field Documentation	11
4.3.2.1	<code>button</code>	11
4.3.2.2	<code>changes</code>	11
4.3.2.3	<code>x</code>	11
4.3.2.4	<code>xbutton</code>	11
4.3.2.5	<code>y</code>	12
4.4	SCREEN Struct Reference	12
4.4.1	Detailed Description	13
4.4.2	Field Documentation	13
4.4.2.1	<code>_map_mbe_to_key</code>	13
4.4.2.2	<code>_preserve</code>	13
4.4.2.3	<code>_restore</code>	13
4.4.2.4	<code>_trap_mbe</code>	13
4.4.2.5	<code>alive</code>	13
4.4.2.6	<code>audible</code>	13
4.4.2.7	<code>autocr</code>	13
4.4.2.8	<code>cbreak</code>	13
4.4.2.9	<code>cols</code>	14
4.4.2.10	<code>curscol</code>	14
4.4.2.11	<code>cursrow</code>	14
4.4.2.12	<code>delaytenths</code>	14
4.4.2.13	<code>echo</code>	14
4.4.2.14	<code>key_code</code>	14
4.4.2.15	<code>line_color</code>	14
4.4.2.16	<code>lines</code>	14

4.4.2.17	linesrippedoff	14
4.4.2.18	linesrippedoffontop	14
4.4.2.19	mono	14
4.4.2.20	mouse_wait	14
4.4.2.21	orig_attr	15
4.4.2.22	orig_back	15
4.4.2.23	orig_cursor	15
4.4.2.24	orig_fore	15
4.4.2.25	raw_inp	15
4.4.2.26	raw_out	15
4.4.2.27	resized	15
4.4.2.28	return_key_modifiers	15
4.4.2.29	save_key_modifiers	15
4.4.2.30	slk_winpтр	15
4.4.2.31	slklines	15
4.4.2.32	visibility	15
5	File Documentation	17
5.1	2048.c File Reference	17
5.1.1	Macro Definition Documentation	19
5.1.1.1	EDOWN	19
5.1.1.2	ELEFT	19
5.1.1.3	ERIGHT	19
5.1.1.4	EUP	19
5.1.1.5	MAX_BOARD_NUM	20
5.1.1.6	MAX_BOARD_SIZE	20
5.1.1.7	MENU_POSITION_X	20
5.1.1.8	MENU_POSITION_Y	20
5.1.1.9	PWD	20
5.1.1.10	PWD_LEN	20
5.1.1.11	WARNING_POSITION_X	20
5.1.1.12	WARNING_POSITION_Y	20
5.1.2	Function Documentation	20
5.1.2.1	AlignCol(int curcol, int direction)	20
5.1.2.2	AlignLine(int curline, int direction)	21
5.1.2.3	c_checksum()	21
5.1.2.4	c_currentStr(bool show)	22
5.1.2.5	c_forceQuit()	23
5.1.2.6	c_loadStr()	23
5.1.2.7	c_loadStr(int iptN, FILE *fp)	23

5.1.2.8	c_readBoard(int from)	24
5.1.2.9	c_readFromDisk(int boards)	25
5.1.2.10	c_saveBoard(int to, bool jmp)	26
5.1.2.11	c_tryQuit()	27
5.1.2.12	c_warning(char *warn)	28
5.1.2.13	c_writeBoardToDisk(char boards)	29
5.1.2.14	CheckEat(char *a, char *b)	30
5.1.2.15	Clrboard(int boardToClr)	31
5.1.2.16	command()	31
5.1.2.17	die()	33
5.1.2.18	Display(char in)	35
5.1.2.19	Eat(bool isH, int direction)	35
5.1.2.20	EatCol(int curcol, int direction)	36
5.1.2.21	EatLine(const int curline, int direction)	37
5.1.2.22	GetRandNums()	38
5.1.2.23	main()	39
5.1.2.24	play()	40
5.1.2.25	Rando(int N)	42
5.1.2.26	settings()	42
5.1.2.27	showBoard(int offy, int offx)	43
5.1.2.28	welcome()	44
5.1.3	Variable Documentation	45
5.1.3.1	board	45
5.1.3.2	boardseed	45
5.1.3.3	boardstr	45
5.1.3.4	col	46
5.1.3.5	cs_pwd	46
5.1.3.6	curs	46
5.1.3.7	display	46
5.1.3.8	eat	46
5.1.3.9	MAX_RANDOM	46
5.1.3.10	N	46
5.1.3.11	NA	46
5.1.3.12	P_RANDOM	46
5.1.3.13	point	47
5.1.3.14	row	47
5.2	ncurses.h File Reference	47
5.2.1	Macro Definition Documentation	64
5.2.1.1	A_ALTCHARSET	64
5.2.1.2	A_ATTRIBUTES	64

5.2.1.3	A_BLINK	64
5.2.1.4	A_BOLD	64
5.2.1.5	A_BUTTON_CHANGED	64
5.2.1.6	A_CHARTEXT	64
5.2.1.7	A_COLOR	64
5.2.1.8	A_DIM	65
5.2.1.9	A_INVIS	65
5.2.1.10	A_ITALIC	65
5.2.1.11	A_LEFTLINE	65
5.2.1.12	A_NORMAL	65
5.2.1.13	A_OVERLINE	65
5.2.1.14	A_PROTECT	65
5.2.1.15	A_REVERSE	65
5.2.1.16	A_RGB	65
5.2.1.17	A_RGB_COLOR	65
5.2.1.18	A_RIGHTLINE	65
5.2.1.19	A_STANDOUT	66
5.2.1.20	A_STRIKEOUT	66
5.2.1.21	A_UNDERLINE	66
5.2.1.22	ACS_BBSS	66
5.2.1.23	ACS_BLOCK	66
5.2.1.24	ACS_BOARD	66
5.2.1.25	ACS_BSBS	66
5.2.1.26	ACS_BSSB	66
5.2.1.27	ACS_BSSS	66
5.2.1.28	ACS_BTEE	66
5.2.1.29	ACS_BULLET	66
5.2.1.30	ACS_CKBOARD	66
5.2.1.31	ACS_DARROW	67
5.2.1.32	ACS_DEGREE	67
5.2.1.33	ACS_DIAMOND	67
5.2.1.34	ACS_GEQUAL	67
5.2.1.35	ACS_HLINE	67
5.2.1.36	ACS_LANTERN	67
5.2.1.37	ACS_LARROW	67
5.2.1.38	ACS_LEQUAL	67
5.2.1.39	ACS_LLCORNER	67
5.2.1.40	ACS_LRCORNER	67
5.2.1.41	ACS_LTEE	67
5.2.1.42	ACS_NEQUAL	67

5.2.1.43	ACS_PI	68
5.2.1.44	ACS_PICK	68
5.2.1.45	ACS_PLMINUS	68
5.2.1.46	ACS_PLUS	68
5.2.1.47	ACS_RARROW	68
5.2.1.48	ACS_RTEE	68
5.2.1.49	ACS_S1	68
5.2.1.50	ACS_S3	68
5.2.1.51	ACS_S7	68
5.2.1.52	ACS_S9	68
5.2.1.53	ACS_SBBS	68
5.2.1.54	ACS_SBSB	68
5.2.1.55	ACS_SBSS	69
5.2.1.56	ACS_SSBB	69
5.2.1.57	ACS_SSBS	69
5.2.1.58	ACS_SSSB	69
5.2.1.59	ACS_SSSS	69
5.2.1.60	ACS_STERLING	69
5.2.1.61	ACS_TTEE	69
5.2.1.62	ACS_UARROW	69
5.2.1.63	ACS_ULCORNER	69
5.2.1.64	ACS_URCORNER	69
5.2.1.65	ACS_VLINE	69
5.2.1.66	ALL_MOUSE_EVENTS	69
5.2.1.67	ALT_0	70
5.2.1.68	ALT_1	70
5.2.1.69	ALT_2	70
5.2.1.70	ALT_3	70
5.2.1.71	ALT_4	70
5.2.1.72	ALT_5	70
5.2.1.73	ALT_6	70
5.2.1.74	ALT_7	70
5.2.1.75	ALT_8	70
5.2.1.76	ALT_9	70
5.2.1.77	ALT_A	70
5.2.1.78	ALT_APPS	70
5.2.1.79	ALT_B	71
5.2.1.80	ALT_BKSP	71
5.2.1.81	ALT_BQUOTE	71
5.2.1.82	ALT_BSLASH	71

5.2.1.83 ALT_C	71
5.2.1.84 ALT_COMMA	71
5.2.1.85 ALT_D	71
5.2.1.86 ALT_DEL	71
5.2.1.87 ALT_DOWN	71
5.2.1.88 ALT_E	71
5.2.1.89 ALT_END	71
5.2.1.90 ALT_ENTER	71
5.2.1.91 ALT_EQUAL	72
5.2.1.92 ALT_ESC	72
5.2.1.93 ALT_F	72
5.2.1.94 ALT_FQUOTE	72
5.2.1.95 ALT_FSLASH	72
5.2.1.96 ALT_G	72
5.2.1.97 ALT_H	72
5.2.1.98 ALT_HOME	72
5.2.1.99 ALT_I	72
5.2.1.100 ALT_INS	72
5.2.1.101 ALT_J	72
5.2.1.102 ALT_K	72
5.2.1.103 ALT_L	73
5.2.1.104 ALT_LBRACKET	73
5.2.1.105 ALT_LEFT	73
5.2.1.106 ALT_M	73
5.2.1.107 ALT_MINUS	73
5.2.1.108 ALT_N	73
5.2.1.109 ALT_O	73
5.2.1.110 ALT_P	73
5.2.1.111 ALT_PAD0	73
5.2.1.112 ALT_PAD1	73
5.2.1.113 ALT_PAD2	73
5.2.1.114 ALT_PAD3	73
5.2.1.115 ALT_PAD4	74
5.2.1.116 ALT_PAD5	74
5.2.1.117 ALT_PAD6	74
5.2.1.118 ALT_PAD7	74
5.2.1.119 ALT_PAD8	74
5.2.1.120 ALT_PAD9	74
5.2.1.121 ALT_PADENTER	74
5.2.1.122 ALT_PADMINUS	74

5.2.1.123 ALT_PADPLUS	74
5.2.1.124 ALT_PADSLASH	74
5.2.1.125 ALT_PADSTAR	74
5.2.1.126 ALT_PADSTOP	74
5.2.1.127 ALT_PGDN	75
5.2.1.128 ALT_PGUP	75
5.2.1.129 ALT_PRINTSCREEN	75
5.2.1.130 ALT_Q	75
5.2.1.131 ALT_R	75
5.2.1.132 ALT_RBRACKET	75
5.2.1.133 ALT_RIGHT	75
5.2.1.134 ALT_S	75
5.2.1.135 ALT_SCROLLLOCK	75
5.2.1.136 ALT_SEMICOLON	75
5.2.1.137 ALT_STOP	75
5.2.1.138 ALT_T	75
5.2.1.139 ALT_TAB	76
5.2.1.140 ALT_U	76
5.2.1.141 ALT_UP	76
5.2.1.142 ALT_V	76
5.2.1.143 ALT_W	76
5.2.1.144 ALT_X	76
5.2.1.145 ALT_Y	76
5.2.1.146 ALT_Z	76
5.2.1.147 ATR_MSK	76
5.2.1.148 ATR_NRM	76
5.2.1.149 BSDcurses	76
5.2.1.150 BUTTON1_CLICKED	76
5.2.1.151 BUTTON1_DOUBLE_CLICKED	77
5.2.1.152 BUTTON1_MOVED	77
5.2.1.153 BUTTON1_PRESSED	77
5.2.1.154 BUTTON1_RELEASED	77
5.2.1.155 BUTTON1_TRIPLE_CLICKED	77
5.2.1.156 BUTTON2_CLICKED	77
5.2.1.157 BUTTON2_DOUBLE_CLICKED	77
5.2.1.158 BUTTON2_MOVED	77
5.2.1.159 BUTTON2_PRESSED	77
5.2.1.160 BUTTON2_RELEASED	77
5.2.1.161 BUTTON2_TRIPLE_CLICKED	77
5.2.1.162 BUTTON3_CLICKED	77

5.2.1.163 BUTTON3_DOUBLE_CLICKED	78
5.2.1.164 BUTTON3_MOVED	78
5.2.1.165 BUTTON3_PRESSED	78
5.2.1.166 BUTTON3_RELEASED	78
5.2.1.167 BUTTON3_TRIPLE_CLICKED	78
5.2.1.168 BUTTON4_CLICKED	78
5.2.1.169 BUTTON4_DOUBLE_CLICKED	78
5.2.1.170 BUTTON4_PRESSED	78
5.2.1.171 BUTTON4_RELEASED	78
5.2.1.172 BUTTON4_TRIPLE_CLICKED	78
5.2.1.173 BUTTON5_CLICKED	78
5.2.1.174 BUTTON5_DOUBLE_CLICKED	78
5.2.1.175 BUTTON5_PRESSED	79
5.2.1.176 BUTTON5_RELEASED	79
5.2.1.177 BUTTON5_TRIPLE_CLICKED	79
5.2.1.178 BUTTON_ACTION_MASK	79
5.2.1.179 BUTTON_ALT	79
5.2.1.180 BUTTON_CHANGED	79
5.2.1.181 BUTTON_CLICKED	79
5.2.1.182 BUTTON_CONTROL	79
5.2.1.183 BUTTON_DOUBLE_CLICKED	79
5.2.1.184 BUTTON_MODIFIER_ALT	79
5.2.1.185 BUTTON_MODIFIER_CONTROL	79
5.2.1.186 BUTTON_MODIFIER_MASK	79
5.2.1.187 BUTTON_MODIFIER_SHIFT	80
5.2.1.188 BUTTON_MOVED	80
5.2.1.189 BUTTON_PRESSED	80
5.2.1.190 BUTTON_RELEASED	80
5.2.1.191 BUTTON_SHIFT	80
5.2.1.192 BUTTON_STATUS	80
5.2.1.193 BUTTON_TRIPLE_CLICKED	80
5.2.1.194 CHR_MSK	80
5.2.1.195 CHTYPE_LONG	80
5.2.1.196 COLOR_BLACK	80
5.2.1.197 COLOR_BLUE	80
5.2.1.198 COLOR_CYAN	80
5.2.1.199 COLOR_GREEN	81
5.2.1.200 COLOR_MAGENTA	81
5.2.1.201 COLOR_PAIR	81
5.2.1.202 COLOR_RED	81

5.2.1.203 COLOR_WHITE	81
5.2.1.204 COLOR_YELLOW	81
5.2.1.205 CTL_0	81
5.2.1.206 CTL_1	81
5.2.1.207 CTL_2	81
5.2.1.208 CTL_3	81
5.2.1.209 CTL_4	81
5.2.1.210 CTL_5	81
5.2.1.211 CTL_6	82
5.2.1.212 CTL_7	82
5.2.1.213 CTL_8	82
5.2.1.214 CTL_9	82
5.2.1.215 CTL_APPS	82
5.2.1.216 CTL_BKSP	82
5.2.1.217 CTL_BQUOTE	82
5.2.1.218 CTL_COMMA	82
5.2.1.219 CTL_DEL	82
5.2.1.220 CTL_DOWN	82
5.2.1.221 CTL_END	82
5.2.1.222 CTL_ENTER	82
5.2.1.223 CTL_EQUAL	83
5.2.1.224 CTL_FSLASH	83
5.2.1.225 CTL_HOME	83
5.2.1.226 CTL_INS	83
5.2.1.227 CTL_LEFT	83
5.2.1.228 CTL_MINUS	83
5.2.1.229 CTL_PAD0	83
5.2.1.230 CTL_PAD1	83
5.2.1.231 CTL_PAD2	83
5.2.1.232 CTL_PAD3	83
5.2.1.233 CTL_PAD4	83
5.2.1.234 CTL_PAD5	83
5.2.1.235 CTL_PAD6	84
5.2.1.236 CTL_PAD7	84
5.2.1.237 CTL_PAD8	84
5.2.1.238 CTL_PAD9	84
5.2.1.239 CTL_PADCENTER	84
5.2.1.240 CTL_PADENTER	84
5.2.1.241 CTL_PADMINUS	84
5.2.1.242 CTL_PADPLUS	84

5.2.1.243 CTL_PADSLASH	84
5.2.1.244 CTL_PADSTAR	84
5.2.1.245 CTL_PADSTOP	84
5.2.1.246 CTL_PAUSE	84
5.2.1.247 CTL_PGDN	85
5.2.1.248 CTL_PGUP	85
5.2.1.249 CTL_RIGHT	85
5.2.1.250 CTL_SEMICOLON	85
5.2.1.251 CTL_STOP	85
5.2.1.252 CTL_TAB	85
5.2.1.253 CTL_UP	85
5.2.1.254 ERR	85
5.2.1.255 FALSE	85
5.2.1.256 getbegyx	85
5.2.1.257 getch	85
5.2.1.258 getmaxyx	85
5.2.1.259 getparyx	86
5.2.1.260 getsyx	86
5.2.1.261 getyx	86
5.2.1.262 KEY_A1	86
5.2.1.263 KEY_A2	86
5.2.1.264 KEY_A3	86
5.2.1.265 KEY_ABORT	86
5.2.1.266 KEY_ABROWSER_BACK	86
5.2.1.267 KEY_ABROWSER_FWD	86
5.2.1.268 KEY_ABROWSER_HOME	86
5.2.1.269 KEY_ABROWSER_REF	86
5.2.1.270 KEY_ABROWSER_STOP	87
5.2.1.271 KEY_AFAVORITES	87
5.2.1.272 KEY_ALAUNCH_APP1	87
5.2.1.273 KEY_ALAUNCH_APP2	87
5.2.1.274 KEY_ALAUNCH_MAIL	87
5.2.1.275 KEY_ALT_L	87
5.2.1.276 KEY_ALT_R	87
5.2.1.277 KEY_AMEDIA_SELECT	87
5.2.1.278 KEY_AMEDIA_STOP	87
5.2.1.279 KEY_ANEXT_TRACK	87
5.2.1.280 KEY_APLAY_PAUSE	87
5.2.1.281 KEY_APPS	87
5.2.1.282 KEY_APREV_TRACK	88

5.2.1.283 KEY_ASEARCH	88
5.2.1.284 KEY_AVOLUME_DOWN	88
5.2.1.285 KEY_AVOLUME_MUTE	88
5.2.1.286 KEY_AVOLUME_UP	88
5.2.1.287 KEY_B1	88
5.2.1.288 KEY_B2	88
5.2.1.289 KEY_B3	88
5.2.1.290 KEY_BACKSPACE	88
5.2.1.291 KEY_BEG	88
5.2.1.292 KEY_BREAK	88
5.2.1.293 KEY_BROWSER_BACK	88
5.2.1.294 KEY_BROWSER_FWD	89
5.2.1.295 KEY_BROWSER_HOME	89
5.2.1.296 KEY_BROWSER_REF	89
5.2.1.297 KEY_BROWSER_STOP	89
5.2.1.298 KEY_BTAB	89
5.2.1.299 KEY_C1	89
5.2.1.300 KEY_C2	89
5.2.1.301 KEY_C3	89
5.2.1.302 KEY_CANCEL	89
5.2.1.303 KEY_CATAB	89
5.2.1.304 KEY_CBROWSER_BACK	89
5.2.1.305 KEY_CBROWSER_FWD	89
5.2.1.306 KEY_CBROWSER_HOME	90
5.2.1.307 KEY_CBROWSER_REF	90
5.2.1.308 KEY_CBROWSER_STOP	90
5.2.1.309 KEY_CFAVORITES	90
5.2.1.310 KEY_CLAUNCH_APP1	90
5.2.1.311 KEY_CLAUNCH_APP2	90
5.2.1.312 KEY_CLAUNCH_MAIL	90
5.2.1.313 KEY_CLEAR	90
5.2.1.314 KEY_CLOSE	90
5.2.1.315 KEY_CMEDIA_SELECT	90
5.2.1.316 KEY_CMEDIA_STOP	90
5.2.1.317 KEY_CNEXT_TRACK	90
5.2.1.318 KEY_CODE_YES	91
5.2.1.319 KEY_COMMAND	91
5.2.1.320 KEY_CONTROL_L	91
5.2.1.321 KEY_CONTROL_R	91
5.2.1.322 KEY_COPY	91

5.2.1.323 KEY_CPLAY_PAUSE	91
5.2.1.324 KEY_CPREV_TRACK	91
5.2.1.325 KEY_CREATE	91
5.2.1.326 KEY_CSEARCH	91
5.2.1.327 KEY_CTAB	91
5.2.1.328 KEY_CVOLUME_DOWN	91
5.2.1.329 KEY_CVOLUME_MUTE	91
5.2.1.330 KEY_CVOLUME_UP	92
5.2.1.331 KEY_DC	92
5.2.1.332 KEY_DL	92
5.2.1.333 KEY_DOWN	92
5.2.1.334 KEY_EIC	92
5.2.1.335 KEY_END	92
5.2.1.336 KEY_ENTER	92
5.2.1.337 KEY_EOL	92
5.2.1.338 KEY_EOS	92
5.2.1.339 KEY_EXIT	92
5.2.1.340 KEY_F	92
5.2.1.341 KEY_F0	92
5.2.1.342 KEY_FAVORITES	93
5.2.1.343 KEY_FIND	93
5.2.1.344 KEY_HELP	93
5.2.1.345 KEY_HOME	93
5.2.1.346 KEY_IC	93
5.2.1.347 KEY_IL	93
5.2.1.348 KEY_LAUNCH_APP1	93
5.2.1.349 KEY_LAUNCH_APP2	93
5.2.1.350 KEY_LAUNCH_MAIL	93
5.2.1.351 KEY_LEFT	93
5.2.1.352 KEY_LHELP	93
5.2.1.353 KEY_LL	93
5.2.1.354 KEY_MARK	94
5.2.1.355 KEY_MAX	94
5.2.1.356 KEY_MEDIA_SELECT	94
5.2.1.357 KEY_MEDIA_STOP	94
5.2.1.358 KEY_MESSAGE	94
5.2.1.359 KEY_MIN	94
5.2.1.360 KEY_MOUSE	94
5.2.1.361 KEY_MOVE	94
5.2.1.362 KEY_NEXT	94

5.2.1.363 KEY_NEXT_TRACK	94
5.2.1.364 KEY_NPAGE	94
5.2.1.365 KEY_OFFSET	94
5.2.1.366 KEY_OPEN	95
5.2.1.367 KEY_OPTIONS	95
5.2.1.368 KEY_PAUSE	95
5.2.1.369 KEY_PLAY_PAUSE	95
5.2.1.370 KEY_PPAGE	95
5.2.1.371 KEY_PREV_TRACK	95
5.2.1.372 KEY_PREVIOUS	95
5.2.1.373 KEY_PRINT	95
5.2.1.374 KEY_PRINTSCREEN	95
5.2.1.375 KEY_REDO	95
5.2.1.376 KEY_REFERENCE	95
5.2.1.377 KEY_REFRESH	95
5.2.1.378 KEY_REPLACE	96
5.2.1.379 KEY_RESET	96
5.2.1.380 KEY_RESIZE	96
5.2.1.381 KEY_RESTART	96
5.2.1.382 KEY_RESUME	96
5.2.1.383 KEY_RIGHT	96
5.2.1.384 KEY_SAPPS	96
5.2.1.385 KEY_SAVE	96
5.2.1.386 KEY_SBEG	96
5.2.1.387 KEY_SBROWSER_BACK	96
5.2.1.388 KEY_SBROWSER_FWD	96
5.2.1.389 KEY_SBROWSER_HOME	96
5.2.1.390 KEY_SBROWSER_REF	97
5.2.1.391 KEY_SBROWSER_STOP	97
5.2.1.392 KEY_SCANCEL	97
5.2.1.393 KEY_SCOMMAND	97
5.2.1.394 KEY_SCOPY	97
5.2.1.395 KEY_SCREATE	97
5.2.1.396 KEY_SCROLLLOCK	97
5.2.1.397 KEY_SDC	97
5.2.1.398 KEY_SDL	97
5.2.1.399 KEY_SDOWN	97
5.2.1.400 KEY_SEARCH	97
5.2.1.401 KEY_SELECT	97
5.2.1.402 KEY_SEND	98

5.2.1.403 KEY_SEOL	98
5.2.1.404 KEY_SEXIT	98
5.2.1.405 KEY_SF	98
5.2.1.406 KEY_SFAVORITES	98
5.2.1.407 KEY_SFIND	98
5.2.1.408 KEY_SHELP	98
5.2.1.409 KEY_SHIFT_L	98
5.2.1.410 KEY_SHIFT_R	98
5.2.1.411 KEY_SHOME	98
5.2.1.412 KEY_SIC	98
5.2.1.413 KEY_SLAUNCH_APP1	98
5.2.1.414 KEY_SLAUNCH_APP2	99
5.2.1.415 KEY_SLAUNCH_MAIL	99
5.2.1.416 KEY_SLEFT	99
5.2.1.417 KEY_SMEDIA_SELECT	99
5.2.1.418 KEY_SMEDIA_STOP	99
5.2.1.419 KEY_SMESSAGE	99
5.2.1.420 KEY_SMOVE	99
5.2.1.421 KEY_SNEXT	99
5.2.1.422 KEY_SNEXT_TRACK	99
5.2.1.423 KEY_SOPTIONS	99
5.2.1.424 KEY_SPAUSE	99
5.2.1.425 KEY_SPLAY_PAUSE	99
5.2.1.426 KEY_SPREV_TRACK	100
5.2.1.427 KEY_SPREVIOUS	100
5.2.1.428 KEY_SPRINT	100
5.2.1.429 KEY_SR	100
5.2.1.430 KEY_SREDO	100
5.2.1.431 KEY_SREPLACE	100
5.2.1.432 KEY_SRESET	100
5.2.1.433 KEY_SRIGHT	100
5.2.1.434 KEY_SRSUME	100
5.2.1.435 KEY_SSAVE	100
5.2.1.436 KEY_SSEARCH	100
5.2.1.437 KEY_SSUSPEND	100
5.2.1.438 KEY_STAB	101
5.2.1.439 KEY_SUNDO	101
5.2.1.440 KEY_SUP	101
5.2.1.441 KEY_SUSPEND	101
5.2.1.442 KEY_SVOLUME_DOWN	101

5.2.1.443 KEY_SVOLUME_MUTE	101
5.2.1.444 KEY_SVOLUME_UP	101
5.2.1.445 KEY_UNDO	101
5.2.1.446 KEY_UP	101
5.2.1.447 KEY_VOLUME_DOWN	101
5.2.1.448 KEY_VOLUME_MUTE	101
5.2.1.449 KEY_VOLUME_UP	101
5.2.1.450 MOUSE_MOVED	102
5.2.1.451 MOUSE_POS_REPORT	102
5.2.1.452 MOUSE_WHEEL_DOWN	102
5.2.1.453 MOUSE_WHEEL_LEFT	102
5.2.1.454 MOUSE_WHEEL_RIGHT	102
5.2.1.455 MOUSE_WHEEL_SCROLL	102
5.2.1.456 MOUSE_WHEEL_UP	102
5.2.1.457 MOUSE_X_POS	102
5.2.1.458 MOUSE_Y_POS	102
5.2.1.459 NULL	102
5.2.1.460 OK	102
5.2.1.461 PAD0	102
5.2.1.462 PADENTER	103
5.2.1.463 PADMINUS	103
5.2.1.464 PADPLUS	103
5.2.1.465 PADSLASH	103
5.2.1.466 PADSTAR	103
5.2.1.467 PADSTOP	103
5.2.1.468 PAIR_NUMBER	103
5.2.1.469 PDC_BUILD	103
5.2.1.470 PDC_BUTTON_ALT	103
5.2.1.471 PDC_BUTTON_CONTROL	103
5.2.1.472 PDC_BUTTON_SHIFT	103
5.2.1.473 PDC_CHARTEXT_BITS	103
5.2.1.474 PDC_CLIP_ACCESS_ERROR	104
5.2.1.475 PDC_CLIP_EMPTY	104
5.2.1.476 PDC_CLIP_MEMORY_ERROR	104
5.2.1.477 PDC_CLIP_SUCCESS	104
5.2.1.478 PDC_COLOR_SHIFT	104
5.2.1.479 PDC_KEY_MODIFIER_ALT	104
5.2.1.480 PDC_KEY_MODIFIER_CONTROL	104
5.2.1.481 PDC_KEY_MODIFIER_NUMLOCK	104
5.2.1.482 PDC_KEY_MODIFIER_SHIFT	104

5.2.1.483 PDC_MAX_MOUSE_BUTTONS	104
5.2.1.484 PDC_MOUSE_MOVED	104
5.2.1.485 PDC_MOUSE_POSITION	104
5.2.1.486 PDC_MOUSE_WHEEL_DOWN	105
5.2.1.487 PDC_MOUSE_WHEEL_LEFT	105
5.2.1.488 PDC_MOUSE_WHEEL_RIGHT	105
5.2.1.489 PDC_MOUSE_WHEEL_UP	105
5.2.1.490 PDC_N_EXTENDED_MOUSE_BUTTONS	105
5.2.1.491 PDCEX	105
5.2.1.492 PDCURSES	105
5.2.1.493 REPORT_MOUSE_POSITION	105
5.2.1.494 SHF_DC	105
5.2.1.495 SHF_DOWN	105
5.2.1.496 SHF_IC	105
5.2.1.497 SHF_PADENTER	105
5.2.1.498 SHF_PADMINUS	106
5.2.1.499 SHF_PADPLUS	106
5.2.1.500 SHF_PADSLASH	106
5.2.1.501 SHF_PADSTAR	106
5.2.1.502 SHF_UP	106
5.2.1.503 SYSVcurses	106
5.2.1.504 TRUE	106
5.2.1.505 ungetch	106
5.2.1.506 WA_ALTCHARSET	106
5.2.1.507 WA_ATTRIBUTES	106
5.2.1.508 WA_BLINK	106
5.2.1.509 WA_BOLD	106
5.2.1.510 WA_DIM	107
5.2.1.511 WA_HORIZONTAL	107
5.2.1.512 WA_INVIS	107
5.2.1.513 WA_LEFT	107
5.2.1.514 WA_LOW	107
5.2.1.515 WA_NORMAL	107
5.2.1.516 WA_PROTECT	107
5.2.1.517 WA_REVERSE	107
5.2.1.518 WA_RIGHT	107
5.2.1.519 WA_STANDOUT	107
5.2.1.520 WA_TOP	107
5.2.1.521 WA_UNDERLINE	107
5.2.1.522 WA_VERTICAL	108

5.2.1.523	WHEEL_SCROLLED	108
5.2.1.524	XOPEN	108
5.2.2	Typedef Documentation	108
5.2.2.1	attr_t	108
5.2.2.2	bool	108
5.2.2.3	cchar_t	108
5.2.2.4	chtype	108
5.2.2.5	mmask_t	108
5.2.2.6	WINDOW	108
5.2.3	Function Documentation	108
5.2.3.1	addch(const chtype)	108
5.2.3.2	addchnstr(const chtype *, int)	108
5.2.3.3	addchstr(const chtype *)	108
5.2.3.4	addnstr(const char *, int)	108
5.2.3.5	addrawch(chtype)	108
5.2.3.6	addstr(const char *)	108
5.2.3.7	assume_default_colors(int, int)	109
5.2.3.8	attr_get(attr_t *, short *, void *)	109
5.2.3.9	attr_off(attr_t, void *)	109
5.2.3.10	attr_on(attr_t, void *)	109
5.2.3.11	attr_set(attr_t, short, void *)	109
5.2.3.12	attroff(chtype)	109
5.2.3.13	attron(chtype)	109
5.2.3.14	attrset(chtype)	110
5.2.3.15	baudrate(void)	110
5.2.3.16	beep(void)	110
5.2.3.17	bkgd(chtype)	110
5.2.3.18	bkgdset(chtype)	110
5.2.3.19	border(chtype, chtype, chtype, chtype, chtype, chtype, chtype, chtype)	110
5.2.3.20	box(WINDOW *, chtype, chtype)	110
5.2.3.21	can_change_color(void)	110
5.2.3.22	cbreak(void)	110
5.2.3.23	chgat(int, attr_t, short, const void *)	110
5.2.3.24	clear(void)	110
5.2.3.25	clearok(WINDOW *, bool)	110
5.2.3.26	clrtoebot(void)	110
5.2.3.27	clrtoeol(void)	111
5.2.3.28	color_content(short, short *, short *, short *)	111
5.2.3.29	color_set(short, void *)	111
5.2.3.30	copywin(const WINDOW *, WINDOW *, int, int, int, int, int, int, int)	111

5.2.3.31	crmode(void)	111
5.2.3.32	curs_set(int)	111
5.2.3.33	curses_version(void)	111
5.2.3.34	def_prog_mode(void)	111
5.2.3.35	def_shell_mode(void)	111
5.2.3.36	delay_output(int)	111
5.2.3.37	delch(void)	112
5.2.3.38	deleteln(void)	112
5.2.3.39	delscreen(SCREEN *)	112
5.2.3.40	delwin(WINDOW *)	112
5.2.3.41	derwin(WINDOW *, int, int, int, int)	112
5.2.3.42	doupdate(void)	112
5.2.3.43	draino(int)	112
5.2.3.44	dupwin(WINDOW *)	112
5.2.3.45	echo(void)	112
5.2.3.46	echochar(const chtype)	112
5.2.3.47	endwin(void)	112
5.2.3.48	erase(void)	112
5.2.3.49	erasechar(void)	112
5.2.3.50	filter(void)	112
5.2.3.51	fixterm(void)	113
5.2.3.52	flash(void)	113
5.2.3.53	flushinp(void)	113
5.2.3.54	getattrs(WINDOW *)	113
5.2.3.55	getbegx(WINDOW *)	113
5.2.3.56	getbegy(WINDOW *)	113
5.2.3.57	getbkgd(WINDOW *)	113
5.2.3.58	getbmap(void)	113
5.2.3.59	getcurx(WINDOW *)	113
5.2.3.60	getcury(WINDOW *)	113
5.2.3.61	getmaxx(WINDOW *)	113
5.2.3.62	getmaxy(WINDOW *)	113
5.2.3.63	getmouse(void)	113
5.2.3.64	getnstr(char *, int)	113
5.2.3.65	getparx(WINDOW *)	113
5.2.3.66	getpary(WINDOW *)	113
5.2.3.67	getstr(char *)	113
5.2.3.68	getwin(FILE *)	113
5.2.3.69	halfdelay(int)	113
5.2.3.70	has_colors(void)	113

5.2.3.71	has_ic(void)	113
5.2.3.72	has_il(void)	114
5.2.3.73	has_key(int)	114
5.2.3.74	hline(chtype, int)	114
5.2.3.75	idcok(WINDOW *, bool)	114
5.2.3.76	idlok(WINDOW *, bool)	114
5.2.3.77	immedok(WINDOW *, bool)	114
5.2.3.78	inch(void)	114
5.2.3.79	inchnstr(chtype *, int)	114
5.2.3.80	inchstr(chtype *)	114
5.2.3.81	init_color(short, short, short, short)	114
5.2.3.82	init_pair(short, short, short)	114
5.2.3.83	initscr(void)	114
5.2.3.84	innstr(char *, int)	114
5.2.3.85	insch(chtype)	114
5.2.3.86	insdelln(int)	114
5.2.3.87	insertln(void)	115
5.2.3.88	insnstr(const char *, int)	115
5.2.3.89	insrawch(chtype)	115
5.2.3.90	insstr(const char *)	115
5.2.3.91	instr(char *)	115
5.2.3.92	intrflush(WINDOW *, bool)	115
5.2.3.93	is_linetouched(WINDOW *, int)	115
5.2.3.94	is_termresized(void)	115
5.2.3.95	is_wintouched(WINDOW *)	115
5.2.3.96	isendwin(void)	115
5.2.3.97	keyname(int)	115
5.2.3.98	keypad(WINDOW *, bool)	115
5.2.3.99	killchar(void)	115
5.2.3.100	leaveok(WINDOW *, bool)	115
5.2.3.101	longname(void)	115
5.2.3.102	map_button(unsigned long)	115
5.2.3.103	meta(WINDOW *, bool)	115
5.2.3.104	mouse_off(unsigned long)	115
5.2.3.105	mouse_on(unsigned long)	115
5.2.3.106	mouse_set(unsigned long)	115
5.2.3.107	mouse_trafo(int *, int *, bool)	116
5.2.3.108	mouseinterval(int)	116
5.2.3.109	mousemask(mmask_t, mmask_t *)	116
5.2.3.110	move(int, int)	116

5.2.3.111 mvaddch(int, int, const chtype)	116
5.2.3.112 mvaddchnstr(int, int, const chtype *, int)	116
5.2.3.113 mvaddchstr(int, int, const chtype *)	116
5.2.3.114 mvaddnstr(int, int, const char *, int)	116
5.2.3.115 mvaddrawch(int, int, chtype)	116
5.2.3.116 mvaddstr(int, int, const char *)	116
5.2.3.117 mvchgat(int, int, int, attr_t, short, const void *)	116
5.2.3.118 mvcur(int, int, int, int)	116
5.2.3.119 mvdelch(int, int)	116
5.2.3.120 mvdeleteln(int, int)	116
5.2.3.121 mvderwin(WINDOW *, int, int)	116
5.2.3.122 mvgetch(int, int)	116
5.2.3.123 mvgetnstr(int, int, char *, int)	116
5.2.3.124 mvgetstr(int, int, char *)	117
5.2.3.125 mvhline(int, int, chtype, int)	117
5.2.3.126 mvinch(int, int)	117
5.2.3.127 mvinchnstr(int, int, chtype *, int)	117
5.2.3.128 mvinchstr(int, int, chtype *)	117
5.2.3.129 mvinnstr(int, int, char *, int)	117
5.2.3.130 mvinsch(int, int, chtype)	117
5.2.3.131 mvinsertln(int, int)	117
5.2.3.132 mvinsnstr(int, int, const char *, int)	117
5.2.3.133 mvinsrawch(int, int, chtype)	117
5.2.3.134 mvinsstr(int, int, const char *)	117
5.2.3.135 mvinstr(int, int, char *)	117
5.2.3.136 mvprintw(int, int, const char *,...)	117
5.2.3.137 mvscanw(int, int, const char *,...)	117
5.2.3.138 mvvline(int, int, chtype, int)	117
5.2.3.139 mvwaddch(WINDOW *, int, int, const chtype)	117
5.2.3.140 mvwaddchnstr(WINDOW *, int, int, const chtype *, int)	118
5.2.3.141 mvwaddchstr(WINDOW *, int, int, const chtype *)	118
5.2.3.142 mvwaddnstr(WINDOW *, int, int, const char *, int)	118
5.2.3.143 mvwaddrawch(WINDOW *, int, int, chtype)	118
5.2.3.144 mvwaddstr(WINDOW *, int, int, const char *)	118
5.2.3.145 mvwchgat(WINDOW *, int, int, int, attr_t, short, const void *)	118
5.2.3.146 mvwdelch(WINDOW *, int, int)	118
5.2.3.147 mvwdeleteln(WINDOW *, int, int)	118
5.2.3.148 mvwgetch(WINDOW *, int, int)	118
5.2.3.149 mvwgetnstr(WINDOW *, int, int, char *, int)	118
5.2.3.150 mvwgetstr(WINDOW *, int, int, char *)	118

5.2.3.151 mvwline(WINDOW *, int, int, chtype, int)	118
5.2.3.152 mvwin(WINDOW *, int, int)	118
5.2.3.153 mvwinch(WINDOW *, int, int)	118
5.2.3.154 mvwinchnstr(WINDOW *, int, int, chtype *, int)	118
5.2.3.155 mvwinchstr(WINDOW *, int, int, chtype *)	118
5.2.3.156 mvwinnstr(WINDOW *, int, int, char *, int)	118
5.2.3.157 mvwinsch(WINDOW *, int, int, chtype)	118
5.2.3.158 mvwinserltn(WINDOW *, int, int)	118
5.2.3.159 mvwinsnstr(WINDOW *, int, int, const char *, int)	118
5.2.3.160 mvwinsrawch(WINDOW *, int, int, chtype)	118
5.2.3.161 mvwinsstr(WINDOW *, int, int, const char *)	118
5.2.3.162 mvwinstr(WINDOW *, int, int, char *)	118
5.2.3.163 mvwprintw(WINDOW *, int, int, const char *,...)	118
5.2.3.164 mvwscanw(WINDOW *, int, int, const char *,...)	118
5.2.3.165 mvwvline(WINDOW *, int, int, chtype, int)	118
5.2.3.166 napms(int)	118
5.2.3.167 nc_getmouse(MEVENT *)	118
5.2.3.168 newpad(int, int)	119
5.2.3.169 newterm(const char *, FILE *, FILE *)	119
5.2.3.170 newwin(int, int, int, int)	119
5.2.3.171 nl(void)	119
5.2.3.172 nocbreak(void)	119
5.2.3.173 nocrmode(void)	119
5.2.3.174 nodelay(WINDOW *, bool)	119
5.2.3.175 noecho(void)	119
5.2.3.176 nonl(void)	119
5.2.3.177 noqiflush(void)	119
5.2.3.178 noraw(void)	119
5.2.3.179 notimeout(WINDOW *, bool)	119
5.2.3.180 overlay(const WINDOW *, WINDOW *)	119
5.2.3.181 overwrite(const WINDOW *, WINDOW *)	119
5.2.3.182 pair_content(short, short *, short *)	119
5.2.3.183 PDC_clearclipboard(void)	119
5.2.3.184 PDC_debug(const char *,...)	119
5.2.3.185 PDC_freeclipboard(char *)	119
5.2.3.186 PDC_get_input_fd(void)	119
5.2.3.187 PDC_get_key_modifiers(void)	119
5.2.3.188 PDC_getclipboard(char **, long *)	120
5.2.3.189 PDC_return_key_modifiers(bool)	120
5.2.3.190 PDC_save_key_modifiers(bool)	120

5.2.3.191 PDC_set_blink(bool)	120
5.2.3.192 PDC_set_line_color(short)	120
5.2.3.193 PDC_set_resize_limits(const int new_min_lines, const int new_max_lines, const int new_min_cols, const int new_max_cols)	120
5.2.3.194 PDC_set_title(const char *)	120
5.2.3.195 PDC_setclipboard(const char *, long)	120
5.2.3.196 PDC_ungetch(int)	120
5.2.3.197 pechochar(WINDOW *, chtype)	120
5.2.3.198 pnoutrefresh(WINDOW *, int, int, int, int, int, int)	120
5.2.3.199 prefresh(WINDOW *, int, int, int, int, int, int)	120
5.2.3.200 printw(const char *,...)	120
5.2.3.201 putwin(WINDOW *, FILE *)	120
5.2.3.202 qiflush(void)	120
5.2.3.203 raw(void)	120
5.2.3.204 raw_output(bool)	121
5.2.3.205 redrawwin(WINDOW *)	121
5.2.3.206 refresh(void)	121
5.2.3.207 request_mouse_pos(void)	121
5.2.3.208 reset_prog_mode(void)	121
5.2.3.209 reset_shell_mode(void)	121
5.2.3.210 resetterm(void)	121
5.2.3.211 resetty(void)	121
5.2.3.212 resize_term(int, int)	121
5.2.3.213 resize_window(WINDOW *, int, int)	121
5.2.3.214 ripoffline(int, int*)(WINDOW *, int)	121
5.2.3.215 saveterm(void)	121
5.2.3.216 savetty(void)	121
5.2.3.217 scanw(const char *,...)	122
5.2.3.218 scr_dump(const char *)	122
5.2.3.219 scr_init(const char *)	122
5.2.3.220 scr_restore(const char *)	122
5.2.3.221 scr_set(const char *)	122
5.2.3.222 scr1(int)	122
5.2.3.223 scroll(WINDOW *)	122
5.2.3.224 scrollok(WINDOW *, bool)	122
5.2.3.225 set_term(SCREEN *)	122
5.2.3.226 setscrreg(int, int)	122
5.2.3.227 setsyx(int, int)	122
5.2.3.228 slk_attr_off(const attr_t, void *)	122
5.2.3.229 slk_attr_on(const attr_t, void *)	122

5.2.3.230 slk_attr_set(const attr_t, short, void *)	122
5.2.3.231 slk_atroff(const chtype)	122
5.2.3.232 slk_attron(const chtype)	122
5.2.3.233 slk_attrset(const chtype)	122
5.2.3.234 slk_clear(void)	122
5.2.3.235 slk_color(short)	122
5.2.3.236 slk_init(int)	122
5.2.3.237 slk_label(int)	122
5.2.3.238 slk_noutrefresh(void)	122
5.2.3.239 slk_refresh(void)	123
5.2.3.240 slk_restore(void)	123
5.2.3.241 slk_set(int, const char *, int)	123
5.2.3.242 slk_touch(void)	123
5.2.3.243 standend(void)	123
5.2.3.244 standout(void)	123
5.2.3.245 start_color(void)	123
5.2.3.246 subpad(WINDOW *, int, int, int, int)	123
5.2.3.247 subwin(WINDOW *, int, int, int, int)	123
5.2.3.248 syncok(WINDOW *, bool)	123
5.2.3.249 term_attrs(void)	123
5.2.3.250 termattrs(void)	123
5.2.3.251 termname(void)	123
5.2.3.252 timeout(int)	123
5.2.3.253 touchline(WINDOW *, int, int)	123
5.2.3.254 touchwin(WINDOW *)	123
5.2.3.255 traceoff(void)	123
5.2.3.256 traceon(void)	123
5.2.3.257 typeahead(int)	123
5.2.3.258 unctrl(chtype)	123
5.2.3.259 ungetmouse(MEVENT *)	123
5.2.3.260 untouchwin(WINDOW *)	124
5.2.3.261 use_default_colors(void)	124
5.2.3.262 use_env(bool)	124
5.2.3.263 vid_attr(attr_t, short, void *)	124
5.2.3.264 vid_puts(attr_t, short, void *, int(*)(int))	124
5.2.3.265 vidattr(chtype)	124
5.2.3.266 vidputs(chtype, int(*)(int))	124
5.2.3.267 vline(chtype, int)	124
5.2.3.268 vw_printw(WINDOW *, const char *, va_list)	124
5.2.3.269 vw_scanw(WINDOW *, const char *, va_list)	124

5.2.3.270 vwprintw(WINDOW *, const char *, va_list)	124
5.2.3.271 vwscanw(WINDOW *, const char *, va_list)	124
5.2.3.272 waddch(WINDOW *, const chtype)	124
5.2.3.273 waddchnstr(WINDOW *, const chtype *, int)	124
5.2.3.274 waddchstr(WINDOW *, const chtype *)	124
5.2.3.275 waddnstr(WINDOW *, const char *, int)	124
5.2.3.276 waddrawch(WINDOW *, chtype)	124
5.2.3.277 waddstr(WINDOW *, const char *)	124
5.2.3.278 wattr_get(WINDOW *, attr_t *, short *, void *)	124
5.2.3.279 wattr_off(WINDOW *, attr_t, void *)	124
5.2.3.280 wattr_on(WINDOW *, attr_t, void *)	124
5.2.3.281 wattr_set(WINDOW *, attr_t, short, void *)	124
5.2.3.282 wattroff(WINDOW *, chtype)	124
5.2.3.283 wattron(WINDOW *, chtype)	124
5.2.3.284 wattrset(WINDOW *, chtype)	124
5.2.3.285 wbkgd(WINDOW *, chtype)	124
5.2.3.286 wbkgdset(WINDOW *, chtype)	124
5.2.3.287 wborder(WINDOW *, chtype, chtype, chtype, chtype, chtype, chtype, chtype, chtype)	124
5.2.3.288 wchgat(WINDOW *, int, attr_t, short, const void *)	125
5.2.3.289 wclear(WINDOW *)	125
5.2.3.290 wclrtoebot(WINDOW *)	125
5.2.3.291 wclrtoeol(WINDOW *)	125
5.2.3.292 wcolor_set(WINDOW *, short, void *)	125
5.2.3.293 wcursyncup(WINDOW *)	125
5.2.3.294 wdelch(WINDOW *)	125
5.2.3.295 wdeleteln(WINDOW *)	125
5.2.3.296 wechochar(WINDOW *, const chtype)	125
5.2.3.297 wenclose(const WINDOW *, int, int)	125
5.2.3.298 werase(WINDOW *)	125
5.2.3.299 wgetch(WINDOW *)	125
5.2.3.300 wgetnstr(WINDOW *, char *, int)	125
5.2.3.301 wgetstr(WINDOW *, char *)	125
5.2.3.302 whline(WINDOW *, chtype, int)	125
5.2.3.303 winch(WINDOW *)	125
5.2.3.304 winchnstr(WINDOW *, chtype *, int)	125
5.2.3.305 winchstr(WINDOW *, chtype *)	125
5.2.3.306 winnstr(WINDOW *, char *, int)	125
5.2.3.307 winsch(WINDOW *, chtype)	125
5.2.3.308 winsdelln(WINDOW *, int)	125

5.2.3.309	winsertln(WINDOW *)	125
5.2.3.310	winsnstr(WINDOW *, const char *, int)	125
5.2.3.311	winsrawch(WINDOW *, chtype)	125
5.2.3.312	winsstr(WINDOW *, const char *)	125
5.2.3.313	winstr(WINDOW *, char *)	125
5.2.3.314	wmouse_position(WINDOW *, int *, int *)	125
5.2.3.315	wmouse_trafo(const WINDOW *, int *, int *, bool)	125
5.2.3.316	wmove(WINDOW *, int, int)	126
5.2.3.317	wnoutrefresh(WINDOW *)	126
5.2.3.318	wordchar(void)	126
5.2.3.319	wprintw(WINDOW *, const char *,...)	126
5.2.3.320	wredrawln(WINDOW *, int, int)	126
5.2.3.321	wrefresh(WINDOW *)	126
5.2.3.322	wresize(WINDOW *, int, int)	126
5.2.3.323	wscanw(WINDOW *, const char *,...)	126
5.2.3.324	wscrln(WINDOW *, int)	126
5.2.3.325	wsetscrreg(WINDOW *, int, int)	126
5.2.3.326	wstandend(WINDOW *)	126
5.2.3.327	wstandout(WINDOW *)	126
5.2.3.328	wsyncdown(WINDOW *)	126
5.2.3.329	wsyncup(WINDOW *)	126
5.2.3.330	wtimeout(WINDOW *, int)	126
5.2.3.331	wtouchln(WINDOW *, int, int, int)	126
5.2.3.332	wvline(WINDOW *, chtype, int)	126
5.2.3.333	Xinitscr(int, char **)	126
5.2.4	Variable Documentation	126
5.2.4.1	acs_map	126
5.2.4.2	COLOR_PAIRS	126
5.2.4.3	COLORS	126
5.2.4.4	COLS	126
5.2.4.5	curscr	127
5.2.4.6	LINES	127
5.2.4.7	Mouse_status	127
5.2.4.8	SP	127
5.2.4.9	stdscr	127
5.2.4.10	TABSIZE	127
5.2.4.11	ttytype	127
5.3	README.md File Reference	127

Chapter 1

2048.hackable.c

An hackable 2048 game written in C

Chapter 2

Data Structure Index

2.1 Data Structures

Here are the data structures with brief descriptions:

_win	7
MEVENT	10
MOUSE_STATUS	11
SCREEN	12

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

2048.c	17
ncurses.h	47

Chapter 4

Data Structure Documentation

4.1 `_win` Struct Reference

```
#include <ncurses.h>
```

Collaboration diagram for `_win`:



Data Fields

- `int _cury`
- `int _curx`
- `int _maxy`
- `int _maxx`
- `int _begy`
- `int _begx`
- `int _flags`
- `chtype _attrs`
- `chtype _bkgd`
- `bool _clear`
- `bool _leaveit`
- `bool _scroll`
- `bool _nodelay`
- `bool _immed`
- `bool _sync`
- `bool _use_keypad`
- `chtype ** _y`
- `int * _firstch`
- `int * _lastch`
- `int _tmarg`
- `int _bmarg`

- `int _delayms`
- `int _parx`
- `int _pary`
- `struct _win * _parent`

4.1.1 Detailed Description

Definition at line 269 of file ncurses.h.

4.1.2 Field Documentation

4.1.2.1 `chtype_attrs`

Definition at line 278 of file ncurses.h.

4.1.2.2 `int_begx`

Definition at line 276 of file ncurses.h.

4.1.2.3 `int_begy`

Definition at line 275 of file ncurses.h.

4.1.2.4 `chtype_bkgd`

Definition at line 279 of file ncurses.h.

4.1.2.5 `int_bmarg`

Definition at line 291 of file ncurses.h.

4.1.2.6 `bool_clear`

Definition at line 280 of file ncurses.h.

4.1.2.7 `int_curx`

Definition at line 272 of file ncurses.h.

4.1.2.8 `int_cury`

Definition at line 271 of file ncurses.h.

4.1.2.9 `int_delayms`

Definition at line 292 of file ncurses.h.

4.1.2.10 int* _firstch

Definition at line 288 of file ncurses.h.

4.1.2.11 int _flags

Definition at line 277 of file ncurses.h.

4.1.2.12 bool _immed

Definition at line 284 of file ncurses.h.

4.1.2.13 int* _lastch

Definition at line 289 of file ncurses.h.

4.1.2.14 bool _leaveit

Definition at line 281 of file ncurses.h.

4.1.2.15 int _maxx

Definition at line 274 of file ncurses.h.

4.1.2.16 int _maxy

Definition at line 273 of file ncurses.h.

4.1.2.17 bool _nodelay

Definition at line 283 of file ncurses.h.

4.1.2.18 struct _win* _parent

Definition at line 294 of file ncurses.h.

4.1.2.19 int _parx

Definition at line 293 of file ncurses.h.

4.1.2.20 int _pary

Definition at line 293 of file ncurses.h.

4.1.2.21 bool _scroll

Definition at line 282 of file ncurses.h.

4.1.2.22 `bool_sync`

Definition at line 285 of file `ncurses.h`.

4.1.2.23 `int_tmarg`

Definition at line 290 of file `ncurses.h`.

4.1.2.24 `bool_use_keypad`

Definition at line 286 of file `ncurses.h`.

4.1.2.25 `chtype**_y`

Definition at line 287 of file `ncurses.h`.

The documentation for this struct was generated from the following file:

- [ncurses.h](#)

4.2 MEVENT Struct Reference

```
#include <ncurses.h>
```

Data Fields

- short [id](#)
- int [x](#)
- int [y](#)
- int [z](#)
- [mmask_t](#) [bstate](#)

4.2.1 Detailed Description

Definition at line 244 of file `ncurses.h`.

4.2.2 Field Documentation

4.2.2.1 `mmask_t bstate`

Definition at line 248 of file `ncurses.h`.

4.2.2.2 `short id`

Definition at line 246 of file `ncurses.h`.

4.2.2.3 `int x`

Definition at line 247 of file `ncurses.h`.

4.2.2.4 int y

Definition at line 247 of file ncurses.h.

4.2.2.5 int z

Definition at line 247 of file ncurses.h.

The documentation for this struct was generated from the following file:

- [ncurses.h](#)

4.3 MOUSE_STATUS Struct Reference

```
#include <ncurses.h>
```

Data Fields

- int [x](#)
- int [y](#)
- short [button](#) [3]
- int [changes](#)
- short [xbutton](#) [[PDC_N_EXTENDED_MOUSE_BUTTONS](#)]

4.3.1 Detailed Description

Definition at line 128 of file ncurses.h.

4.3.2 Field Documentation

4.3.2.1 short button[3]

Definition at line 132 of file ncurses.h.

4.3.2.2 int changes

Definition at line 133 of file ncurses.h.

4.3.2.3 int x

Definition at line 130 of file ncurses.h.

4.3.2.4 short xbutton[PDC_N_EXTENDED_MOUSE_BUTTONS]

Definition at line 134 of file ncurses.h.

4.3.2.5 int y

Definition at line 131 of file ncurses.h.

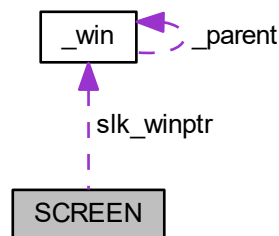
The documentation for this struct was generated from the following file:

- [ncurses.h](#)

4.4 SCREEN Struct Reference

```
#include <ncurses.h>
```

Collaboration diagram for SCREEN:



Data Fields

- [bool alive](#)
- [bool autocr](#)
- [bool cbreak](#)
- [bool echo](#)
- [bool raw_inp](#)
- [bool raw_out](#)
- [bool audible](#)
- [bool mono](#)
- [bool resized](#)
- [bool orig_attr](#)
- [short orig_fore](#)
- [short orig_back](#)
- [int cursrow](#)
- [int curscol](#)
- [int visibility](#)
- [int orig_cursor](#)
- [int lines](#)
- [int cols](#)
- [unsigned long _trap_mbe](#)
- [unsigned long _map_mbe_to_key](#)
- [int mouse_wait](#)
- [int slklines](#)
- [WINDOW * slk_winptr](#)

- int [linesrippedoff](#)
- int [linesrippedoffontop](#)
- int [delaytenths](#)
- bool [_preserve](#)
- int [_restore](#)
- bool [save_key_modifiers](#)
- bool [return_key_modifiers](#)
- bool [key_code](#)
- short [line_color](#)

4.4.1 Detailed Description

Definition at line 300 of file ncurses.h.

4.4.2 Field Documentation

4.4.2.1 unsigned long _map_mbe_to_key

Definition at line 321 of file ncurses.h.

4.4.2.2 bool _preserve

Definition at line 332 of file ncurses.h.

4.4.2.3 int _restore

Definition at line 334 of file ncurses.h.

4.4.2.4 unsigned long _trap_mbe

Definition at line 320 of file ncurses.h.

4.4.2.5 bool alive

Definition at line 302 of file ncurses.h.

4.4.2.6 bool audible

Definition at line 308 of file ncurses.h.

4.4.2.7 bool autocr

Definition at line 303 of file ncurses.h.

4.4.2.8 bool cbreak

Definition at line 304 of file ncurses.h.

4.4.2.9 int cols

Definition at line 319 of file ncurses.h.

4.4.2.10 int curscol

Definition at line 315 of file ncurses.h.

4.4.2.11 int cursrow

Definition at line 314 of file ncurses.h.

4.4.2.12 int delaytenths

Definition at line 330 of file ncurses.h.

4.4.2.13 bool echo

Definition at line 305 of file ncurses.h.

4.4.2.14 bool key_code

Definition at line 340 of file ncurses.h.

4.4.2.15 short line_color

Definition at line 352 of file ncurses.h.

4.4.2.16 int lines

Definition at line 318 of file ncurses.h.

4.4.2.17 int linesrippedoff

Definition at line 327 of file ncurses.h.

4.4.2.18 int linesrippedoffontop

Definition at line 328 of file ncurses.h.

4.4.2.19 bool mono

Definition at line 309 of file ncurses.h.

4.4.2.20 int mouse_wait

Definition at line 322 of file ncurses.h.

4.4.2.21 bool orig_attr

Definition at line 311 of file ncurses.h.

4.4.2.22 short orig_back

Definition at line 313 of file ncurses.h.

4.4.2.23 int orig_cursor

Definition at line 317 of file ncurses.h.

4.4.2.24 short orig_fore

Definition at line 312 of file ncurses.h.

4.4.2.25 bool raw_inp

Definition at line 306 of file ncurses.h.

4.4.2.26 bool raw_out

Definition at line 307 of file ncurses.h.

4.4.2.27 bool resized

Definition at line 310 of file ncurses.h.

4.4.2.28 bool return_key_modifiers

Definition at line 338 of file ncurses.h.

4.4.2.29 bool save_key_modifiers

Definition at line 336 of file ncurses.h.

4.4.2.30 WINDOW* slk_winptr

Definition at line 326 of file ncurses.h.

4.4.2.31 int slklines

Definition at line 325 of file ncurses.h.

4.4.2.32 int visibility

Definition at line 316 of file ncurses.h.

The documentation for this struct was generated from the following file:

- [ncurses.h](#)

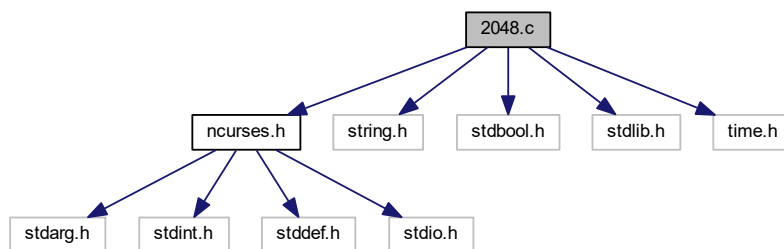
Chapter 5

File Documentation

5.1 2048.c File Reference

```
#include <ncurses.h>
#include <string.h>
#include <stdbool.h>
#include <stdlib.h>
#include <time.h>
```

Include dependency graph for 2048.c:



Macros

- #define PWD ":2048"
- #define PWD_LEN 5
- #define MAX_BOARD_NUM 16
- #define MAX_BOARD_SIZE 16
- #define EUP false,1
- #define EDOWN false,-1
- #define ELEFT true,1
- #define ERIGHT true,-1
- #define MENU_POSITION_Y row-1
- #define MENU_POSITION_X 0
- #define WARNING_POSITION_Y row-2
- #define WARNING_POSITION_X 0

Functions

- void `settings` ()
Set the global settings.
- char `AlignCol` (int curcol, int direction)
Align the vertical direction.
- char `AlignLine` (int curline, int direction)
Align the horizontal direction.
- char `CheckEat` (char *a, char *b)
Check while the two number can be eaten.
- void `Clrboard` (int boardToClr)
Empty the board specified.
- char * `Display` (char in)
Get the grid's display string.
- char `Eat` (bool isH, int direction)
Eat the board at the direction specified.
- char `EatCol` (int curcol, int direction)
Eat the vertical direction.
- char `EatLine` (const int curline, int direction)
Eat the horizontal direction.
- int `GetRandNums` ()
Get random grid ranged from 0 to MAX_RANDOMUM on board.
- unsigned int `Rando` (int N)
Generate random num ranged from 0 to N-1.
- void `command` ()
Show and handle commands inputed by :
- void `die` ()
Handle when no empty grid present.
- void `play` ()
Handle for main game.
- void `showBoard` (int offy, int offx)
Print the board to screen.
- void `welcome` ()
Print welcome message and input the size of the board.
- int `c_checksum` ()
Calculate the checksum for saving.
- void `c_currentStr` (bool show)
Genetate the string representing current board.
- void `c_forceQuit` ()
Quit the game.
- void `c_loadStr` ()
- void `c_readBoard` (int from)
Read the saved board.
- void `c_readFromDisk` (int boards)
Read the saved file.
- void `c_saveBoard` (int to, bool jmp)
Save the board in memory.
- void `c_tryQuit` ()
Ask player whether to quit.
- void `c_warning` (char *warn)
Print a warning to screen.

- `bool c_writeBoardToDisk` (char boards)
Write the board to disk.
- `void c_loadStr` (int iptN, FILE *fp)
Load the string representing saved board.
- `int main` ()
Main executable.

Variables

- `const int NA` =127
- `int P_RANDOMUM` =30
- `int MAX_RANDOMUM` =2
- `const char cs_pwd` [PWD_LEN+1] =PWD
- `char board` [MAX_BOARD_NUM][MAX_BOARD_SIZE][MAX_BOARD_SIZE]
- `char boardstr` [MAX_BOARD_SIZE][MAX_BOARD_SIZE][5]
- `int boardseed` [MAX_BOARD_NUM]
- `unsigned char curs` =0
- `char eat` [256][256][2]
Eat table(TODO:use eat array in CheckEat)
- `char display` [256][16]
- `int point` [256]
Point table(TODO:use eat array in CheckEat)
- `char N` =5
- `int row`
- `int col`

5.1.1 Macro Definition Documentation

5.1.1.1 #define EDOWN false,-1

Eat down

Definition at line 40 of file 2048.c.

5.1.1.2 #define ELEFT true,1

Eat left

Definition at line 41 of file 2048.c.

5.1.1.3 #define ERIGHT true,-1

Rat right

Definition at line 42 of file 2048.c.

5.1.1.4 #define EUP false,1

Eat up

Definition at line 39 of file 2048.c.

5.1.1.5 #define MAX_BOARD_NUM 16

The maxium board num (for saving in game)

Definition at line 20 of file 2048.c.

5.1.1.6 #define MAX_BOARD_SIZE 16

The maxium board size

Definition at line 24 of file 2048.c.

5.1.1.7 #define MENU_POSITION_X 0

the X position to print menu

Definition at line 55 of file 2048.c.

5.1.1.8 #define MENU_POSITION_Y row-1

the Y position to print menu

Definition at line 54 of file 2048.c.

5.1.1.9 #define PWD ":2048"

The password for the save file and represent the version of the game Should and only be changed when the saved file isn't/shouldn't compatible with others

Definition at line 5 of file 2048.c.

5.1.1.10 #define PWD_LEN 5

The length of PWD

Definition at line 9 of file 2048.c.

5.1.1.11 #define WARNING_POSITION_X 0

the X position to print warning

Definition at line 57 of file 2048.c.

5.1.1.12 #define WARNING_POSITION_Y row-2

the Y position to print warning

Definition at line 56 of file 2048.c.

5.1.2 Function Documentation**5.1.2.1 char AlignCol (int *curcol*, int *direction*)**

Align the vertical direction.

Parameters

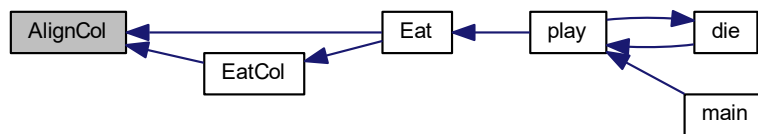
<i>curcol</i>	Current column to align
<i>direction</i>	The direction to align to positive for up and negative for down

Returns

The number of blank grid in the column

Definition at line 169 of file 2048.c.

Here is the caller graph for this function:



5.1.2.2 char AlignLine (int curline, int direction)

Align the horizontal direction.

Parameters

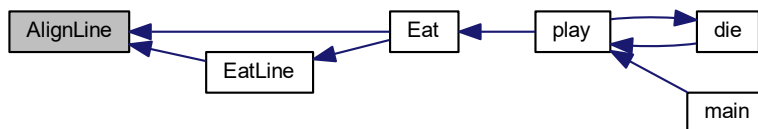
<i>curline</i>	Current line to align
<i>direction</i>	The direction to align to positive for left and negative for right

Returns

The number of blank grid in the column

Definition at line 198 of file 2048.c.

Here is the caller graph for this function:



5.1.2.3 int c_checksum ()

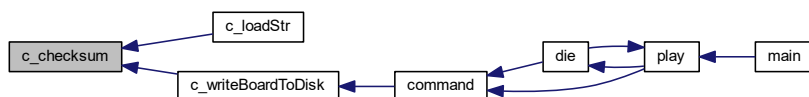
Calculate the checksum for saving.

Returns

The checksum for current board

Definition at line 492 of file 2048.c.

Here is the caller graph for this function:

**5.1.2.4 void c_currentStr (bool show)**

Genetate the string representing current board.

Parameters

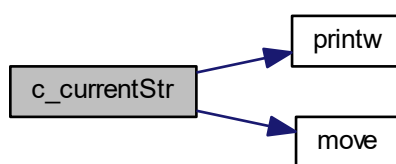
<i>show</i>	If need to print the string to screen
-------------	---------------------------------------

Returns

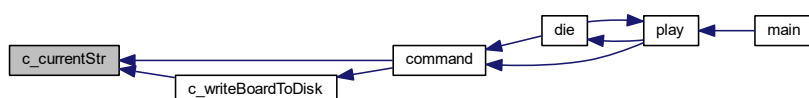
void

Definition at line 517 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.5 void c_forceQuit ()

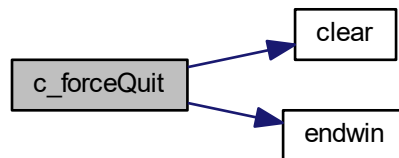
Quit the game.

Returns

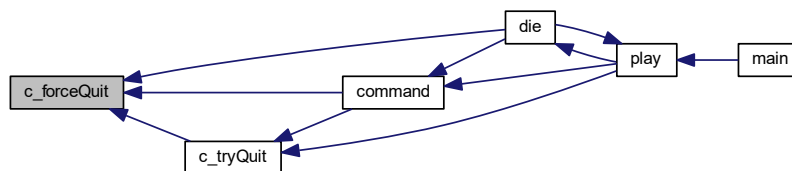
void

Definition at line 546 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.6 void c_loadStr ()

Here is the caller graph for this function:



5.1.2.7 void c_loadStr (int iptN, FILE * fp)

Load the string representing saved board.

Parameters

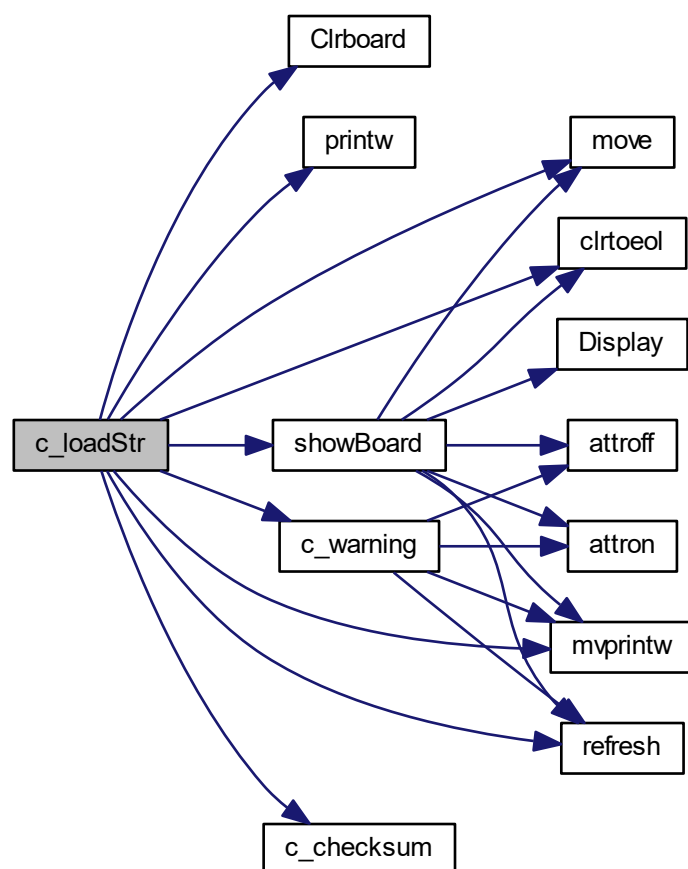
<i>iptN</i>	The N in the saved game
<i>fp</i>	The file stream to read from

Returns

void

Definition at line 555 of file 2048.c.

Here is the call graph for this function:



5.1.2.8 void c_readBoard (int from)

Read the saved board.

Parameters

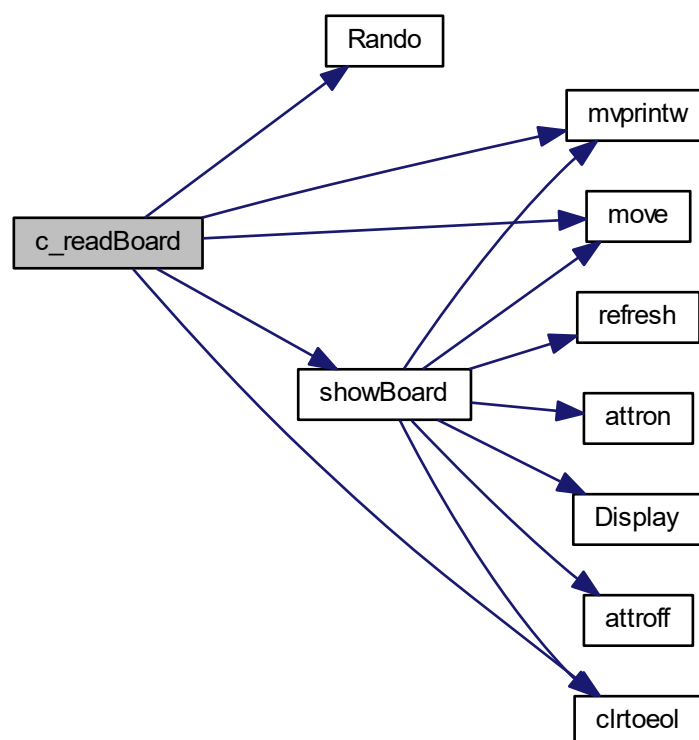
<i>from</i>	The number of board to read from
-------------	----------------------------------

Returns

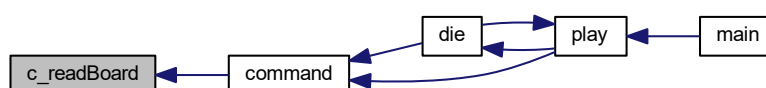
void

Definition at line 602 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.9 void c_readFromDisk (int boards)

Read the saved file.

Parameters

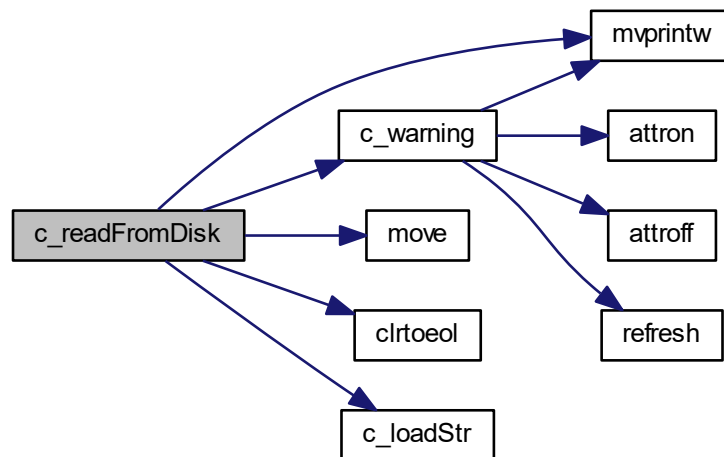
<i>boards</i>	The number of the saved board.NA for not to use
---------------	---

Returns

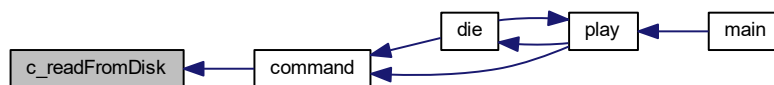
void

Definition at line 619 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.10 void c_saveBoard (int to, bool jmp)

Save the board in memory.

Parameters

<i>to</i>	The number of the board to save to.NA for auto find nnext
-----------	---

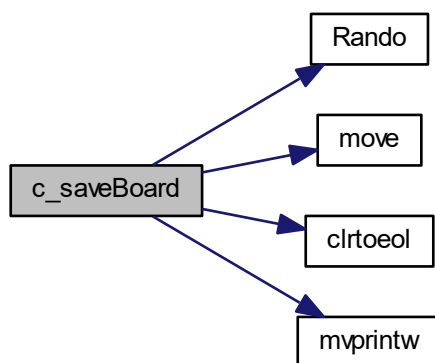
<i>jmp</i>	If should jump to new board
------------	-----------------------------

Returns

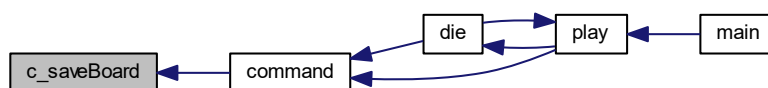
void

Definition at line 657 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:

**5.1.2.11 void c_tryQuit ()**

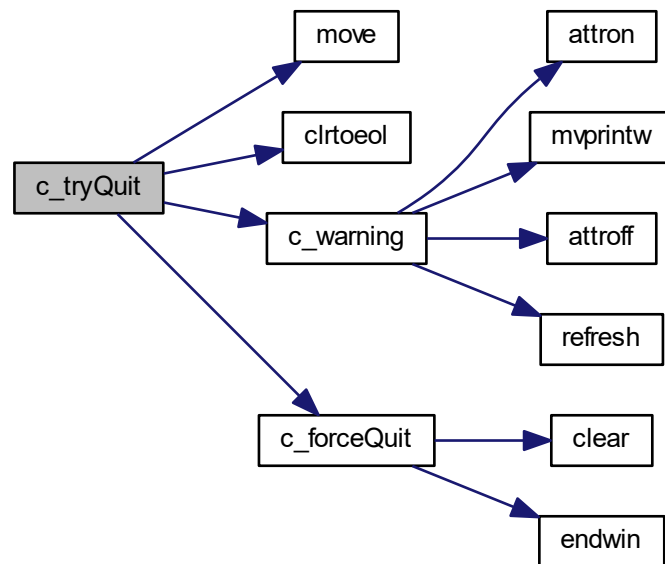
Ask player whether to quit.

Returns

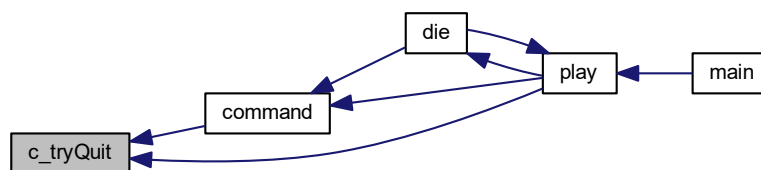
void

Definition at line 671 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:

**5.1.2.12 void c_warning (char * warn)**

Print a warning to screen.

Parameters

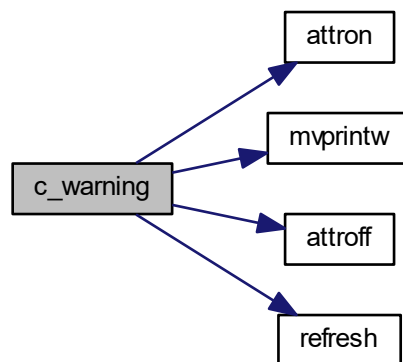
<i>warn</i>	The string to print
-------------	---------------------

Returns

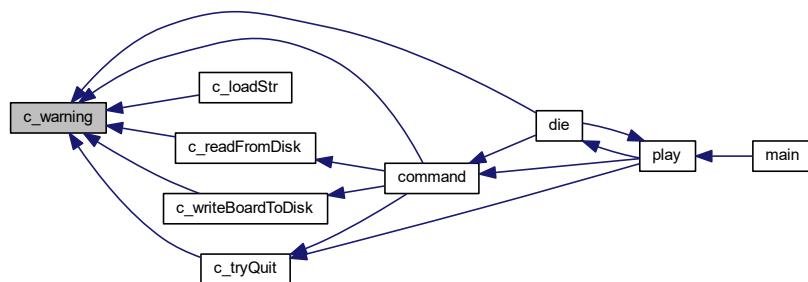
void

Definition at line 746 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:

**5.1.2.13 bool c_writeBoardToDisk (char *boards*)**

Write the board to disk.

Parameters

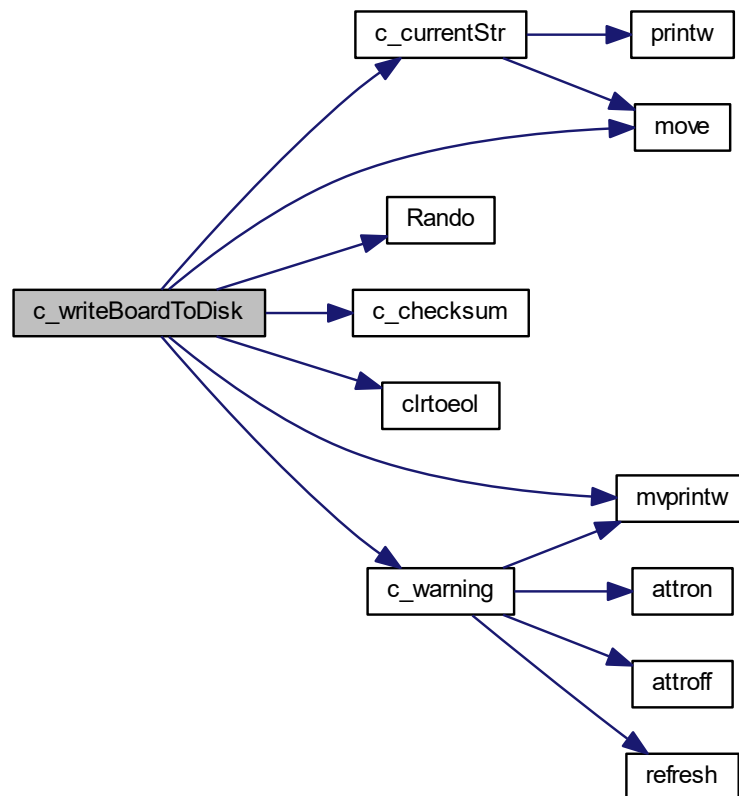
<i>boards</i>	The number of board to save
---------------	-----------------------------

Returns

Whether the file is saved successfully

Definition at line 694 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.14 char CheckEat (char * a, char * b)

Check while the two number can be eaten.

Parameters

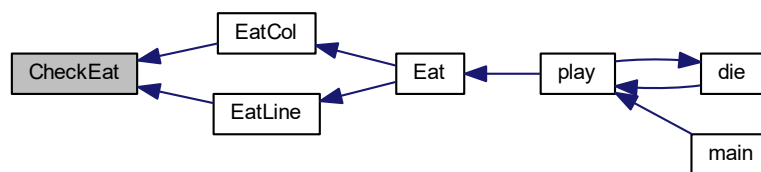
<i>a</i>	Value a
<i>b</i>	Value b

Returns

The final value of a

Definition at line 226 of file 2048.c.

Here is the caller graph for this function:

5.1.2.15 void Clrboard (int *boardToClr*)

Empty the board specified.

Parameters

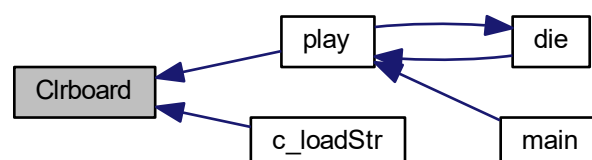
<i>boardToClr</i>	The board to empty
-------------------	--------------------

Returns

void

Definition at line 236 of file 2048.c.

Here is the caller graph for this function:



5.1.2.16 void command ()

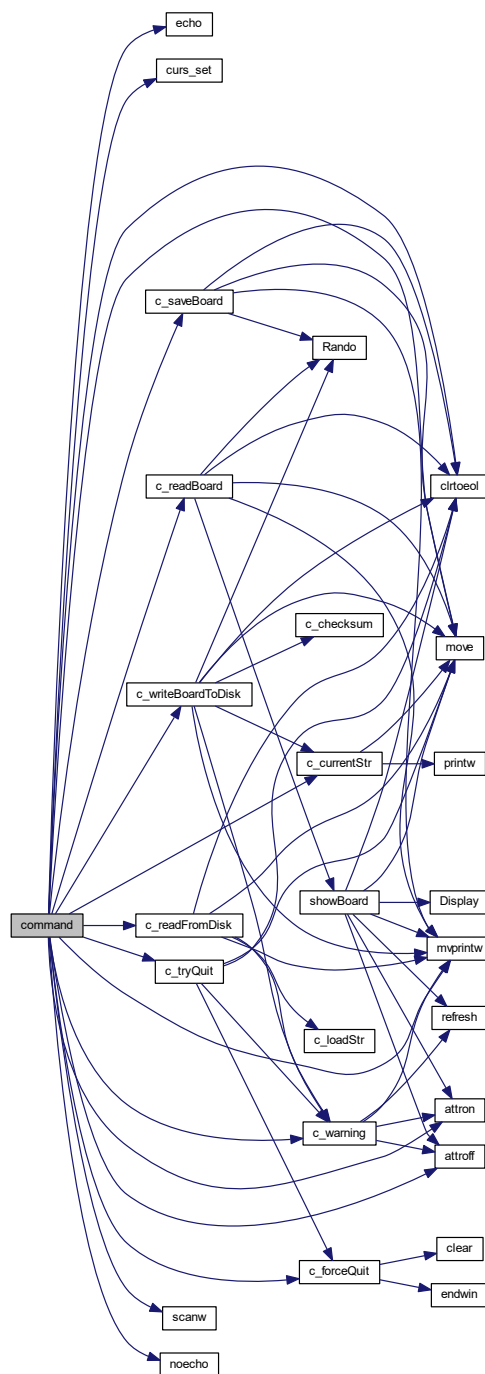
Show and handle commands inputed by :

Returns

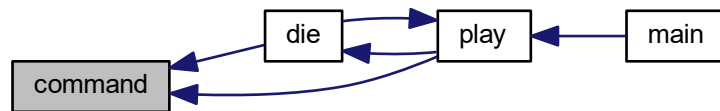
void

Definition at line 336 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.17 void die ()

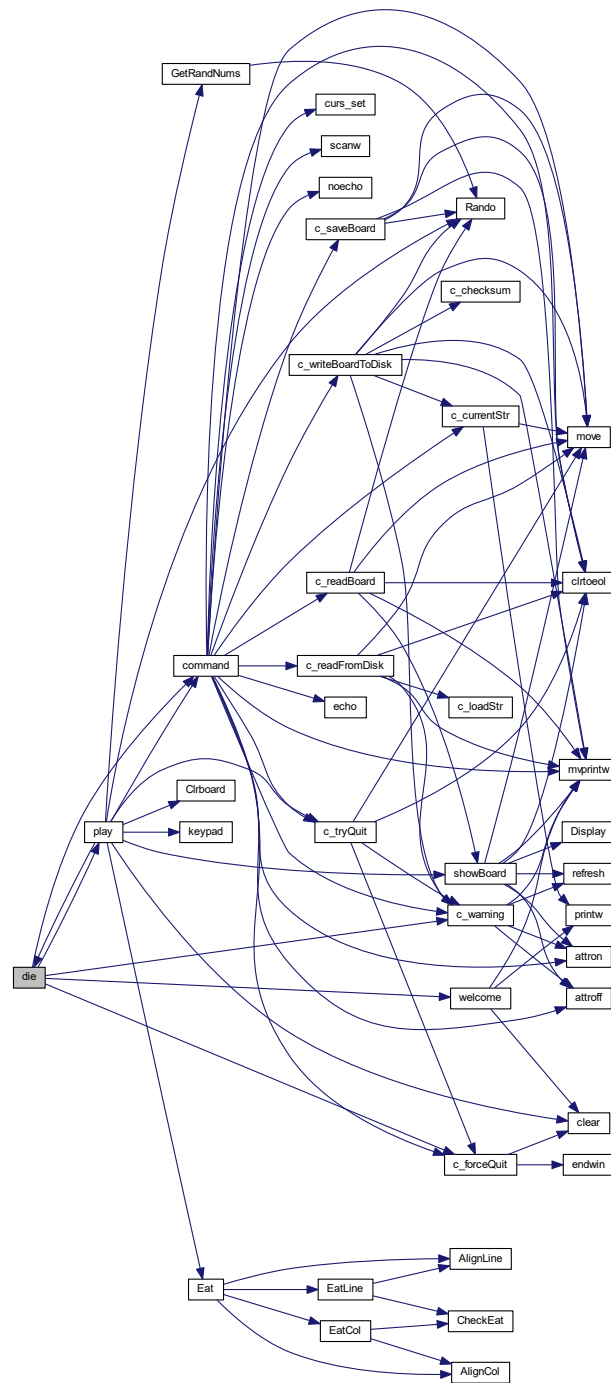
Handle when no empty grid present.

Returns

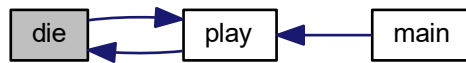
void

Definition at line 380 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.18 char * Display (char *in*)

Get the grid's display string.

Parameters

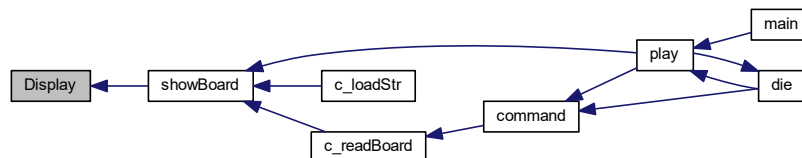
<i>in</i>	The grid's value
-----------	------------------

Returns

The string to display

Definition at line 242 of file 2048.c.

Here is the caller graph for this function:



5.1.2.19 char Eat (bool *isH*, int *direction*)

Eat the board at the direction specified.

Parameters

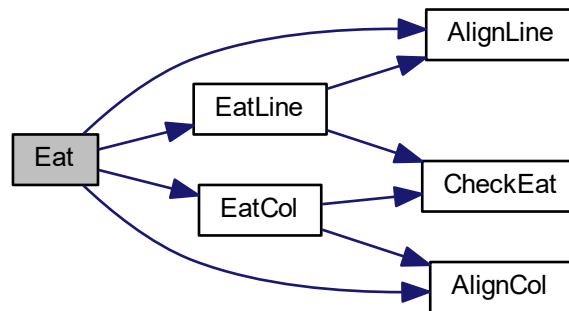
<i>isH</i>	Is horizontal true for horizontal and false for vertical
<i>direction</i>	The direction to eat to positive for left/up and negative for right/down

Returns

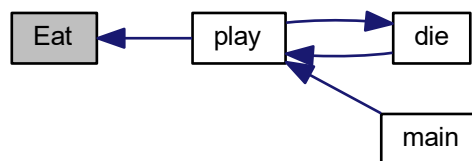
The number of empty grids

Definition at line 252 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.20 char EatCol (int *curcol*, int *direction*)

Eat the vertical direction.

Parameters

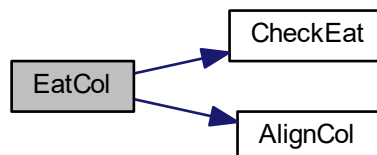
<i>curcol</i>	Current column to eat
<i>direction</i>	The direction to eat positive for up and negative for down

Returns

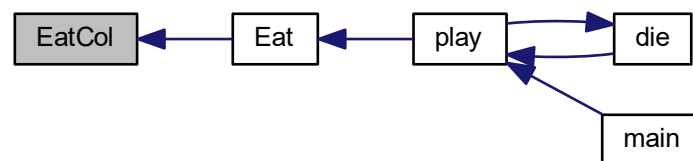
The number of blank grid in the column

Definition at line 272 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.21 char EatLine (const int *curline*, int *direction*)

Eat the horizontal direction.

Parameters

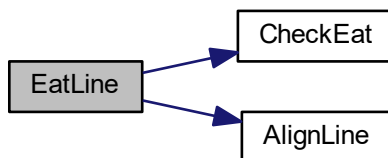
<i>curline</i>	Current line to eat
<i>direction</i>	The direction to eat positive for left and negative for right

Returns

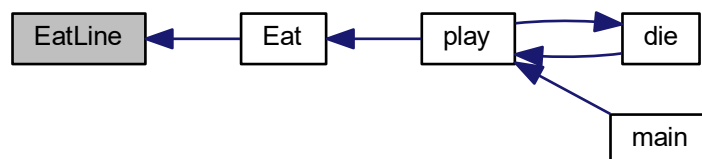
The number of blank grid in the column

Definition at line 286 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:

**5.1.2.22 int GetRandNums ()**

Get random grid ranged from 0 to MAX_RANDOMUM on board.

Returns

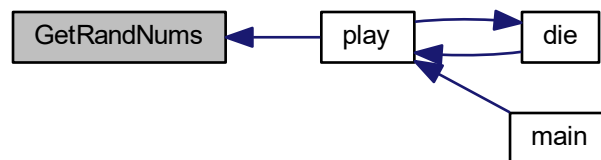
The number of random grid generated

Definition at line 297 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.23 `int main ()`

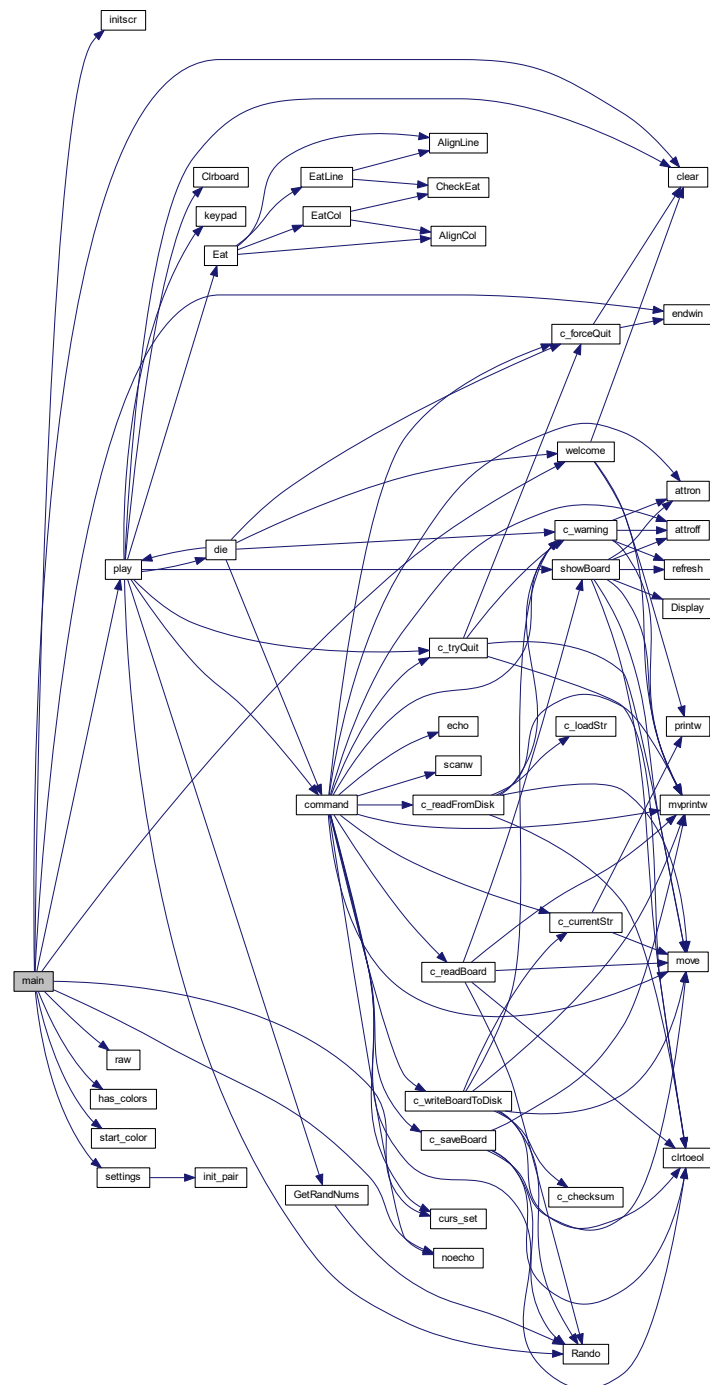
Main executable.

Returns

0

Definition at line 754 of file 2048.c.

Here is the call graph for this function:



5.1.2.24 void play ()

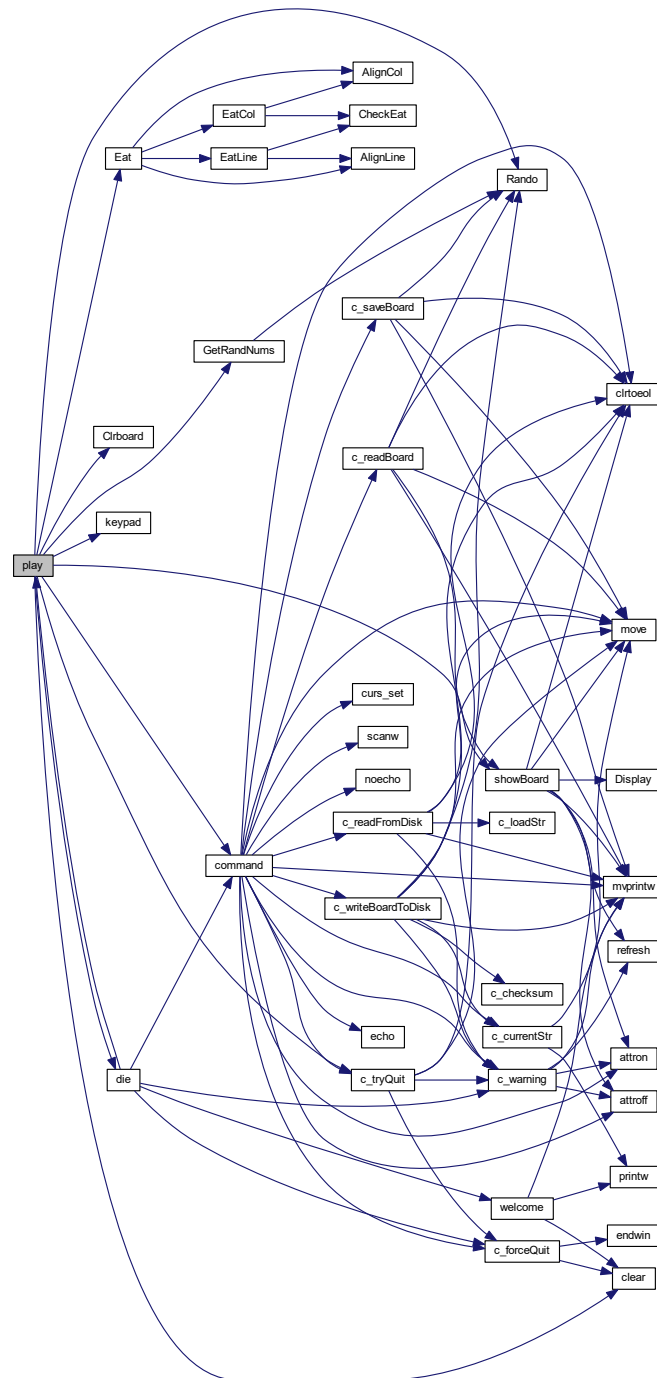
Handle for main game.

Returns

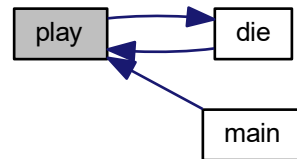
void

Definition at line 402 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.25 unsigned int Rando (int *N*)

Generate random num ranged from 0 to *N*-1.

Parameters

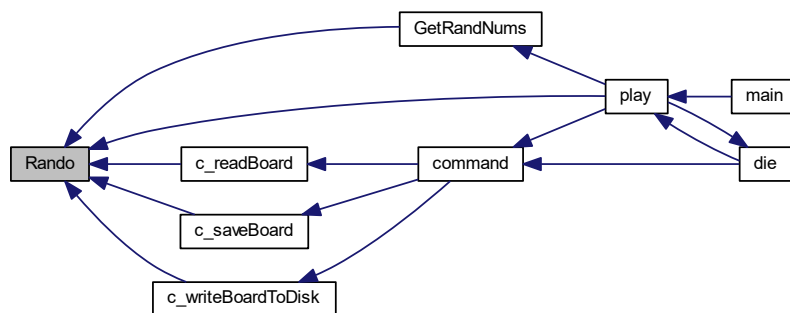
<i>N</i>	The upper bound of the random number
----------	--------------------------------------

Returns

The random number

Definition at line 309 of file 2048.c.

Here is the caller graph for this function:



5.1.2.26 void settings ()

Set the global settings.

TODO:use a ini instead?

Returns

void

Definition at line 92 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:

**5.1.2.27 void showBoard (int *offy*, int *offx*)**

Print the board to screen.

Parameters

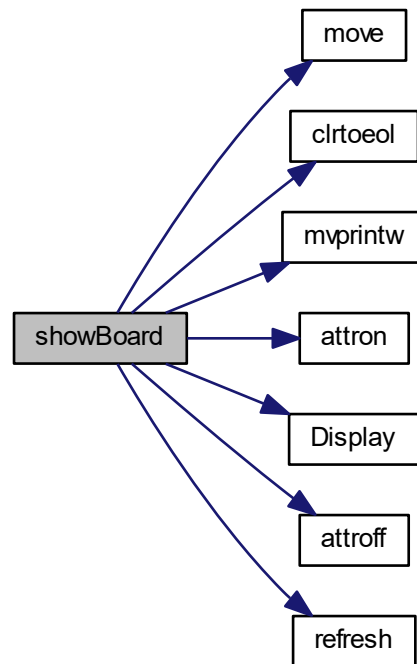
<i>offy</i>	The y position for the left-up corner
<i>offx</i>	The x position for the left-up corner

Returns

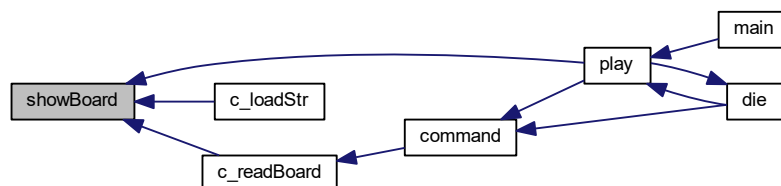
void

Definition at line 443 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.28 void welcome ()

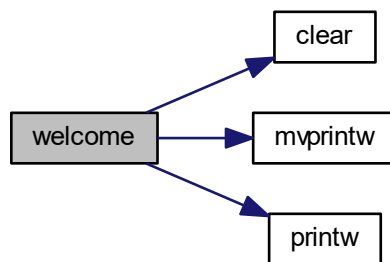
Print welcome message and input the size of the board.

Returns

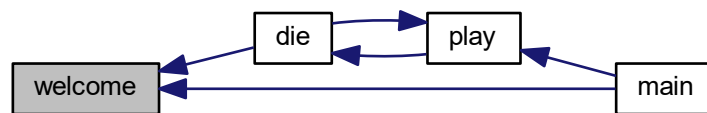
void

Definition at line 470 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:

**5.1.3 Variable Documentation****5.1.3.1 char board[MAX_BOARD_NUM][MAX_BOARD_SIZE][MAX_BOARD_SIZE]**

The boards to storage game progress

Definition at line 61 of file 2048.c.

5.1.3.2 int boardseed[MAX_BOARD_NUM]

The random seed of the boards

Definition at line 65 of file 2048.c.

5.1.3.3 char boardstr[MAX_BOARD_SIZE][MAX_BOARD_SIZE][5]

The output string

Definition at line 63 of file 2048.c.

5.1.3.4 int col

Definition at line 86 of file 2048.c.

5.1.3.5 const char cs_pwd[PWD_LEN+1] =PWD

The password when generating the checksum

Definition at line 59 of file 2048.c.

5.1.3.6 unsigned char curs =0

Current board

Definition at line 67 of file 2048.c.

5.1.3.7 char display[256][16]

Display table

Will display a as string display[a]

Definition at line 77 of file 2048.c.

5.1.3.8 char eat[256][256][2]

Eat table(TODO:use eat array in CheckEat)

Will set a=eat[a][b][0] and b=eat[a][b][1] when eating a and b

Definition at line 73 of file 2048.c.

5.1.3.9 int MAX_RANDOM =2

The maxium level of filling an grid.

Definition at line 26 of file 2048.c.

5.1.3.10 char N =5

The size of the board

Definition at line 83 of file 2048.c.

5.1.3.11 const int NA =127

Stand for invalid grid.

Definition at line 16 of file 2048.c.

5.1.3.12 int P_RANDOM =30

The probability of an empty grid becoming filled

Definition at line 25 of file 2048.c.

5.1.3.13 int point[256]

Point table(TODO:use eat array in CheckEat)

Will count a by point[a] when adding up the points

Definition at line 81 of file 2048.c.

5.1.3.14 int row

The size of the screen

Definition at line 86 of file 2048.c.

5.2 ncurses.h File Reference

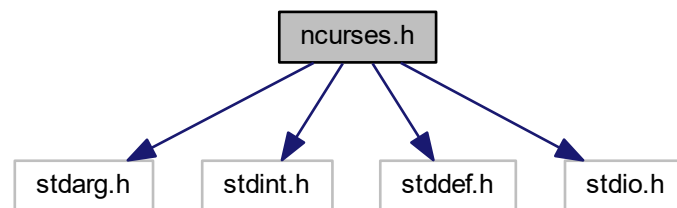
```
#include <stdarg.h>
```

```
#include <stdint.h>
```

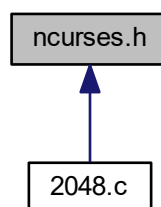
```
#include <stddef.h>
```

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

Include dependency graph for ncurses.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [MOUSE_STATUS](#)

- struct [MEVENT](#)
- struct [_win](#)
- struct [SCREEN](#)

Macros

- #define [PDC_BUILD](#) 3401
- #define [PDCURSES](#) 1 /* PDCurses-only routines */
- #define [XOPEN](#) 1 /* X/Open Curses routines */
- #define [SYSVcurses](#) 1 /* System V Curses routines */
- #define [BSDcurses](#) 1 /* BSD Curses routines */
- #define [CHTYPE_LONG](#) 2 /* chtypes will be (default) 64 bits */
- #define [FALSE](#) 0
- #define [TRUE](#) 1
- #define [NULL](#) (void *)0
- #define [ERR](#) (-1)
- #define [OK](#) 0
- #define [PDC_MAX_MOUSE_BUTTONS](#) 9
- #define [PDC_N_EXTENDED_MOUSE_BUTTONS](#) 6
- #define [BUTTON_RELEASED](#) 0x0000
- #define [BUTTON_PRESSED](#) 0x0001
- #define [BUTTON_CLICKED](#) 0x0002
- #define [BUTTON_DOUBLE_CLICKED](#) 0x0003
- #define [BUTTON_TRIPLE_CLICKED](#) 0x0004
- #define [BUTTON_MOVED](#) 0x0005 /* PDCurses */
- #define [WHEEL_SCROLLED](#) 0x0006 /* PDCurses */
- #define [BUTTON_ACTION_MASK](#) 0x0007 /* PDCurses */
- #define [PDC_BUTTON_SHIFT](#) 0x0008 /* PDCurses */
- #define [PDC_BUTTON_CONTROL](#) 0x0010 /* PDCurses */
- #define [PDC_BUTTON_ALT](#) 0x0020 /* PDCurses */
- #define [BUTTON_MODIFIER_MASK](#) 0x0038 /* PDCurses */
- #define [MOUSE_X_POS](#) (Mouse_status.x)
- #define [MOUSE_Y_POS](#) (Mouse_status.y)
- #define [PDC_MOUSE_MOVED](#) 0x0008
- #define [PDC_MOUSE_POSITION](#) 0x0010
- #define [PDC_MOUSE_WHEEL_UP](#) 0x0020
- #define [PDC_MOUSE_WHEEL_DOWN](#) 0x0040
- #define [PDC_MOUSE_WHEEL_LEFT](#) 0x0080
- #define [PDC_MOUSE_WHEEL_RIGHT](#) 0x0100
- #define [A_BUTTON_CHANGED](#) (Mouse_status.changes & 7)
- #define [MOUSE_MOVED](#) (Mouse_status.changes & [PDC_MOUSE_MOVED](#))
- #define [MOUSE_POS_REPORT](#) (Mouse_status.changes & [PDC_MOUSE_POSITION](#))
- #define [BUTTON_CHANGED](#)(x) (Mouse_status.changes & (1 << ((x) - ((x)<4 ? 1 : -5))))
- #define [BUTTON_STATUS](#)(x) (Mouse_status.button[(x) - 1])
- #define [MOUSE_WHEEL_UP](#) (Mouse_status.changes & [PDC_MOUSE_WHEEL_UP](#))
- #define [MOUSE_WHEEL_DOWN](#) (Mouse_status.changes & [PDC_MOUSE_WHEEL_DOWN](#))
- #define [MOUSE_WHEEL_LEFT](#) (Mouse_status.changes & [PDC_MOUSE_WHEEL_LEFT](#))
- #define [MOUSE_WHEEL_RIGHT](#) (Mouse_status.changes & [PDC_MOUSE_WHEEL_RIGHT](#))
- #define [BUTTON1_RELEASED](#) 0x00000001L
- #define [BUTTON1_PRESSED](#) 0x00000002L
- #define [BUTTON1_CLICKED](#) 0x00000004L
- #define [BUTTON1_DOUBLE_CLICKED](#) 0x00000008L
- #define [BUTTON1_TRIPLE_CLICKED](#) 0x00000010L
- #define [BUTTON1_MOVED](#) 0x00000010L /* PDCurses */

- #define `BUTTON2_RELEASED` 0x00000020L
- #define `BUTTON2_PRESSED` 0x00000040L
- #define `BUTTON2_CLICKED` 0x00000080L
- #define `BUTTON2_DOUBLE_CLICKED` 0x00000100L
- #define `BUTTON2_TRIPLE_CLICKED` 0x00000200L
- #define `BUTTON2_MOVED` 0x00000200L /* PDCurses */
- #define `BUTTON3_RELEASED` 0x00000400L
- #define `BUTTON3_PRESSED` 0x00000800L
- #define `BUTTON3_CLICKED` 0x00001000L
- #define `BUTTON3_DOUBLE_CLICKED` 0x00002000L
- #define `BUTTON3_TRIPLE_CLICKED` 0x00004000L
- #define `BUTTON3_MOVED` 0x00004000L /* PDCurses */
- #define `BUTTON4_RELEASED` 0x00008000L
- #define `BUTTON4_PRESSED` 0x00010000L
- #define `BUTTON4_CLICKED` 0x00020000L
- #define `BUTTON4_DOUBLE_CLICKED` 0x00040000L
- #define `BUTTON4_TRIPLE_CLICKED` 0x00080000L
- #define `BUTTON5_RELEASED` 0x00100000L
- #define `BUTTON5_PRESSED` 0x00200000L
- #define `BUTTON5_CLICKED` 0x00400000L
- #define `BUTTON5_DOUBLE_CLICKED` 0x00800000L
- #define `BUTTON5_TRIPLE_CLICKED` 0x01000000L
- #define `MOUSE_WHEEL_SCROLL` 0x02000000L /* PDCurses */
- #define `BUTTON_MODIFIER_SHIFT` 0x04000000L /* PDCurses */
- #define `BUTTON_MODIFIER_CONTROL` 0x08000000L /* PDCurses */
- #define `BUTTON_MODIFIER_ALT` 0x10000000L /* PDCurses */
- #define `ALL_MOUSE_EVENTS` 0x1fffffffL
- #define `REPORT_MOUSE_POSITION` 0x20000000L
- #define `BUTTON_SHIFT` PDC_BUTTON_SHIFT
- #define `BUTTON_CONTROL` PDC_BUTTON_CONTROL
- #define `BUTTON_ALT` PDC_BUTTON_ALT
- #define `PDCEX` extern
- #define `A_NORMAL` (chtype)0
- #define `PDC_CHARTEXT_BITS` 21
- #define `A_CHARTEXT` (chtype)((chtype)0x1 << PDC_CHARTEXT_BITS) - 1)
- #define `A_ALTCHARSET` ((chtype)0x001 << PDC_CHARTEXT_BITS)
- #define `A_RIGHTLINE` ((chtype)0x002 << PDC_CHARTEXT_BITS)
- #define `A_LEFTLINE` ((chtype)0x004 << PDC_CHARTEXT_BITS)
- #define `A_INVIS` ((chtype)0x008 << PDC_CHARTEXT_BITS)
- #define `A_UNDERLINE` ((chtype)0x010 << PDC_CHARTEXT_BITS)
- #define `A_REVERSE` ((chtype)0x020 << PDC_CHARTEXT_BITS)
- #define `A_BLINK` ((chtype)0x040 << PDC_CHARTEXT_BITS)
- #define `A_BOLD` ((chtype)0x080 << PDC_CHARTEXT_BITS)
- #define `A_OVERLINE` ((chtype)0x100 << PDC_CHARTEXT_BITS)
- #define `A_STRIKEOUT` ((chtype)0x200 << PDC_CHARTEXT_BITS)
- #define `A_DIM` ((chtype)0x400 << PDC_CHARTEXT_BITS)
- #define `PDC_COLOR_SHIFT` (PDC_CHARTEXT_BITS + 12)
- #define `A_COLOR` ((chtype)0x7fffffff << PDC_COLOR_SHIFT)
- #define `A_RGB_COLOR` ((chtype)0x40000000 << PDC_COLOR_SHIFT)
- #define `A_ATTRIBUTES` (((chtype)0xfff << PDC_CHARTEXT_BITS) | A_COLOR)
- #define `A_RGB`(rfore, gfore, bfore, rback, gback, bback)
- #define `A_ITALIC` A_INVIS
- #define `A_PROTECT` (A_UNDERLINE | A_LEFTLINE | A_RIGHTLINE)
- #define `A_STANDOUT` (A_REVERSE | A_BOLD) /* X/Open */
- #define `CHR_MSK` A_CHARTEXT /* Obsolete */

- `#define ATR_MSK A_ATTRIBUTES /* Obsolete */`
- `#define ATR_NRM A_NORMAL /* Obsolete */`
- `#define WA_NORMAL A_NORMAL`
- `#define WA_ALTCHARSET A_ALTCHARSET`
- `#define WA_BLINK A_BLINK`
- `#define WA_BOLD A_BOLD`
- `#define WA_DIM A_DIM`
- `#define WA_INVIS A_INVIS`
- `#define WA_LEFT A_LEFTLINE`
- `#define WA_PROTECT A_PROTECT`
- `#define WA_REVERSE A_REVERSE`
- `#define WA_RIGHT A_RIGHTLINE`
- `#define WA_STANDOUT A_STANDOUT`
- `#define WA_UNDERLINE A_UNDERLINE`
- `#define WA_HORIZONTAL A_NORMAL`
- `#define WA_LOW A_NORMAL`
- `#define WA_TOP A_NORMAL`
- `#define WA_VERTICAL A_NORMAL`
- `#define WA_ATTRIBUTES A_ATTRIBUTES`
- `#define ACS_PICK(w, n) ((chtype)w | A_ALTCHARSET)`
- `#define ACS_ULCORNER ACS_PICK('l', '+')`
- `#define ACS_LLCORNER ACS_PICK('m', '+')`
- `#define ACS_URCORNER ACS_PICK('k', '+')`
- `#define ACS_LRCORNER ACS_PICK('j', '+')`
- `#define ACS_RTEE ACS_PICK('u', '+')`
- `#define ACS_LTEE ACS_PICK('t', '+')`
- `#define ACS_BTEE ACS_PICK('v', '+')`
- `#define ACS_TTEE ACS_PICK('w', '+')`
- `#define ACS_HLINE ACS_PICK('q', '-')`
- `#define ACS_VLINE ACS_PICK('x', '|')`
- `#define ACS_PLUS ACS_PICK('n', '+')`
- `#define ACS_S1 ACS_PICK('o', '-')`
- `#define ACS_S9 ACS_PICK('s', '_')`
- `#define ACS_DIAMOND ACS_PICK('"', '+')`
- `#define ACS_CKBOARD ACS_PICK('a', ':')`
- `#define ACS_DEGREE ACS_PICK('f', '^')`
- `#define ACS_PLMINUS ACS_PICK('g', '#')`
- `#define ACS_BULLET ACS_PICK('~', 'o')`
- `#define ACS_LARROW ACS_PICK('<', '<')`
- `#define ACS_RARROW ACS_PICK('>', '>')`
- `#define ACS_DARROW ACS_PICK('.', 'v')`
- `#define ACS_UARROW ACS_PICK('-', '^')`
- `#define ACS_BOARD ACS_PICK('h', '#')`
- `#define ACS_LANTERN ACS_PICK('i', '*')`
- `#define ACS_BLOCK ACS_PICK('O', '#')`
- `#define ACS_S3 ACS_PICK('p', '-')`
- `#define ACS_S7 ACS_PICK('r', '-')`
- `#define ACS_LEQUAL ACS_PICK('y', '<')`
- `#define ACS_GEQUAL ACS_PICK('z', '>')`
- `#define ACS_PI ACS_PICK('{', 'n')`
- `#define ACS_NEQUAL ACS_PICK('|', '+')`
- `#define ACS_STERLING ACS_PICK('}', 'L')`
- `#define ACS_BSSB ACS_ULCORNER`
- `#define ACS_SSBB ACS_LLCORNER`
- `#define ACS_BBSS ACS_URCORNER`

- `#define ACS_SBBS ACS_LRCORNER`
- `#define ACS_SBSS ACS_RTEE`
- `#define ACS_SSSB ACS_LTEE`
- `#define ACS_SSBS ACS_BTEE`
- `#define ACS_BSSS ACS_TTEE`
- `#define ACS_BSBS ACS_HLINE`
- `#define ACS_SBSB ACS_VLINE`
- `#define ACS_SSSS ACS_PLUS`
- `#define COLOR_BLACK 0`
- `#define COLOR_BLUE 1`
- `#define COLOR_GREEN 2`
- `#define COLOR_RED 4`
- `#define COLOR_CYAN (COLOR_BLUE | COLOR_GREEN)`
- `#define COLOR_MAGENTA (COLOR_RED | COLOR_BLUE)`
- `#define COLOR_YELLOW (COLOR_RED | COLOR_GREEN)`
- `#define COLOR_WHITE 7`
- `#define KEY_OFFSET 0x100`
- `#define KEY_CODE_YES (KEY_OFFSET + 0x00) /* If get_wch() gives a key code */`
- `#define KEY_BREAK (KEY_OFFSET + 0x01) /* Not on PC KBD */`
- `#define KEY_DOWN (KEY_OFFSET + 0x02) /* Down arrow key */`
- `#define KEY_UP (KEY_OFFSET + 0x03) /* Up arrow key */`
- `#define KEY_LEFT (KEY_OFFSET + 0x04) /* Left arrow key */`
- `#define KEY_RIGHT (KEY_OFFSET + 0x05) /* Right arrow key */`
- `#define KEY_HOME (KEY_OFFSET + 0x06) /* home key */`
- `#define KEY_BACKSPACE (KEY_OFFSET + 0x07) /* not on pc */`
- `#define KEY_F0 (KEY_OFFSET + 0x08) /* function keys; 64 reserved */`
- `#define KEY_DL (KEY_OFFSET + 0x48) /* delete line */`
- `#define KEY_IL (KEY_OFFSET + 0x49) /* insert line */`
- `#define KEY_DC (KEY_OFFSET + 0x4a) /* delete character */`
- `#define KEY_IC (KEY_OFFSET + 0x4b) /* insert char or enter ins mode */`
- `#define KEY_EIC (KEY_OFFSET + 0x4c) /* exit insert char mode */`
- `#define KEY_CLEAR (KEY_OFFSET + 0x4d) /* clear screen */`
- `#define KEY_EOS (KEY_OFFSET + 0x4e) /* clear to end of screen */`
- `#define KEY_EOL (KEY_OFFSET + 0x4f) /* clear to end of line */`
- `#define KEY_SF (KEY_OFFSET + 0x50) /* scroll 1 line forward */`
- `#define KEY_SR (KEY_OFFSET + 0x51) /* scroll 1 line back (reverse) */`
- `#define KEY_NPAGE (KEY_OFFSET + 0x52) /* next page */`
- `#define KEY_PPAGE (KEY_OFFSET + 0x53) /* previous page */`
- `#define KEY_STAB (KEY_OFFSET + 0x54) /* set tab */`
- `#define KEY_CTAB (KEY_OFFSET + 0x55) /* clear tab */`
- `#define KEY_CATAB (KEY_OFFSET + 0x56) /* clear all tabs */`
- `#define KEY_ENTER (KEY_OFFSET + 0x57) /* enter or send (unreliable) */`
- `#define KEY_SRESET (KEY_OFFSET + 0x58) /* soft/reset (partial/unreliable) */`
- `#define KEY_RESET (KEY_OFFSET + 0x59) /* reset/hard reset (unreliable) */`
- `#define KEY_PRINT (KEY_OFFSET + 0x5a) /* print/copy */`
- `#define KEY_LL (KEY_OFFSET + 0x5b) /* home down/bottom (lower left) */`
- `#define KEY_ABORT (KEY_OFFSET + 0x5c) /* abort/terminate key (any) */`
- `#define KEY_SHELP (KEY_OFFSET + 0x5d) /* short help */`
- `#define KEY_LHELP (KEY_OFFSET + 0x5e) /* long help */`
- `#define KEY_BTAB (KEY_OFFSET + 0x5f) /* Back tab key */`
- `#define KEY_BEG (KEY_OFFSET + 0x60) /* beg(inning) key */`
- `#define KEY_CANCEL (KEY_OFFSET + 0x61) /* cancel key */`
- `#define KEY_CLOSE (KEY_OFFSET + 0x62) /* close key */`
- `#define KEY_COMMAND (KEY_OFFSET + 0x63) /* cmd (command) key */`
- `#define KEY_COPY (KEY_OFFSET + 0x64) /* copy key */`

- #define KEY_CREATE (KEY_OFFSET + 0x65) /* create key */
- #define KEY_END (KEY_OFFSET + 0x66) /* end key */
- #define KEY_EXIT (KEY_OFFSET + 0x67) /* exit key */
- #define KEY_FIND (KEY_OFFSET + 0x68) /* find key */
- #define KEY_HELP (KEY_OFFSET + 0x69) /* help key */
- #define KEY_MARK (KEY_OFFSET + 0x6a) /* mark key */
- #define KEY_MESSAGE (KEY_OFFSET + 0x6b) /* message key */
- #define KEY_MOVE (KEY_OFFSET + 0x6c) /* move key */
- #define KEY_NEXT (KEY_OFFSET + 0x6d) /* next object key */
- #define KEY_OPEN (KEY_OFFSET + 0x6e) /* open key */
- #define KEY_OPTIONS (KEY_OFFSET + 0x6f) /* options key */
- #define KEY_PREVIOUS (KEY_OFFSET + 0x70) /* previous object key */
- #define KEY_REDO (KEY_OFFSET + 0x71) /* redo key */
- #define KEY_REFERENCE (KEY_OFFSET + 0x72) /* ref(erence) key */
- #define KEY_REFRESH (KEY_OFFSET + 0x73) /* refresh key */
- #define KEY_REPLACE (KEY_OFFSET + 0x74) /* replace key */
- #define KEY_RESTART (KEY_OFFSET + 0x75) /* restart key */
- #define KEY_RESUME (KEY_OFFSET + 0x76) /* resume key */
- #define KEY_SAVE (KEY_OFFSET + 0x77) /* save key */
- #define KEY_SBEG (KEY_OFFSET + 0x78) /* shifted beginning key */
- #define KEY_SCANCEL (KEY_OFFSET + 0x79) /* shifted cancel key */
- #define KEY_SCOMMAND (KEY_OFFSET + 0x7a) /* shifted command key */
- #define KEY_SCOPY (KEY_OFFSET + 0x7b) /* shifted copy key */
- #define KEY_SCREATE (KEY_OFFSET + 0x7c) /* shifted create key */
- #define KEY_SDC (KEY_OFFSET + 0x7d) /* shifted delete char key */
- #define KEY_SDL (KEY_OFFSET + 0x7e) /* shifted delete line key */
- #define KEY_SELECT (KEY_OFFSET + 0x7f) /* select key */
- #define KEY_SEND (KEY_OFFSET + 0x80) /* shifted end key */
- #define KEY_SEOL (KEY_OFFSET + 0x81) /* shifted clear line key */
- #define KEY_SEXIT (KEY_OFFSET + 0x82) /* shifted exit key */
- #define KEY_SFIND (KEY_OFFSET + 0x83) /* shifted find key */
- #define KEY_SHOME (KEY_OFFSET + 0x84) /* shifted home key */
- #define KEY_SIC (KEY_OFFSET + 0x85) /* shifted input key */
- #define KEY_SLEFT (KEY_OFFSET + 0x87) /* shifted left arrow key */
- #define KEY_SMESSAGE (KEY_OFFSET + 0x88) /* shifted message key */
- #define KEY_SMOVE (KEY_OFFSET + 0x89) /* shifted move key */
- #define KEY_SNEXT (KEY_OFFSET + 0x8a) /* shifted next key */
- #define KEY_SOPTIONS (KEY_OFFSET + 0x8b) /* shifted options key */
- #define KEY_SPREVIOUS (KEY_OFFSET + 0x8c) /* shifted prev key */
- #define KEY_SPRINT (KEY_OFFSET + 0x8d) /* shifted print key */
- #define KEY_SREDO (KEY_OFFSET + 0x8e) /* shifted redo key */
- #define KEY_SREPLACE (KEY_OFFSET + 0x8f) /* shifted replace key */
- #define KEY_SRIGHT (KEY_OFFSET + 0x90) /* shifted right arrow */
- #define KEY_SRSUME (KEY_OFFSET + 0x91) /* shifted resume key */
- #define KEY_SSAVE (KEY_OFFSET + 0x92) /* shifted save key */
- #define KEY_SSUSPEND (KEY_OFFSET + 0x93) /* shifted suspend key */
- #define KEY_SUNDO (KEY_OFFSET + 0x94) /* shifted undo key */
- #define KEY_SUSPEND (KEY_OFFSET + 0x95) /* suspend key */
- #define KEY_UNDO (KEY_OFFSET + 0x96) /* undo key */
- #define ALT_0 (KEY_OFFSET + 0x97)
- #define ALT_1 (KEY_OFFSET + 0x98)
- #define ALT_2 (KEY_OFFSET + 0x99)
- #define ALT_3 (KEY_OFFSET + 0x9a)
- #define ALT_4 (KEY_OFFSET + 0x9b)
- #define ALT_5 (KEY_OFFSET + 0x9c)

- `#define ALT_6 (KEY_OFFSET + 0x9d)`
- `#define ALT_7 (KEY_OFFSET + 0x9e)`
- `#define ALT_8 (KEY_OFFSET + 0x9f)`
- `#define ALT_9 (KEY_OFFSET + 0xa0)`
- `#define ALT_A (KEY_OFFSET + 0xa1)`
- `#define ALT_B (KEY_OFFSET + 0xa2)`
- `#define ALT_C (KEY_OFFSET + 0xa3)`
- `#define ALT_D (KEY_OFFSET + 0xa4)`
- `#define ALT_E (KEY_OFFSET + 0xa5)`
- `#define ALT_F (KEY_OFFSET + 0xa6)`
- `#define ALT_G (KEY_OFFSET + 0xa7)`
- `#define ALT_H (KEY_OFFSET + 0xa8)`
- `#define ALT_I (KEY_OFFSET + 0xa9)`
- `#define ALT_J (KEY_OFFSET + 0xaa)`
- `#define ALT_K (KEY_OFFSET + 0xab)`
- `#define ALT_L (KEY_OFFSET + 0xac)`
- `#define ALT_M (KEY_OFFSET + 0xad)`
- `#define ALT_N (KEY_OFFSET + 0xae)`
- `#define ALT_O (KEY_OFFSET + 0xaf)`
- `#define ALT_P (KEY_OFFSET + 0xb0)`
- `#define ALT_Q (KEY_OFFSET + 0xb1)`
- `#define ALT_R (KEY_OFFSET + 0xb2)`
- `#define ALT_S (KEY_OFFSET + 0xb3)`
- `#define ALT_T (KEY_OFFSET + 0xb4)`
- `#define ALT_U (KEY_OFFSET + 0xb5)`
- `#define ALT_V (KEY_OFFSET + 0xb6)`
- `#define ALT_W (KEY_OFFSET + 0xb7)`
- `#define ALT_X (KEY_OFFSET + 0xb8)`
- `#define ALT_Y (KEY_OFFSET + 0xb9)`
- `#define ALT_Z (KEY_OFFSET + 0xba)`
- `#define CTL_LEFT (KEY_OFFSET + 0xbb) /* Control-Left-Arrow */`
- `#define CTL_RIGHT (KEY_OFFSET + 0xbc)`
- `#define CTL_PGUP (KEY_OFFSET + 0xbd)`
- `#define CTL_PGDN (KEY_OFFSET + 0xbe)`
- `#define CTL_HOME (KEY_OFFSET + 0xbf)`
- `#define CTL_END (KEY_OFFSET + 0xc0)`
- `#define KEY_A1 (KEY_OFFSET + 0xc1) /* upper left on Virtual keypad */`
- `#define KEY_A2 (KEY_OFFSET + 0xc2) /* upper middle on Virt. keypad */`
- `#define KEY_A3 (KEY_OFFSET + 0xc3) /* upper right on Vir. keypad */`
- `#define KEY_B1 (KEY_OFFSET + 0xc4) /* middle left on Virt. keypad */`
- `#define KEY_B2 (KEY_OFFSET + 0xc5) /* center on Virt. keypad */`
- `#define KEY_B3 (KEY_OFFSET + 0xc6) /* middle right on Vir. keypad */`
- `#define KEY_C1 (KEY_OFFSET + 0xc7) /* lower left on Virt. keypad */`
- `#define KEY_C2 (KEY_OFFSET + 0xc8) /* lower middle on Virt. keypad */`
- `#define KEY_C3 (KEY_OFFSET + 0xc9) /* lower right on Vir. keypad */`
- `#define PADSLASH (KEY_OFFSET + 0xca) /* slash on keypad */`
- `#define PADENTER (KEY_OFFSET + 0xcb) /* enter on keypad */`
- `#define CTL_PADENTER (KEY_OFFSET + 0xcc) /* ctl-enter on keypad */`
- `#define ALT_PADENTER (KEY_OFFSET + 0xcd) /* alt-enter on keypad */`
- `#define PADSTOP (KEY_OFFSET + 0xce) /* stop on keypad */`
- `#define PADSTAR (KEY_OFFSET + 0xcf) /* star on keypad */`
- `#define PADMINUS (KEY_OFFSET + 0xd0) /* minus on keypad */`
- `#define PADPLUS (KEY_OFFSET + 0xd1) /* plus on keypad */`
- `#define CTL_PADSTOP (KEY_OFFSET + 0xd2) /* ctl-stop on keypad */`
- `#define CTL_PADCENTER (KEY_OFFSET + 0xd3) /* ctl-enter on keypad */`

- #define CTL_PADPLUS (KEY_OFFSET + 0xd4) /* ctl-plus on keypad */
- #define CTL_PADMINUS (KEY_OFFSET + 0xd5) /* ctl-minus on keypad */
- #define CTL_PADSLASH (KEY_OFFSET + 0xd6) /* ctl-slash on keypad */
- #define CTL_PADSTAR (KEY_OFFSET + 0xd7) /* ctl-star on keypad */
- #define ALT_PADPLUS (KEY_OFFSET + 0xd8) /* alt-plus on keypad */
- #define ALT_PADMINUS (KEY_OFFSET + 0xd9) /* alt-minus on keypad */
- #define ALT_PADSLASH (KEY_OFFSET + 0xda) /* alt-slash on keypad */
- #define ALT_PADSTAR (KEY_OFFSET + 0xdb) /* alt-star on keypad */
- #define ALT_PADSTOP (KEY_OFFSET + 0xdc) /* alt-stop on keypad */
- #define CTL_INS (KEY_OFFSET + 0xdd) /* ctl-insert */
- #define ALT_DEL (KEY_OFFSET + 0xde) /* alt-delete */
- #define ALT_INS (KEY_OFFSET + 0xdf) /* alt-insert */
- #define CTL_UP (KEY_OFFSET + 0xe0) /* ctl-up arrow */
- #define CTL_DOWN (KEY_OFFSET + 0xe1) /* ctl-down arrow */
- #define CTL_TAB (KEY_OFFSET + 0xe2) /* ctl-tab */
- #define ALT_TAB (KEY_OFFSET + 0xe3)
- #define ALT_MINUS (KEY_OFFSET + 0xe4)
- #define ALT_EQUAL (KEY_OFFSET + 0xe5)
- #define ALT_HOME (KEY_OFFSET + 0xe6)
- #define ALT_PGUP (KEY_OFFSET + 0xe7)
- #define ALT_PGDN (KEY_OFFSET + 0xe8)
- #define ALT_END (KEY_OFFSET + 0xe9)
- #define ALT_UP (KEY_OFFSET + 0xea) /* alt-up arrow */
- #define ALT_DOWN (KEY_OFFSET + 0xeb) /* alt-down arrow */
- #define ALT_RIGHT (KEY_OFFSET + 0xec) /* alt-right arrow */
- #define ALT_LEFT (KEY_OFFSET + 0xed) /* alt-left arrow */
- #define ALT_ENTER (KEY_OFFSET + 0xee) /* alt-enter */
- #define ALT_ESC (KEY_OFFSET + 0xef) /* alt-escape */
- #define ALT_BQUOTE (KEY_OFFSET + 0xf0) /* alt-back quote */
- #define ALT_LBRACKET (KEY_OFFSET + 0xf1) /* alt-left bracket */
- #define ALT_RBRACKET (KEY_OFFSET + 0xf2) /* alt-right bracket */
- #define ALT_SEMICOLON (KEY_OFFSET + 0xf3) /* alt-semi-colon */
- #define ALT_FQUOTE (KEY_OFFSET + 0xf4) /* alt-forward quote */
- #define ALT_COMMA (KEY_OFFSET + 0xf5) /* alt-comma */
- #define ALT_STOP (KEY_OFFSET + 0xf6) /* alt-stop */
- #define ALT_FSLASH (KEY_OFFSET + 0xf7) /* alt-forward slash */
- #define ALT_BKSP (KEY_OFFSET + 0xf8) /* alt-backspace */
- #define CTL_BKSP (KEY_OFFSET + 0xf9) /* ctl-backspace */
- #define PAD0 (KEY_OFFSET + 0xfa) /* keypad 0 */
- #define CTL_PAD0 (KEY_OFFSET + 0xfb) /* ctl-keypad 0 */
- #define CTL_PAD1 (KEY_OFFSET + 0xfc)
- #define CTL_PAD2 (KEY_OFFSET + 0xfd)
- #define CTL_PAD3 (KEY_OFFSET + 0xfe)
- #define CTL_PAD4 (KEY_OFFSET + 0xff)
- #define CTL_PAD5 (KEY_OFFSET + 0x100)
- #define CTL_PAD6 (KEY_OFFSET + 0x101)
- #define CTL_PAD7 (KEY_OFFSET + 0x102)
- #define CTL_PAD8 (KEY_OFFSET + 0x103)
- #define CTL_PAD9 (KEY_OFFSET + 0x104)
- #define ALT_PAD0 (KEY_OFFSET + 0x105) /* alt-keypad 0 */
- #define ALT_PAD1 (KEY_OFFSET + 0x106)
- #define ALT_PAD2 (KEY_OFFSET + 0x107)
- #define ALT_PAD3 (KEY_OFFSET + 0x108)
- #define ALT_PAD4 (KEY_OFFSET + 0x109)
- #define ALT_PAD5 (KEY_OFFSET + 0x10a)

- #define ALT_PAD6 (KEY_OFFSET + 0x10b)
- #define ALT_PAD7 (KEY_OFFSET + 0x10c)
- #define ALT_PAD8 (KEY_OFFSET + 0x10d)
- #define ALT_PAD9 (KEY_OFFSET + 0x10e)
- #define CTL_DEL (KEY_OFFSET + 0x10f) /* clt-delete */
- #define ALT_BSLASH (KEY_OFFSET + 0x110) /* alt-back slash */
- #define CTL_ENTER (KEY_OFFSET + 0x111) /* ctl-enter */
- #define SHF_PADENTER (KEY_OFFSET + 0x112) /* shift-enter on keypad */
- #define SHF_PADSLASH (KEY_OFFSET + 0x113) /* shift-slash on keypad */
- #define SHF_PADSTAR (KEY_OFFSET + 0x114) /* shift-star on keypad */
- #define SHF_PADPLUS (KEY_OFFSET + 0x115) /* shift-plus on keypad */
- #define SHF_PADMINUS (KEY_OFFSET + 0x116) /* shift-minus on keypad */
- #define SHF_UP (KEY_OFFSET + 0x117) /* shift-up on keypad */
- #define SHF_DOWN (KEY_OFFSET + 0x118) /* shift-down on keypad */
- #define SHF_IC (KEY_OFFSET + 0x119) /* shift-insert on keypad */
- #define SHF_DC (KEY_OFFSET + 0x11a) /* shift-delete on keypad */
- #define KEY_MOUSE (KEY_OFFSET + 0x11b) /* "mouse" key */
- #define KEY_SHIFT_L (KEY_OFFSET + 0x11c) /* Left-shift */
- #define KEY_SHIFT_R (KEY_OFFSET + 0x11d) /* Right-shift */
- #define KEY_CONTROL_L (KEY_OFFSET + 0x11e) /* Left-control */
- #define KEY_CONTROL_R (KEY_OFFSET + 0x11f) /* Right-control */
- #define KEY_ALT_L (KEY_OFFSET + 0x120) /* Left-alt */
- #define KEY_ALT_R (KEY_OFFSET + 0x121) /* Right-alt */
- #define KEY_RESIZE (KEY_OFFSET + 0x122) /* Window resize */
- #define KEY_SUP (KEY_OFFSET + 0x123) /* Shifted up arrow */
- #define KEY_SDOWN (KEY_OFFSET + 0x124) /* Shifted down arrow */
- #define CTL_SEMICOLON (KEY_OFFSET + 0x125)
- #define CTL_EQUAL (KEY_OFFSET + 0x126)
- #define CTL_COMMA (KEY_OFFSET + 0x127)
- #define CTL_MINUS (KEY_OFFSET + 0x128)
- #define CTL_STOP (KEY_OFFSET + 0x129)
- #define CTL_FSLASH (KEY_OFFSET + 0x12a)
- #define CTL_BQUOTE (KEY_OFFSET + 0x12b)
- #define KEY_APPS (KEY_OFFSET + 0x12c)
- #define KEY_SAPPS (KEY_OFFSET + 0x12d)
- #define CTL_APPS (KEY_OFFSET + 0x12e)
- #define ALT_APPS (KEY_OFFSET + 0x12f)
- #define KEY_PAUSE (KEY_OFFSET + 0x130)
- #define KEY_SPAUSE (KEY_OFFSET + 0x131)
- #define CTL_PAUSE (KEY_OFFSET + 0x132)
- #define KEY_PRINTSCREEN (KEY_OFFSET + 0x133)
- #define ALT_PRINTSCREEN (KEY_OFFSET + 0x134)
- #define KEY_SCROLLLOCK (KEY_OFFSET + 0x135)
- #define ALT_SCROLLLOCK (KEY_OFFSET + 0x136)
- #define CTL_0 (KEY_OFFSET + 0x137)
- #define CTL_1 (KEY_OFFSET + 0x138)
- #define CTL_2 (KEY_OFFSET + 0x139)
- #define CTL_3 (KEY_OFFSET + 0x13a)
- #define CTL_4 (KEY_OFFSET + 0x13b)
- #define CTL_5 (KEY_OFFSET + 0x13c)
- #define CTL_6 (KEY_OFFSET + 0x13d)
- #define CTL_7 (KEY_OFFSET + 0x13e)
- #define CTL_8 (KEY_OFFSET + 0x13f)
- #define CTL_9 (KEY_OFFSET + 0x140)
- #define KEY_BROWSER_BACK (KEY_OFFSET + 0x141)

- #define KEY_SBROWSER_BACK (KEY_OFFSET + 0x142)
- #define KEY_CBROWSER_BACK (KEY_OFFSET + 0x143)
- #define KEY_ABROWSER_BACK (KEY_OFFSET + 0x144)
- #define KEY_BROWSER_FWD (KEY_OFFSET + 0x145)
- #define KEY_SBROWSER_FWD (KEY_OFFSET + 0x146)
- #define KEY_CBROWSER_FWD (KEY_OFFSET + 0x147)
- #define KEY_ABROWSER_FWD (KEY_OFFSET + 0x148)
- #define KEY_BROWSER_REF (KEY_OFFSET + 0x149)
- #define KEY_SBROWSER_REF (KEY_OFFSET + 0x14A)
- #define KEY_CBROWSER_REF (KEY_OFFSET + 0x14B)
- #define KEY_ABROWSER_REF (KEY_OFFSET + 0x14C)
- #define KEY_BROWSER_STOP (KEY_OFFSET + 0x14D)
- #define KEY_SBROWSER_STOP (KEY_OFFSET + 0x14E)
- #define KEY_CBROWSER_STOP (KEY_OFFSET + 0x14F)
- #define KEY_ABROWSER_STOP (KEY_OFFSET + 0x150)
- #define KEY_SEARCH (KEY_OFFSET + 0x151)
- #define KEY_SSEARCH (KEY_OFFSET + 0x152)
- #define KEY_CSEARCH (KEY_OFFSET + 0x153)
- #define KEY_ASEARCH (KEY_OFFSET + 0x154)
- #define KEY_FAVORITES (KEY_OFFSET + 0x155)
- #define KEY_SFAVORITES (KEY_OFFSET + 0x156)
- #define KEY_CFAVORITES (KEY_OFFSET + 0x157)
- #define KEY_AFAVORITES (KEY_OFFSET + 0x158)
- #define KEY_BROWSER_HOME (KEY_OFFSET + 0x159)
- #define KEY_SBROWSER_HOME (KEY_OFFSET + 0x15A)
- #define KEY_CBROWSER_HOME (KEY_OFFSET + 0x15B)
- #define KEY_ABROWSER_HOME (KEY_OFFSET + 0x15C)
- #define KEY_VOLUME_MUTE (KEY_OFFSET + 0x15D)
- #define KEY_SVOLUME_MUTE (KEY_OFFSET + 0x15E)
- #define KEY_CVOLUME_MUTE (KEY_OFFSET + 0x15F)
- #define KEY_AVOLUME_MUTE (KEY_OFFSET + 0x160)
- #define KEY_VOLUME_DOWN (KEY_OFFSET + 0x161)
- #define KEY_SVOLUME_DOWN (KEY_OFFSET + 0x162)
- #define KEY_CVOLUME_DOWN (KEY_OFFSET + 0x163)
- #define KEY_AVOLUME_DOWN (KEY_OFFSET + 0x164)
- #define KEY_VOLUME_UP (KEY_OFFSET + 0x165)
- #define KEY_SVOLUME_UP (KEY_OFFSET + 0x166)
- #define KEY_CVOLUME_UP (KEY_OFFSET + 0x167)
- #define KEY_AVOLUME_UP (KEY_OFFSET + 0x168)
- #define KEY_NEXT_TRACK (KEY_OFFSET + 0x169)
- #define KEY_SNEXT_TRACK (KEY_OFFSET + 0x16A)
- #define KEY_CNEXT_TRACK (KEY_OFFSET + 0x16B)
- #define KEY_ANEXT_TRACK (KEY_OFFSET + 0x16C)
- #define KEY_PREV_TRACK (KEY_OFFSET + 0x16D)
- #define KEY_SPREV_TRACK (KEY_OFFSET + 0x16E)
- #define KEY_CPREV_TRACK (KEY_OFFSET + 0x16F)
- #define KEY_APREV_TRACK (KEY_OFFSET + 0x170)
- #define KEY_MEDIA_STOP (KEY_OFFSET + 0x171)
- #define KEY_SMEDIA_STOP (KEY_OFFSET + 0x172)
- #define KEY_CMEDIA_STOP (KEY_OFFSET + 0x173)
- #define KEY_AMEDIA_STOP (KEY_OFFSET + 0x174)
- #define KEY_PLAY_PAUSE (KEY_OFFSET + 0x175)
- #define KEY_SPLAY_PAUSE (KEY_OFFSET + 0x176)
- #define KEY_CPLAY_PAUSE (KEY_OFFSET + 0x177)
- #define KEY_APLAY_PAUSE (KEY_OFFSET + 0x178)

- #define `KEY_LAUNCH_MAIL` (`KEY_OFFSET` + 0x179)
- #define `KEY_SLAUNCH_MAIL` (`KEY_OFFSET` + 0x17A)
- #define `KEY_CLAUNCH_MAIL` (`KEY_OFFSET` + 0x17B)
- #define `KEY_ALAUNCH_MAIL` (`KEY_OFFSET` + 0x17C)
- #define `KEY_MEDIA_SELECT` (`KEY_OFFSET` + 0x17D)
- #define `KEY_SMEDIA_SELECT` (`KEY_OFFSET` + 0x17E)
- #define `KEY_CMEDIA_SELECT` (`KEY_OFFSET` + 0x17F)
- #define `KEY_AMEDIA_SELECT` (`KEY_OFFSET` + 0x180)
- #define `KEY_LAUNCH_APP1` (`KEY_OFFSET` + 0x181)
- #define `KEY_SLAUNCH_APP1` (`KEY_OFFSET` + 0x182)
- #define `KEY_CLAUNCH_APP1` (`KEY_OFFSET` + 0x183)
- #define `KEY_ALAUNCH_APP1` (`KEY_OFFSET` + 0x184)
- #define `KEY_LAUNCH_APP2` (`KEY_OFFSET` + 0x185)
- #define `KEY_SLAUNCH_APP2` (`KEY_OFFSET` + 0x186)
- #define `KEY_CLAUNCH_APP2` (`KEY_OFFSET` + 0x187)
- #define `KEY_ALAUNCH_APP2` (`KEY_OFFSET` + 0x188)
- #define `KEY_MIN` `KEY_BREAK` /* Minimum curses key value */
- #define `KEY_MAX` `KEY_ALAUNCH_APP2` /* Maximum curses key */
- #define `KEY_F(n)` (`KEY_F0` + (n))
- #define `getch()` `wgetch(stdscr)`
- #define `ungetch(ch)` `PDC_ungetch(ch)`
- #define `COLOR_PAIR(n)` (((`chtype`)(n) << `PDC_COLOR_SHIFT`) & `A_COLOR`)
- #define `PAIR_NUMBER(n)` (((n) & `A_COLOR`) >> `PDC_COLOR_SHIFT`) & 0xff
- #define `getbegyx(w, y, x)` (y = `getbegy(w)`, x = `getbegx(w)`)
- #define `getmaxyx(w, y, x)` (y = `getmaxy(w)`, x = `getmaxx(w)`)
- #define `getparyx(w, y, x)` (y = `getpary(w)`, x = `getparx(w)`)
- #define `getyx(w, y, x)` (y = `getcury(w)`, x = `getcurx(w)`)
- #define `getsyx(y, x)`
- #define `PDC_CLIP_SUCCESS` 0
- #define `PDC_CLIP_ACCESS_ERROR` 1
- #define `PDC_CLIP_EMPTY` 2
- #define `PDC_CLIP_MEMORY_ERROR` 3
- #define `PDC_KEY_MODIFIER_SHIFT` 1
- #define `PDC_KEY_MODIFIER_CONTROL` 2
- #define `PDC_KEY_MODIFIER_ALT` 4
- #define `PDC_KEY_MODIFIER_NUMLOCK` 8

Typedefs

- typedef unsigned char `bool`
- typedef uint64_t `chtype`
- typedef `chtype` `cchar_t`
- typedef `chtype` `attr_t`
- typedef unsigned long `mmask_t`
- typedef struct `_win` `WINDOW`

Functions

- int [addch](#) (const [chtype](#))
- int [addchnstr](#) (const [chtype](#) *, int)
- int [addchstr](#) (const [chtype](#) *)
- int [addnstr](#) (const char *, int)
- int [addstr](#) (const char *)
- int [attroff](#) ([chtype](#))
- int [attron](#) ([chtype](#))
- int [attrset](#) ([chtype](#))
- int [attr_get](#) ([attr_t](#) *, short *, void *)
- int [attr_off](#) ([attr_t](#), void *)
- int [attr_on](#) ([attr_t](#), void *)
- int [attr_set](#) ([attr_t](#), short, void *)
- int [baudrate](#) (void)
- int [beep](#) (void)
- int [bkgd](#) ([chtype](#))
- void [bkgdset](#) ([chtype](#))
- int [border](#) ([chtype](#), [chtype](#), [chtype](#), [chtype](#), [chtype](#), [chtype](#), [chtype](#), [chtype](#))
- int [box](#) ([WINDOW](#) *, [chtype](#), [chtype](#))
- bool [can_change_color](#) (void)
- int [cbreak](#) (void)
- int [chgat](#) (int, [attr_t](#), short, const void *)
- int [clearok](#) ([WINDOW](#) *, bool)
- int [clear](#) (void)
- int [clrtoebot](#) (void)
- int [clrtoeol](#) (void)
- int [color_content](#) (short, short *, short *, short *)
- int [color_set](#) (short, void *)
- int [copywin](#) (const [WINDOW](#) *, [WINDOW](#) *, int, int, int, int, int, int, int)
- int [curs_set](#) (int)
- int [def_prog_mode](#) (void)
- int [def_shell_mode](#) (void)
- int [delay_output](#) (int)
- int [delch](#) (void)
- int [deleteln](#) (void)
- void [delscreen](#) ([SCREEN](#) *)
- int [delwin](#) ([WINDOW](#) *)
- [WINDOW](#) * [derwin](#) ([WINDOW](#) *, int, int, int, int)
- int [doupdate](#) (void)
- [WINDOW](#) * [dupwin](#) ([WINDOW](#) *)
- int [echochar](#) (const [chtype](#))
- int [echo](#) (void)
- int [endwin](#) (void)
- char [erasechar](#) (void)
- int [erase](#) (void)
- void [filter](#) (void)
- int [flash](#) (void)
- int [flushinp](#) (void)
- [chtype](#) [getbkgd](#) ([WINDOW](#) *)
- int [getnstr](#) (char *, int)
- int [getstr](#) (char *)
- [WINDOW](#) * [getwin](#) ([FILE](#) *)
- int [halfdelay](#) (int)
- bool [has_colors](#) (void)

- [bool has_ic](#) (void)
- [bool has_il](#) (void)
- [int hline](#) ([chtype](#), int)
- [void idcok](#) ([WINDOW](#) *, [bool](#))
- [int idlok](#) ([WINDOW](#) *, [bool](#))
- [void immedok](#) ([WINDOW](#) *, [bool](#))
- [int inchnstr](#) ([chtype](#) *, int)
- [int inchstr](#) ([chtype](#) *)
- [chtype inch](#) (void)
- [int init_color](#) (short, short, short, short)
- [int init_pair](#) (short, short, short)
- [WINDOW](#) * [initscr](#) (void)
- [int innstr](#) (char *, int)
- [int insch](#) ([chtype](#))
- [int insdelln](#) (int)
- [int insertln](#) (void)
- [int insnstr](#) (const char *, int)
- [int insstr](#) (const char *)
- [int instr](#) (char *)
- [int intrflush](#) ([WINDOW](#) *, [bool](#))
- [bool isendwin](#) (void)
- [bool is_linetouched](#) ([WINDOW](#) *, int)
- [bool is_wintouched](#) ([WINDOW](#) *)
- [char](#) * [keyname](#) (int)
- [int keypad](#) ([WINDOW](#) *, [bool](#))
- [char](#) [killchar](#) (void)
- [int leaveok](#) ([WINDOW](#) *, [bool](#))
- [char](#) * [longname](#) (void)
- [int meta](#) ([WINDOW](#) *, [bool](#))
- [int move](#) (int, int)
- [int mvaddch](#) (int, int, const [chtype](#))
- [int mvaddchnstr](#) (int, int, const [chtype](#) *, int)
- [int mvaddchstr](#) (int, int, const [chtype](#) *)
- [int mvaddnstr](#) (int, int, const char *, int)
- [int mvaddstr](#) (int, int, const char *)
- [int mvchgat](#) (int, int, int, [attr_t](#), short, const void *)
- [int mvcur](#) (int, int, int, int)
- [int mvdelch](#) (int, int)
- [int mvderwin](#) ([WINDOW](#) *, int, int)
- [int mvgetch](#) (int, int)
- [int mvgetnstr](#) (int, int, char *, int)
- [int mvgetstr](#) (int, int, char *)
- [int mvhline](#) (int, int, [chtype](#), int)
- [chtype mvinch](#) (int, int)
- [int mvinchnstr](#) (int, int, [chtype](#) *, int)
- [int mvinchstr](#) (int, int, [chtype](#) *)
- [int mvinnstr](#) (int, int, char *, int)
- [int mvinsch](#) (int, int, [chtype](#))
- [int mvinsnstr](#) (int, int, const char *, int)
- [int mvinsstr](#) (int, int, const char *)
- [int mvinstr](#) (int, int, char *)
- [int mvprintw](#) (int, int, const char *,...)
- [int mvscanw](#) (int, int, const char *,...)
- [int mvvline](#) (int, int, [chtype](#), int)
- [int mvwaddchnstr](#) ([WINDOW](#) *, int, int, const [chtype](#) *, int)

- int `mvwaddchstr` (`WINDOW *`, int, int, const `chtype *`)
- int `mvwaddch` (`WINDOW *`, int, int, const `chtype`)
- int `mvwaddnstr` (`WINDOW *`, int, int, const char *, int)
- int `mvwaddstr` (`WINDOW *`, int, int, const char *)
- int `mvwchgat` (`WINDOW *`, int, int, int, `attr_t`, short, const void *)
- int `mvwdelch` (`WINDOW *`, int, int)
- int `mvwgetch` (`WINDOW *`, int, int)
- int `mvwgetnstr` (`WINDOW *`, int, int, char *, int)
- int `mvwgetstr` (`WINDOW *`, int, int, char *)
- int `mvwhline` (`WINDOW *`, int, int, `chtype`, int)
- int `mvwinchnstr` (`WINDOW *`, int, int, `chtype *`, int)
- int `mvwinchstr` (`WINDOW *`, int, int, `chtype *`)
- `chtype` `mvwinch` (`WINDOW *`, int, int)
- int `mvwinnstr` (`WINDOW *`, int, int, char *, int)
- int `mvwinsch` (`WINDOW *`, int, int, `chtype`)
- int `mvwinsnstr` (`WINDOW *`, int, int, const char *, int)
- int `mvwinsstr` (`WINDOW *`, int, int, const char *)
- int `mvwinstr` (`WINDOW *`, int, int, char *)
- int `mvwin` (`WINDOW *`, int, int)
- int `mvwprintw` (`WINDOW *`, int, int, const char *,...)
- int `mvwscanw` (`WINDOW *`, int, int, const char *,...)
- int `mvwvline` (`WINDOW *`, int, int, `chtype`, int)
- int `napms` (int)
- `WINDOW *` `newpad` (int, int)
- `SCREEN *` `newterm` (const char *, FILE *, FILE *)
- `WINDOW *` `newwin` (int, int, int, int)
- int `nl` (void)
- int `nocbreak` (void)
- int `nodelay` (`WINDOW *`, bool)
- int `noecho` (void)
- int `nonl` (void)
- void `noqiflush` (void)
- int `noraw` (void)
- int `notimeout` (`WINDOW *`, bool)
- int `overlay` (const `WINDOW *`, `WINDOW *`)
- int `overwrite` (const `WINDOW *`, `WINDOW *`)
- int `pair_content` (short, short *, short *)
- int `pechochar` (`WINDOW *`, `chtype`)
- int `pnoutrefresh` (`WINDOW *`, int, int, int, int, int, int)
- int `prefresh` (`WINDOW *`, int, int, int, int, int, int)
- int `printw` (const char *,...)
- int `putwin` (`WINDOW *`, FILE *)
- void `qiflush` (void)
- int `raw` (void)
- int `redrawwin` (`WINDOW *`)
- int `refresh` (void)
- int `reset_prog_mode` (void)
- int `reset_shell_mode` (void)
- int `resetty` (void)
- int `ripline` (int, int*)(`WINDOW *`, int)
- int `savetty` (void)
- int `scanw` (const char *,...)
- int `scr_dump` (const char *)
- int `scr_init` (const char *)
- int `scr_restore` (const char *)

- int [scr_set](#) (const char *)
- int [scl](#) (int)
- int [scroll](#) (WINDOW *, int)
- int [scrollok](#) (WINDOW *, bool)
- SCREEN * [set_term](#) (SCREEN *)
- int [setscrreg](#) (int, int)
- int [slk_attr](#) (const chtype)
- int [slk_attr_off](#) (const attr_t, void *)
- int [slk_attron](#) (const chtype)
- int [slk_attr_on](#) (const attr_t, void *)
- int [slk_attrset](#) (const chtype)
- int [slk_attr_set](#) (const attr_t, short, void *)
- int [slk_clear](#) (void)
- int [slk_color](#) (short)
- int [slk_init](#) (int)
- char * [slk_label](#) (int)
- int [slk_noutrefresh](#) (void)
- int [slk_refresh](#) (void)
- int [slk_restore](#) (void)
- int [slk_set](#) (int, const char *, int)
- int [slk_touch](#) (void)
- int [standend](#) (void)
- int [standout](#) (void)
- int [start_color](#) (void)
- WINDOW * [subpad](#) (WINDOW *, int, int, int, int)
- WINDOW * [subwin](#) (WINDOW *, int, int, int, int)
- int [syncok](#) (WINDOW *, bool)
- chtype [termattrs](#) (void)
- attr_t [term_attrs](#) (void)
- char * [termname](#) (void)
- void [timeout](#) (int)
- int [touchline](#) (WINDOW *, int, int)
- int [touchwin](#) (WINDOW *)
- int [typeahead](#) (int)
- int [untouchwin](#) (WINDOW *)
- void [use_env](#) (bool)
- int [vidattr](#) (chtype)
- int [vid_attr](#) (attr_t, short, void *)
- int [vidputs](#) (chtype, int (*)(int))
- int [vid_puts](#) (attr_t, short, void *, int (*)(int))
- int [vline](#) (chtype, int)
- int [vw_printw](#) (WINDOW *, const char *, va_list)
- int [vwprintw](#) (WINDOW *, const char *, va_list)
- int [vw_scanw](#) (WINDOW *, const char *, va_list)
- int [vwscanw](#) (WINDOW *, const char *, va_list)
- int [waddchnstr](#) (WINDOW *, const chtype *, int)
- int [waddchstr](#) (WINDOW *, const chtype *)
- int [waddch](#) (WINDOW *, const chtype)
- int [waddnstr](#) (WINDOW *, const char *, int)
- int [waddstr](#) (WINDOW *, const char *)
- int [wattroff](#) (WINDOW *, chtype)
- int [wattron](#) (WINDOW *, chtype)
- int [wattrset](#) (WINDOW *, chtype)
- int [wattr_get](#) (WINDOW *, attr_t *, short *, void *)
- int [wattr_off](#) (WINDOW *, attr_t, void *)

- int [wattr_on](#) (WINDOW *, attr_t, void *)
- int [wattr_set](#) (WINDOW *, attr_t, short, void *)
- void [wbkgdset](#) (WINDOW *, chtype)
- int [wbkgd](#) (WINDOW *, chtype)
- int [wborder](#) (WINDOW *, chtype, chtype, chtype, chtype, chtype, chtype, chtype, chtype)
- int [wchgat](#) (WINDOW *, int, attr_t, short, const void *)
- int [wclear](#) (WINDOW *)
- int [wclrtobot](#) (WINDOW *)
- int [wclrtoeol](#) (WINDOW *)
- int [wcolor_set](#) (WINDOW *, short, void *)
- void [wcursyncup](#) (WINDOW *)
- int [wdelch](#) (WINDOW *)
- int [wdeleteln](#) (WINDOW *)
- int [wechochar](#) (WINDOW *, const chtype)
- int [werase](#) (WINDOW *)
- int [wgetch](#) (WINDOW *)
- int [wgetnstr](#) (WINDOW *, char *, int)
- int [wgetstr](#) (WINDOW *, char *)
- int [whline](#) (WINDOW *, chtype, int)
- int [winchnstr](#) (WINDOW *, chtype *, int)
- int [winchstr](#) (WINDOW *, chtype *)
- chtype [winch](#) (WINDOW *)
- int [winnstr](#) (WINDOW *, char *, int)
- int [winsch](#) (WINDOW *, chtype)
- int [winsdelln](#) (WINDOW *, int)
- int [winsertln](#) (WINDOW *)
- int [winsnstr](#) (WINDOW *, const char *, int)
- int [winsstr](#) (WINDOW *, const char *)
- int [winstr](#) (WINDOW *, char *)
- int [wmove](#) (WINDOW *, int, int)
- int [wnoutrefresh](#) (WINDOW *)
- int [wprintw](#) (WINDOW *, const char *,...)
- int [wredrawln](#) (WINDOW *, int, int)
- int [wrefresh](#) (WINDOW *)
- int [wscanw](#) (WINDOW *, const char *,...)
- int [wsclrl](#) (WINDOW *, int)
- int [wsetscrreg](#) (WINDOW *, int, int)
- int [wstandend](#) (WINDOW *)
- int [wstandout](#) (WINDOW *)
- void [wsyncdown](#) (WINDOW *)
- void [wsyncup](#) (WINDOW *)
- void [wtimeout](#) (WINDOW *, int)
- int [wtouchln](#) (WINDOW *, int, int, int)
- int [wvline](#) (WINDOW *, chtype, int)
- chtype [getattrs](#) (WINDOW *)
- int [getbegx](#) (WINDOW *)
- int [getbegy](#) (WINDOW *)
- int [getmaxx](#) (WINDOW *)
- int [getmaxy](#) (WINDOW *)
- int [getparx](#) (WINDOW *)
- int [getpary](#) (WINDOW *)
- int [getcurx](#) (WINDOW *)
- int [getcury](#) (WINDOW *)
- void [traceoff](#) (void)
- void [traceon](#) (void)

- char * [unctrl](#) (chtype)
- int [crmode](#) (void)
- int [nocrmode](#) (void)
- int [draino](#) (int)
- int [resetterm](#) (void)
- int [fixterm](#) (void)
- int [saveterm](#) (void)
- int [setsyx](#) (int, int)
- int [mouse_set](#) (unsigned long)
- int [mouse_on](#) (unsigned long)
- int [mouse_off](#) (unsigned long)
- int [request_mouse_pos](#) (void)
- int [map_button](#) (unsigned long)
- void [wmouse_position](#) (WINDOW *, int *, int *)
- unsigned long [getmouse](#) (void)
- unsigned long [getbmap](#) (void)
- int [assume_default_colors](#) (int, int)
- const char * [curses_version](#) (void)
- bool [has_key](#) (int)
- int [use_default_colors](#) (void)
- int [wresize](#) (WINDOW *, int, int)
- int [mouseinterval](#) (int)
- [mmask_t](#) [mousemask](#) ([mmask_t](#), [mmask_t](#) *)
- bool [mouse_trafo](#) (int *, int *, bool)
- int [nc_getmouse](#) (MEVENT *)
- int [ungetmouse](#) (MEVENT *)
- bool [wenclose](#) (const WINDOW *, int, int)
- bool [wmouse_trafo](#) (const WINDOW *, int *, int *, bool)
- int [addrawch](#) (chtype)
- int [insrawch](#) (chtype)
- bool [is_termresized](#) (void)
- int [mvaddrawch](#) (int, int, chtype)
- int [mvdeleteln](#) (int, int)
- int [mvinsertln](#) (int, int)
- int [mvinsrawch](#) (int, int, chtype)
- int [mvwaddrawch](#) (WINDOW *, int, int, chtype)
- int [mvwdeleteln](#) (WINDOW *, int, int)
- int [mvwininsertln](#) (WINDOW *, int, int)
- int [mvwinsrawch](#) (WINDOW *, int, int, chtype)
- int [raw_output](#) (bool)
- int [resize_term](#) (int, int)
- WINDOW * [resize_window](#) (WINDOW *, int, int)
- int [waddrawch](#) (WINDOW *, chtype)
- int [winsrawch](#) (WINDOW *, chtype)
- char [wordchar](#) (void)
- void [PDC_debug](#) (const char *,...)
- int [PDC_ungetch](#) (int)
- int [PDC_set_blink](#) (bool)
- int [PDC_set_line_color](#) (short)
- void [PDC_set_title](#) (const char *)
- int [PDC_clearclipboard](#) (void)
- int [PDC_freeclipboard](#) (char *)
- int [PDC_getclipboard](#) (char **, long *)
- int [PDC_setclipboard](#) (const char *, long)
- unsigned long [PDC_get_input_fd](#) (void)

- unsigned long [PDC_get_key_modifiers](#) (void)
- int [PDC_return_key_modifiers](#) (bool)
- int [PDC_save_key_modifiers](#) (bool)
- void [PDC_set_resize_limits](#) (const int new_min_lines, const int new_max_lines, const int new_min_cols, const int new_max_cols)
- [WINDOW](#) * [Xinitscr](#) (int, char **)

Variables

- [PDCEX](#) int [LINES](#)
- [PDCEX](#) int [COLS](#)
- [PDCEX](#) [WINDOW](#) * [stdscr](#)
- [PDCEX](#) [WINDOW](#) * [curscr](#)
- [PDCEX](#) [SCREEN](#) * [SP](#)
- [PDCEX](#) [MOUSE_STATUS](#) [Mouse_status](#)
- [PDCEX](#) int [COLORS](#)
- [PDCEX](#) int [COLOR_PAIRS](#)
- [PDCEX](#) int [TABSIZ](#)
- [PDCEX](#) chtype [acs_map](#) []
- [PDCEX](#) char [ttytype](#) []

5.2.1 Macro Definition Documentation

5.2.1.1 `#define A_ALTCHARSET ((chtype)0x001 << PDC_CHARTEXT_BITS)`

Definition at line 452 of file ncurses.h.

5.2.1.2 `#define A_ATTRIBUTES (((chtype)0xfff << PDC_CHARTEXT_BITS) | A_COLOR)`

Definition at line 471 of file ncurses.h.

5.2.1.3 `#define A_BLINK ((chtype)0x040 << PDC_CHARTEXT_BITS)`

Definition at line 458 of file ncurses.h.

5.2.1.4 `#define A_BOLD ((chtype)0x080 << PDC_CHARTEXT_BITS)`

Definition at line 459 of file ncurses.h.

5.2.1.5 `#define A_BUTTON_CHANGED (Mouse_status.changes & 7)`

Definition at line 182 of file ncurses.h.

5.2.1.6 `#define A_CHARTEXT (chtype)((chtype)0x1 << PDC_CHARTEXT_BITS) - 1)`

Definition at line 451 of file ncurses.h.

5.2.1.7 `#define A_COLOR ((chtype)0x7fffffff << PDC_COLOR_SHIFT)`

Definition at line 469 of file ncurses.h.

5.2.1.8 #define A_DIM ((chtype)0x400 << PDC_CHARTEXT_BITS)

Definition at line 462 of file ncurses.h.

5.2.1.9 #define A_INVIS ((chtype)0x008 << PDC_CHARTEXT_BITS)

Definition at line 455 of file ncurses.h.

5.2.1.10 #define A_ITALIC A_INVIS

Definition at line 499 of file ncurses.h.

5.2.1.11 #define A_LEFTLINE ((chtype)0x004 << PDC_CHARTEXT_BITS)

Definition at line 454 of file ncurses.h.

5.2.1.12 #define A_NORMAL (chtype)0

Definition at line 445 of file ncurses.h.

5.2.1.13 #define A_OVERLINE ((chtype)0x100 << PDC_CHARTEXT_BITS)

Definition at line 460 of file ncurses.h.

5.2.1.14 #define A_PROTECT (A_UNDERLINE | A_LEFTLINE | A_RIGHTLINE)

Definition at line 500 of file ncurses.h.

5.2.1.15 #define A_REVERSE ((chtype)0x020 << PDC_CHARTEXT_BITS)

Definition at line 457 of file ncurses.h.

5.2.1.16 #define A_RGB(rfore, gfore, bfore, rback, gback, bback)

Value:

```
(( (((chtype) (bfore) << 25) \
    | ((chtype) (gfore) << 20) \
    | ((chtype) (rfore) << 15) \
    | ((chtype) (bback) << 10) \
    | ((chtype) (gback) << 5) \
    | ((chtype) (rback)      )) << PDC_COLOR_SHIFT) |
A_RGB_COLOR)
```

Definition at line 472 of file ncurses.h.

5.2.1.17 #define A_RGB_COLOR ((chtype)0x40000000 << PDC_COLOR_SHIFT)

Definition at line 470 of file ncurses.h.

5.2.1.18 #define A_RIGHTLINE ((chtype)0x002 << PDC_CHARTEXT_BITS)

Definition at line 453 of file ncurses.h.

5.2.1.19 #define A_STANDOUT (A_REVERSE | A_BOLD) /* X/Open */

Definition at line 525 of file ncurses.h.

5.2.1.20 #define A_STRIKEOUT ((chtype)0x200 << PDC_CHARTEXT_BITS)

Definition at line 461 of file ncurses.h.

5.2.1.21 #define A_UNDERLINE ((chtype)0x010 << PDC_CHARTEXT_BITS)

Definition at line 456 of file ncurses.h.

5.2.1.22 #define ACS_BBSS ACS_URCORNER

Definition at line 618 of file ncurses.h.

5.2.1.23 #define ACS_BLOCK ACS_PICK('0', '#')

Definition at line 601 of file ncurses.h.

5.2.1.24 #define ACS_BOARD ACS_PICK('h', '#')

Definition at line 599 of file ncurses.h.

5.2.1.25 #define ACS_BSBS ACS_HLINE

Definition at line 624 of file ncurses.h.

5.2.1.26 #define ACS_BSSB ACS_ULCORNER

Definition at line 616 of file ncurses.h.

5.2.1.27 #define ACS_BSSS ACS_TTEE

Definition at line 623 of file ncurses.h.

5.2.1.28 #define ACS_BTEE ACS_PICK('v', '+')

Definition at line 575 of file ncurses.h.

5.2.1.29 #define ACS_BULLET ACS_PICK('~', 'o')

Definition at line 589 of file ncurses.h.

5.2.1.30 #define ACS_CKBOARD ACS_PICK('a', ':')

Definition at line 586 of file ncurses.h.

5.2.1.31 `#define ACS_DARROW ACS_PICK('.', 'v')`

Definition at line 597 of file ncurses.h.

5.2.1.32 `#define ACS_DEGREE ACS_PICK('f', '\u00b0')`

Definition at line 587 of file ncurses.h.

5.2.1.33 `#define ACS_DIAMOND ACS_PICK('\u2666', '+')`

Definition at line 585 of file ncurses.h.

5.2.1.34 `#define ACS_GEQUAL ACS_PICK('z', '>')`

Definition at line 609 of file ncurses.h.

5.2.1.35 `#define ACS_HLINE ACS_PICK('q', '-')`

Definition at line 577 of file ncurses.h.

5.2.1.36 `#define ACS_LANTERN ACS_PICK('i', '*')`

Definition at line 600 of file ncurses.h.

5.2.1.37 `#define ACS_LARROW ACS_PICK('l', '<')`

Definition at line 595 of file ncurses.h.

5.2.1.38 `#define ACS_LEQUAL ACS_PICK('y', '<')`

Definition at line 608 of file ncurses.h.

5.2.1.39 `#define ACS_LLCORNER ACS_PICK('m', '+')`

Definition at line 570 of file ncurses.h.

5.2.1.40 `#define ACS_LRCORNER ACS_PICK('j', '+')`

Definition at line 572 of file ncurses.h.

5.2.1.41 `#define ACS_LTEE ACS_PICK('t', '+')`

Definition at line 574 of file ncurses.h.

5.2.1.42 `#define ACS_NEQUAL ACS_PICK('|', '+')`

Definition at line 611 of file ncurses.h.

5.2.1.43 `#define ACS_PI ACS_PICK('{', 'n')`

Definition at line 610 of file ncurses.h.

5.2.1.44 `#define ACS_PICK(w, n)((chtype)w | A_ALTCHARSET)`

Definition at line 562 of file ncurses.h.

5.2.1.45 `#define ACS_PLMINUS ACS_PICK('g', '#')`

Definition at line 588 of file ncurses.h.

5.2.1.46 `#define ACS_PLUS ACS_PICK('n', '+')`

Definition at line 579 of file ncurses.h.

5.2.1.47 `#define ACS_RARROW ACS_PICK('+', '>')`

Definition at line 596 of file ncurses.h.

5.2.1.48 `#define ACS_RTEE ACS_PICK('u', '+')`

Definition at line 573 of file ncurses.h.

5.2.1.49 `#define ACS_S1 ACS_PICK('o', '-')`

Definition at line 583 of file ncurses.h.

5.2.1.50 `#define ACS_S3 ACS_PICK('p', '-')`

Definition at line 606 of file ncurses.h.

5.2.1.51 `#define ACS_S7 ACS_PICK('r', '-')`

Definition at line 607 of file ncurses.h.

5.2.1.52 `#define ACS_S9 ACS_PICK('s', '-')`

Definition at line 584 of file ncurses.h.

5.2.1.53 `#define ACS_SBBS ACS_LRCORNER`

Definition at line 619 of file ncurses.h.

5.2.1.54 `#define ACS_SBSB ACS_VLINE`

Definition at line 625 of file ncurses.h.

5.2.1.55 #define ACS_SBSS ACS_RTEE

Definition at line 620 of file ncurses.h.

5.2.1.56 #define ACS_SSBB ACS_LLCORNER

Definition at line 617 of file ncurses.h.

5.2.1.57 #define ACS_SSBS ACS_BTEE

Definition at line 622 of file ncurses.h.

5.2.1.58 #define ACS_SSSB ACS_LTEE

Definition at line 621 of file ncurses.h.

5.2.1.59 #define ACS_SSSS ACS_PLUS

Definition at line 626 of file ncurses.h.

5.2.1.60 #define ACS_STERLING ACS_PICK('}', 'L')

Definition at line 612 of file ncurses.h.

5.2.1.61 #define ACS_TTEE ACS_PICK('w', '+')

Definition at line 576 of file ncurses.h.

5.2.1.62 #define ACS_UARROW ACS_PICK('-', '^')

Definition at line 598 of file ncurses.h.

5.2.1.63 #define ACS_ULCORNER ACS_PICK('l', '+')

Definition at line 569 of file ncurses.h.

5.2.1.64 #define ACS_URCORNER ACS_PICK('k', '+')

Definition at line 571 of file ncurses.h.

5.2.1.65 #define ACS_VLINE ACS_PICK('x', '|')

Definition at line 578 of file ncurses.h.

5.2.1.66 #define ALL_MOUSE_EVENTS 0xffffffffL

Definition at line 237 of file ncurses.h.

5.2.1.67 `#define ALT_0 (KEY_OFFSET + 0x97)`

Definition at line 806 of file ncurses.h.

5.2.1.68 `#define ALT_1 (KEY_OFFSET + 0x98)`

Definition at line 807 of file ncurses.h.

5.2.1.69 `#define ALT_2 (KEY_OFFSET + 0x99)`

Definition at line 808 of file ncurses.h.

5.2.1.70 `#define ALT_3 (KEY_OFFSET + 0x9a)`

Definition at line 809 of file ncurses.h.

5.2.1.71 `#define ALT_4 (KEY_OFFSET + 0x9b)`

Definition at line 810 of file ncurses.h.

5.2.1.72 `#define ALT_5 (KEY_OFFSET + 0x9c)`

Definition at line 811 of file ncurses.h.

5.2.1.73 `#define ALT_6 (KEY_OFFSET + 0x9d)`

Definition at line 812 of file ncurses.h.

5.2.1.74 `#define ALT_7 (KEY_OFFSET + 0x9e)`

Definition at line 813 of file ncurses.h.

5.2.1.75 `#define ALT_8 (KEY_OFFSET + 0x9f)`

Definition at line 814 of file ncurses.h.

5.2.1.76 `#define ALT_9 (KEY_OFFSET + 0xa0)`

Definition at line 815 of file ncurses.h.

5.2.1.77 `#define ALT_A (KEY_OFFSET + 0xa1)`

Definition at line 816 of file ncurses.h.

5.2.1.78 `#define ALT_APPS (KEY_OFFSET + 0x12f)`

Definition at line 971 of file ncurses.h.

5.2.1.79 `#define ALT_B (KEY_OFFSET + 0xa2)`

Definition at line 817 of file ncurses.h.

5.2.1.80 `#define ALT_BKSP (KEY_OFFSET + 0xf8) /* alt-backspace */`

Definition at line 906 of file ncurses.h.

5.2.1.81 `#define ALT_BQUOTE (KEY_OFFSET + 0xf0) /* alt-back quote */`

Definition at line 898 of file ncurses.h.

5.2.1.82 `#define ALT_BSLASH (KEY_OFFSET + 0x110) /* alt-back slash */`

Definition at line 933 of file ncurses.h.

5.2.1.83 `#define ALT_C (KEY_OFFSET + 0xa3)`

Definition at line 818 of file ncurses.h.

5.2.1.84 `#define ALT_COMMA (KEY_OFFSET + 0xf5) /* alt-comma */`

Definition at line 903 of file ncurses.h.

5.2.1.85 `#define ALT_D (KEY_OFFSET + 0xa4)`

Definition at line 819 of file ncurses.h.

5.2.1.86 `#define ALT_DEL (KEY_OFFSET + 0xde) /* alt-delete */`

Definition at line 880 of file ncurses.h.

5.2.1.87 `#define ALT_DOWN (KEY_OFFSET + 0xeb) /* alt-down arrow */`

Definition at line 893 of file ncurses.h.

5.2.1.88 `#define ALT_E (KEY_OFFSET + 0xa5)`

Definition at line 820 of file ncurses.h.

5.2.1.89 `#define ALT_END (KEY_OFFSET + 0xe9)`

Definition at line 891 of file ncurses.h.

5.2.1.90 `#define ALT_ENTER (KEY_OFFSET + 0xee) /* alt-enter */`

Definition at line 896 of file ncurses.h.

5.2.1.91 `#define ALT_EQUAL (KEY_OFFSET + 0xe5)`

Definition at line 887 of file ncurses.h.

5.2.1.92 `#define ALT_ESC (KEY_OFFSET + 0xef) /* alt-escape */`

Definition at line 897 of file ncurses.h.

5.2.1.93 `#define ALT_F (KEY_OFFSET + 0xa6)`

Definition at line 821 of file ncurses.h.

5.2.1.94 `#define ALT_FQUOTE (KEY_OFFSET + 0xf4) /* alt-forward quote */`

Definition at line 902 of file ncurses.h.

5.2.1.95 `#define ALT_FSLASH (KEY_OFFSET + 0xf7) /* alt-forward slash */`

Definition at line 905 of file ncurses.h.

5.2.1.96 `#define ALT_G (KEY_OFFSET + 0xa7)`

Definition at line 822 of file ncurses.h.

5.2.1.97 `#define ALT_H (KEY_OFFSET + 0xa8)`

Definition at line 823 of file ncurses.h.

5.2.1.98 `#define ALT_HOME (KEY_OFFSET + 0xe6)`

Definition at line 888 of file ncurses.h.

5.2.1.99 `#define ALT_I (KEY_OFFSET + 0xa9)`

Definition at line 824 of file ncurses.h.

5.2.1.100 `#define ALT_INS (KEY_OFFSET + 0xdf) /* alt-insert */`

Definition at line 881 of file ncurses.h.

5.2.1.101 `#define ALT_J (KEY_OFFSET + 0xaa)`

Definition at line 825 of file ncurses.h.

5.2.1.102 `#define ALT_K (KEY_OFFSET + 0xab)`

Definition at line 826 of file ncurses.h.

5.2.1.103 `#define ALT_L (KEY_OFFSET + 0xac)`

Definition at line 827 of file ncurses.h.

5.2.1.104 `#define ALT_LBRACKET (KEY_OFFSET + 0xf1) /* alt-left bracket */`

Definition at line 899 of file ncurses.h.

5.2.1.105 `#define ALT_LEFT (KEY_OFFSET + 0xed) /* alt-left arrow */`

Definition at line 895 of file ncurses.h.

5.2.1.106 `#define ALT_M (KEY_OFFSET + 0xad)`

Definition at line 828 of file ncurses.h.

5.2.1.107 `#define ALT_MINUS (KEY_OFFSET + 0xe4)`

Definition at line 886 of file ncurses.h.

5.2.1.108 `#define ALT_N (KEY_OFFSET + 0xae)`

Definition at line 829 of file ncurses.h.

5.2.1.109 `#define ALT_O (KEY_OFFSET + 0xaf)`

Definition at line 830 of file ncurses.h.

5.2.1.110 `#define ALT_P (KEY_OFFSET + 0xb0)`

Definition at line 831 of file ncurses.h.

5.2.1.111 `#define ALT_PAD0 (KEY_OFFSET + 0x105) /* alt-keypad 0 */`

Definition at line 921 of file ncurses.h.

5.2.1.112 `#define ALT_PAD1 (KEY_OFFSET + 0x106)`

Definition at line 922 of file ncurses.h.

5.2.1.113 `#define ALT_PAD2 (KEY_OFFSET + 0x107)`

Definition at line 923 of file ncurses.h.

5.2.1.114 `#define ALT_PAD3 (KEY_OFFSET + 0x108)`

Definition at line 924 of file ncurses.h.

5.2.1.115 `#define ALT_PAD4 (KEY_OFFSET + 0x109)`

Definition at line 925 of file ncurses.h.

5.2.1.116 `#define ALT_PAD5 (KEY_OFFSET + 0x10a)`

Definition at line 926 of file ncurses.h.

5.2.1.117 `#define ALT_PAD6 (KEY_OFFSET + 0x10b)`

Definition at line 927 of file ncurses.h.

5.2.1.118 `#define ALT_PAD7 (KEY_OFFSET + 0x10c)`

Definition at line 928 of file ncurses.h.

5.2.1.119 `#define ALT_PAD8 (KEY_OFFSET + 0x10d)`

Definition at line 929 of file ncurses.h.

5.2.1.120 `#define ALT_PAD9 (KEY_OFFSET + 0x10e)`

Definition at line 930 of file ncurses.h.

5.2.1.121 `#define ALT_PADENTER (KEY_OFFSET + 0xcd) /* alt-enter on keypad */`

Definition at line 863 of file ncurses.h.

5.2.1.122 `#define ALT_PADMINUS (KEY_OFFSET + 0xd9) /* alt-minus on keypad */`

Definition at line 875 of file ncurses.h.

5.2.1.123 `#define ALT_PADPLUS (KEY_OFFSET + 0xd8) /* alt-plus on keypad */`

Definition at line 874 of file ncurses.h.

5.2.1.124 `#define ALT_PADSLASH (KEY_OFFSET + 0xda) /* alt-slash on keypad */`

Definition at line 876 of file ncurses.h.

5.2.1.125 `#define ALT_PADSTAR (KEY_OFFSET + 0xdb) /* alt-star on keypad */`

Definition at line 877 of file ncurses.h.

5.2.1.126 `#define ALT_PADSTOP (KEY_OFFSET + 0xdc) /* alt-stop on keypad */`

Definition at line 878 of file ncurses.h.

5.2.1.127 `#define ALT_PGDN (KEY_OFFSET + 0xe8)`

Definition at line 890 of file ncurses.h.

5.2.1.128 `#define ALT_PGUP (KEY_OFFSET + 0xe7)`

Definition at line 889 of file ncurses.h.

5.2.1.129 `#define ALT_PRINTSCREEN (KEY_OFFSET + 0x134)`

Definition at line 978 of file ncurses.h.

5.2.1.130 `#define ALT_Q (KEY_OFFSET + 0xb1)`

Definition at line 832 of file ncurses.h.

5.2.1.131 `#define ALT_R (KEY_OFFSET + 0xb2)`

Definition at line 833 of file ncurses.h.

5.2.1.132 `#define ALT_RBRACKET (KEY_OFFSET + 0xf2) /* alt-right bracket */`

Definition at line 900 of file ncurses.h.

5.2.1.133 `#define ALT_RIGHT (KEY_OFFSET + 0xec) /* alt-right arrow */`

Definition at line 894 of file ncurses.h.

5.2.1.134 `#define ALT_S (KEY_OFFSET + 0xb3)`

Definition at line 834 of file ncurses.h.

5.2.1.135 `#define ALT_SCROLLLOCK (KEY_OFFSET + 0x136)`

Definition at line 980 of file ncurses.h.

5.2.1.136 `#define ALT_SEMICOLON (KEY_OFFSET + 0xf3) /* alt-semi-colon */`

Definition at line 901 of file ncurses.h.

5.2.1.137 `#define ALT_STOP (KEY_OFFSET + 0xf6) /* alt-stop */`

Definition at line 904 of file ncurses.h.

5.2.1.138 `#define ALT_T (KEY_OFFSET + 0xb4)`

Definition at line 835 of file ncurses.h.

5.2.1.139 `#define ALT_TAB (KEY_OFFSET + 0xe3)`

Definition at line 885 of file ncurses.h.

5.2.1.140 `#define ALT_U (KEY_OFFSET + 0xb5)`

Definition at line 836 of file ncurses.h.

5.2.1.141 `#define ALT_UP (KEY_OFFSET + 0xea) /* alt-up arrow */`

Definition at line 892 of file ncurses.h.

5.2.1.142 `#define ALT_V (KEY_OFFSET + 0xb6)`

Definition at line 837 of file ncurses.h.

5.2.1.143 `#define ALT_W (KEY_OFFSET + 0xb7)`

Definition at line 838 of file ncurses.h.

5.2.1.144 `#define ALT_X (KEY_OFFSET + 0xb8)`

Definition at line 839 of file ncurses.h.

5.2.1.145 `#define ALT_Y (KEY_OFFSET + 0xb9)`

Definition at line 840 of file ncurses.h.

5.2.1.146 `#define ALT_Z (KEY_OFFSET + 0xba)`

Definition at line 841 of file ncurses.h.

5.2.1.147 `#define ATR_MSK A_ATTRIBUTES /* Obsolete */`

Definition at line 528 of file ncurses.h.

5.2.1.148 `#define ATR_NRM A_NORMAL /* Obsolete */`

Definition at line 529 of file ncurses.h.

5.2.1.149 `#define BSDcurses 1 /* BSD Curses routines */`

Definition at line 38 of file ncurses.h.

5.2.1.150 `#define BUTTON1_CLICKED 0x00000004L`

Definition at line 196 of file ncurses.h.

5.2.1.151 `#define BUTTON1_DOUBLE_CLICKED 0x00000008L`

Definition at line 197 of file ncurses.h.

5.2.1.152 `#define BUTTON1_MOVED 0x00000010L /* PDCurses */`

Definition at line 199 of file ncurses.h.

5.2.1.153 `#define BUTTON1_PRESSED 0x00000002L`

Definition at line 195 of file ncurses.h.

5.2.1.154 `#define BUTTON1_RELEASED 0x00000001L`

Definition at line 194 of file ncurses.h.

5.2.1.155 `#define BUTTON1_TRIPLE_CLICKED 0x00000010L`

Definition at line 198 of file ncurses.h.

5.2.1.156 `#define BUTTON2_CLICKED 0x00000080L`

Definition at line 203 of file ncurses.h.

5.2.1.157 `#define BUTTON2_DOUBLE_CLICKED 0x00000100L`

Definition at line 204 of file ncurses.h.

5.2.1.158 `#define BUTTON2_MOVED 0x00000200L /* PDCurses */`

Definition at line 206 of file ncurses.h.

5.2.1.159 `#define BUTTON2_PRESSED 0x00000040L`

Definition at line 202 of file ncurses.h.

5.2.1.160 `#define BUTTON2_RELEASED 0x00000020L`

Definition at line 201 of file ncurses.h.

5.2.1.161 `#define BUTTON2_TRIPLE_CLICKED 0x00000200L`

Definition at line 205 of file ncurses.h.

5.2.1.162 `#define BUTTON3_CLICKED 0x00001000L`

Definition at line 210 of file ncurses.h.

5.2.1.163 `#define BUTTON3_DOUBLE_CLICKED 0x00002000L`

Definition at line 211 of file ncurses.h.

5.2.1.164 `#define BUTTON3_MOVED 0x00004000L /* PDCurses */`

Definition at line 213 of file ncurses.h.

5.2.1.165 `#define BUTTON3_PRESSED 0x00000800L`

Definition at line 209 of file ncurses.h.

5.2.1.166 `#define BUTTON3_RELEASED 0x00000400L`

Definition at line 208 of file ncurses.h.

5.2.1.167 `#define BUTTON3_TRIPLE_CLICKED 0x00004000L`

Definition at line 212 of file ncurses.h.

5.2.1.168 `#define BUTTON4_CLICKED 0x00020000L`

Definition at line 222 of file ncurses.h.

5.2.1.169 `#define BUTTON4_DOUBLE_CLICKED 0x00040000L`

Definition at line 223 of file ncurses.h.

5.2.1.170 `#define BUTTON4_PRESSED 0x00010000L`

Definition at line 221 of file ncurses.h.

5.2.1.171 `#define BUTTON4_RELEASED 0x00008000L`

Definition at line 220 of file ncurses.h.

5.2.1.172 `#define BUTTON4_TRIPLE_CLICKED 0x00080000L`

Definition at line 224 of file ncurses.h.

5.2.1.173 `#define BUTTON5_CLICKED 0x00400000L`

Definition at line 228 of file ncurses.h.

5.2.1.174 `#define BUTTON5_DOUBLE_CLICKED 0x00800000L`

Definition at line 229 of file ncurses.h.

5.2.1.175 `#define BUTTON5_PRESSED 0x00200000L`

Definition at line 227 of file ncurses.h.

5.2.1.176 `#define BUTTON5_RELEASED 0x00100000L`

Definition at line 226 of file ncurses.h.

5.2.1.177 `#define BUTTON5_TRIPLE_CLICKED 0x01000000L`

Definition at line 230 of file ncurses.h.

5.2.1.178 `#define BUTTON_ACTION_MASK 0x0007 /* PDCurses */`

Definition at line 144 of file ncurses.h.

5.2.1.179 `#define BUTTON_ALT PDC_BUTTON_ALT`

Definition at line 260 of file ncurses.h.

5.2.1.180 `#define BUTTON_CHANGED(x) (Mouse_status.changes & (1 << ((x) - ((x)<4 ? 1 : -5))))`

Definition at line 185 of file ncurses.h.

5.2.1.181 `#define BUTTON_CLICKED 0x0002`

Definition at line 139 of file ncurses.h.

5.2.1.182 `#define BUTTON_CONTROL PDC_BUTTON_CONTROL`

Definition at line 259 of file ncurses.h.

5.2.1.183 `#define BUTTON_DOUBLE_CLICKED 0x0003`

Definition at line 140 of file ncurses.h.

5.2.1.184 `#define BUTTON_MODIFIER_ALT 0x10000000L /* PDCurses */`

Definition at line 235 of file ncurses.h.

5.2.1.185 `#define BUTTON_MODIFIER_CONTROL 0x08000000L /* PDCurses */`

Definition at line 234 of file ncurses.h.

5.2.1.186 `#define BUTTON_MODIFIER_MASK 0x0038 /* PDCurses */`

Definition at line 149 of file ncurses.h.

5.2.1.187 `#define BUTTON_MODIFIER_SHIFT 0x04000000L /* PDCurses */`

Definition at line 233 of file ncurses.h.

5.2.1.188 `#define BUTTON_MOVED 0x0005 /* PDCurses */`

Definition at line 142 of file ncurses.h.

5.2.1.189 `#define BUTTON_PRESSED 0x0001`

Definition at line 138 of file ncurses.h.

5.2.1.190 `#define BUTTON_RELEASED 0x0000`

Definition at line 137 of file ncurses.h.

5.2.1.191 `#define BUTTON_SHIFT PDC_BUTTON_SHIFT`

Definition at line 258 of file ncurses.h.

5.2.1.192 `#define BUTTON_STATUS(x) (Mouse_status.button[(x) - 1])`

Definition at line 186 of file ncurses.h.

5.2.1.193 `#define BUTTON_TRIPLE_CLICKED 0x0004`

Definition at line 141 of file ncurses.h.

5.2.1.194 `#define CHR_MSK A_CHARTEXT /* Obsolete */`

Definition at line 527 of file ncurses.h.

5.2.1.195 `#define CHTYPE_LONG 2 /* chtypes will be (default) 64 bits */`

Definition at line 44 of file ncurses.h.

5.2.1.196 `#define COLOR_BLACK 0`

Definition at line 682 of file ncurses.h.

5.2.1.197 `#define COLOR_BLUE 1`

Definition at line 689 of file ncurses.h.

5.2.1.198 `#define COLOR_CYAN (COLOR_BLUE | COLOR_GREEN)`

Definition at line 694 of file ncurses.h.

5.2.1.199 `#define COLOR_GREEN 2`

Definition at line 690 of file ncurses.h.

5.2.1.200 `#define COLOR_MAGENTA (COLOR_RED | COLOR_BLUE)`

Definition at line 695 of file ncurses.h.

5.2.1.201 `#define COLOR_PAIR(n) (((ctype)(n) << PDC_COLOR_SHIFT) & A_COLOR)`

Definition at line 1552 of file ncurses.h.

5.2.1.202 `#define COLOR_RED 4`

Definition at line 691 of file ncurses.h.

5.2.1.203 `#define COLOR_WHITE 7`

Definition at line 698 of file ncurses.h.

5.2.1.204 `#define COLOR_YELLOW (COLOR_RED | COLOR_GREEN)`

Definition at line 696 of file ncurses.h.

5.2.1.205 `#define CTL_0 (KEY_OFFSET + 0x137)`

Definition at line 982 of file ncurses.h.

5.2.1.206 `#define CTL_1 (KEY_OFFSET + 0x138)`

Definition at line 983 of file ncurses.h.

5.2.1.207 `#define CTL_2 (KEY_OFFSET + 0x139)`

Definition at line 984 of file ncurses.h.

5.2.1.208 `#define CTL_3 (KEY_OFFSET + 0x13a)`

Definition at line 985 of file ncurses.h.

5.2.1.209 `#define CTL_4 (KEY_OFFSET + 0x13b)`

Definition at line 986 of file ncurses.h.

5.2.1.210 `#define CTL_5 (KEY_OFFSET + 0x13c)`

Definition at line 987 of file ncurses.h.

5.2.1.211 `#define CTL_6 (KEY_OFFSET + 0x13d)`

Definition at line 988 of file ncurses.h.

5.2.1.212 `#define CTL_7 (KEY_OFFSET + 0x13e)`

Definition at line 989 of file ncurses.h.

5.2.1.213 `#define CTL_8 (KEY_OFFSET + 0x13f)`

Definition at line 990 of file ncurses.h.

5.2.1.214 `#define CTL_9 (KEY_OFFSET + 0x140)`

Definition at line 991 of file ncurses.h.

5.2.1.215 `#define CTL_APPS (KEY_OFFSET + 0x12e)`

Definition at line 970 of file ncurses.h.

5.2.1.216 `#define CTL_BKSP (KEY_OFFSET + 0xf9) /* ctl-backspace */`

Definition at line 907 of file ncurses.h.

5.2.1.217 `#define CTL_BQUOTE (KEY_OFFSET + 0x12b)`

Definition at line 966 of file ncurses.h.

5.2.1.218 `#define CTL_COMMA (KEY_OFFSET + 0x127)`

Definition at line 962 of file ncurses.h.

5.2.1.219 `#define CTL_DEL (KEY_OFFSET + 0x10f) /* clt-delete */`

Definition at line 932 of file ncurses.h.

5.2.1.220 `#define CTL_DOWN (KEY_OFFSET + 0xe1) /* ctl-down arrow */`

Definition at line 883 of file ncurses.h.

5.2.1.221 `#define CTL_END (KEY_OFFSET + 0xc0)`

Definition at line 848 of file ncurses.h.

5.2.1.222 `#define CTL_ENTER (KEY_OFFSET + 0x111) /* ctl-enter */`

Definition at line 934 of file ncurses.h.

5.2.1.223 `#define CTL_EQUAL (KEY_OFFSET + 0x126)`

Definition at line 961 of file ncurses.h.

5.2.1.224 `#define CTL_FSLASH (KEY_OFFSET + 0x12a)`

Definition at line 965 of file ncurses.h.

5.2.1.225 `#define CTL_HOME (KEY_OFFSET + 0xbf)`

Definition at line 847 of file ncurses.h.

5.2.1.226 `#define CTL_INS (KEY_OFFSET + 0xdd) /* ctl-insert */`

Definition at line 879 of file ncurses.h.

5.2.1.227 `#define CTL_LEFT (KEY_OFFSET + 0xbb) /* Control-Left-Arrow */`

Definition at line 843 of file ncurses.h.

5.2.1.228 `#define CTL_MINUS (KEY_OFFSET + 0x128)`

Definition at line 963 of file ncurses.h.

5.2.1.229 `#define CTL_PAD0 (KEY_OFFSET + 0xfb) /* ctl-keypad 0 */`

Definition at line 910 of file ncurses.h.

5.2.1.230 `#define CTL_PAD1 (KEY_OFFSET + 0xfc)`

Definition at line 911 of file ncurses.h.

5.2.1.231 `#define CTL_PAD2 (KEY_OFFSET + 0xfd)`

Definition at line 912 of file ncurses.h.

5.2.1.232 `#define CTL_PAD3 (KEY_OFFSET + 0xfe)`

Definition at line 913 of file ncurses.h.

5.2.1.233 `#define CTL_PAD4 (KEY_OFFSET + 0xff)`

Definition at line 914 of file ncurses.h.

5.2.1.234 `#define CTL_PAD5 (KEY_OFFSET + 0x100)`

Definition at line 915 of file ncurses.h.

5.2.1.235 `#define CTL_PAD6 (KEY_OFFSET + 0x101)`

Definition at line 916 of file ncurses.h.

5.2.1.236 `#define CTL_PAD7 (KEY_OFFSET + 0x102)`

Definition at line 917 of file ncurses.h.

5.2.1.237 `#define CTL_PAD8 (KEY_OFFSET + 0x103)`

Definition at line 918 of file ncurses.h.

5.2.1.238 `#define CTL_PAD9 (KEY_OFFSET + 0x104)`

Definition at line 919 of file ncurses.h.

5.2.1.239 `#define CTL_PADCENTER (KEY_OFFSET + 0xd3) /* ctl-enter on keypad */`

Definition at line 869 of file ncurses.h.

5.2.1.240 `#define CTL_PADENTER (KEY_OFFSET + 0xcc) /* ctl-enter on keypad */`

Definition at line 862 of file ncurses.h.

5.2.1.241 `#define CTL_PADMINUS (KEY_OFFSET + 0xd5) /* ctl-minus on keypad */`

Definition at line 871 of file ncurses.h.

5.2.1.242 `#define CTL_PADPLUS (KEY_OFFSET + 0xd4) /* ctl-plus on keypad */`

Definition at line 870 of file ncurses.h.

5.2.1.243 `#define CTL_PADSLASH (KEY_OFFSET + 0xd6) /* ctl-slash on keypad */`

Definition at line 872 of file ncurses.h.

5.2.1.244 `#define CTL_PADSTAR (KEY_OFFSET + 0xd7) /* ctl-star on keypad */`

Definition at line 873 of file ncurses.h.

5.2.1.245 `#define CTL_PADSTOP (KEY_OFFSET + 0xd2) /* ctl-stop on keypad */`

Definition at line 868 of file ncurses.h.

5.2.1.246 `#define CTL_PAUSE (KEY_OFFSET + 0x132)`

Definition at line 975 of file ncurses.h.

5.2.1.247 **#define CTL_PGDN (KEY_OFFSET + 0xbe)**

Definition at line 846 of file ncurses.h.

5.2.1.248 **#define CTL_PGUP (KEY_OFFSET + 0xbd)**

Definition at line 845 of file ncurses.h.

5.2.1.249 **#define CTL_RIGHT (KEY_OFFSET + 0xbc)**

Definition at line 844 of file ncurses.h.

5.2.1.250 **#define CTL_SEMICOLON (KEY_OFFSET + 0x125)**

Definition at line 960 of file ncurses.h.

5.2.1.251 **#define CTL_STOP (KEY_OFFSET + 0x129)**

Definition at line 964 of file ncurses.h.

5.2.1.252 **#define CTL_TAB (KEY_OFFSET + 0xe2) /* ctl-tab */**

Definition at line 884 of file ncurses.h.

5.2.1.253 **#define CTL_UP (KEY_OFFSET + 0xe0) /* ctl-up arrow */**

Definition at line 882 of file ncurses.h.

5.2.1.254 **#define ERR (-1)**

Definition at line 80 of file ncurses.h.

5.2.1.255 **#define FALSE 0**

Definition at line 71 of file ncurses.h.

5.2.1.256 **#define getbegyx(w, y, x) (y = getbegy(w), x = getbegx(w))**

Definition at line 1557 of file ncurses.h.

5.2.1.257 **#define getch() wgetch(stdscr)**

Definition at line 1549 of file ncurses.h.

5.2.1.258 **#define getmaxyx(w, y, x) (y = getmaxy(w), x = getmaxx(w))**

Definition at line 1558 of file ncurses.h.

5.2.1.259 **#define** getparyx(w, y, x) (y = getpary(w), x = getparx(w))

Definition at line 1559 of file ncurses.h.

5.2.1.260 **#define** getsyx(y, x)

Value:

```
{ if (curscr->_leaveit) (y)=(x)=-1; \
  else getyx(curscr, (y), (x)); }
```

Definition at line 1562 of file ncurses.h.

5.2.1.261 **#define** getyx(w, y, x) (y = getcury(w), x = getcurx(w))

Definition at line 1560 of file ncurses.h.

5.2.1.262 **#define** KEY_A1 (KEY_OFFSET + 0xc1) /* upper left on Virtual keypad */

Definition at line 850 of file ncurses.h.

5.2.1.263 **#define** KEY_A2 (KEY_OFFSET + 0xc2) /* upper middle on Virt. keypad */

Definition at line 851 of file ncurses.h.

5.2.1.264 **#define** KEY_A3 (KEY_OFFSET + 0xc3) /* upper right on Vir. keypad */

Definition at line 852 of file ncurses.h.

5.2.1.265 **#define** KEY_ABORT (KEY_OFFSET + 0x5c) /* abort/terminate key (any) */

Definition at line 744 of file ncurses.h.

5.2.1.266 **#define** KEY_ABROWSER_BACK (KEY_OFFSET + 0x144)

Definition at line 996 of file ncurses.h.

5.2.1.267 **#define** KEY_ABROWSER_FWD (KEY_OFFSET + 0x148)

Definition at line 1000 of file ncurses.h.

5.2.1.268 **#define** KEY_ABROWSER_HOME (KEY_OFFSET + 0x15C)

Definition at line 1020 of file ncurses.h.

5.2.1.269 **#define** KEY_ABROWSER_REF (KEY_OFFSET + 0x14C)

Definition at line 1004 of file ncurses.h.

5.2.1.270 `#define KEY_ABROWSER_STOP (KEY_OFFSET + 0x150)`

Definition at line 1008 of file ncurses.h.

5.2.1.271 `#define KEY_AFAVORITES (KEY_OFFSET + 0x158)`

Definition at line 1016 of file ncurses.h.

5.2.1.272 `#define KEY_ALAUNCH_APP1 (KEY_OFFSET + 0x184)`

Definition at line 1060 of file ncurses.h.

5.2.1.273 `#define KEY_ALAUNCH_APP2 (KEY_OFFSET + 0x188)`

Definition at line 1064 of file ncurses.h.

5.2.1.274 `#define KEY_ALAUNCH_MAIL (KEY_OFFSET + 0x17C)`

Definition at line 1052 of file ncurses.h.

5.2.1.275 `#define KEY_ALT_L (KEY_OFFSET + 0x120) /* Left-alt */`

Definition at line 951 of file ncurses.h.

5.2.1.276 `#define KEY_ALT_R (KEY_OFFSET + 0x121) /* Right-alt */`

Definition at line 952 of file ncurses.h.

5.2.1.277 `#define KEY_AMEDIA_SELECT (KEY_OFFSET + 0x180)`

Definition at line 1056 of file ncurses.h.

5.2.1.278 `#define KEY_AMEDIA_STOP (KEY_OFFSET + 0x174)`

Definition at line 1044 of file ncurses.h.

5.2.1.279 `#define KEY_ANEXT_TRACK (KEY_OFFSET + 0x16C)`

Definition at line 1036 of file ncurses.h.

5.2.1.280 `#define KEY_APLAY_PAUSE (KEY_OFFSET + 0x178)`

Definition at line 1048 of file ncurses.h.

5.2.1.281 `#define KEY_APPS (KEY_OFFSET + 0x12c)`

Definition at line 968 of file ncurses.h.

5.2.1.282 `#define KEY_APREV_TRACK (KEY_OFFSET + 0x170)`

Definition at line 1040 of file ncurses.h.

5.2.1.283 `#define KEY_ASEARCH (KEY_OFFSET + 0x154)`

Definition at line 1012 of file ncurses.h.

5.2.1.284 `#define KEY_AVOLUME_DOWN (KEY_OFFSET + 0x164)`

Definition at line 1028 of file ncurses.h.

5.2.1.285 `#define KEY_AVOLUME_MUTE (KEY_OFFSET + 0x160)`

Definition at line 1024 of file ncurses.h.

5.2.1.286 `#define KEY_AVOLUME_UP (KEY_OFFSET + 0x168)`

Definition at line 1032 of file ncurses.h.

5.2.1.287 `#define KEY_B1 (KEY_OFFSET + 0xc4) /* middle left on Virt. keypad */`

Definition at line 853 of file ncurses.h.

5.2.1.288 `#define KEY_B2 (KEY_OFFSET + 0xc5) /* center on Virt. keypad */`

Definition at line 854 of file ncurses.h.

5.2.1.289 `#define KEY_B3 (KEY_OFFSET + 0xc6) /* middle right on Vir. keypad */`

Definition at line 855 of file ncurses.h.

5.2.1.290 `#define KEY_BACKSPACE (KEY_OFFSET + 0x07) /* not on pc */`

Definition at line 721 of file ncurses.h.

5.2.1.291 `#define KEY_BEG (KEY_OFFSET + 0x60) /* beg(inning) key */`

Definition at line 748 of file ncurses.h.

5.2.1.292 `#define KEY_BREAK (KEY_OFFSET + 0x01) /* Not on PC KBD */`

Definition at line 715 of file ncurses.h.

5.2.1.293 `#define KEY_BROWSER_BACK (KEY_OFFSET + 0x141)`

Definition at line 993 of file ncurses.h.

5.2.1.294 `#define KEY_BROWSER_FWD (KEY_OFFSET + 0x145)`

Definition at line 997 of file ncurses.h.

5.2.1.295 `#define KEY_BROWSER_HOME (KEY_OFFSET + 0x159)`

Definition at line 1017 of file ncurses.h.

5.2.1.296 `#define KEY_BROWSER_REF (KEY_OFFSET + 0x149)`

Definition at line 1001 of file ncurses.h.

5.2.1.297 `#define KEY_BROWSER_STOP (KEY_OFFSET + 0x14D)`

Definition at line 1005 of file ncurses.h.

5.2.1.298 `#define KEY_BTAB (KEY_OFFSET + 0x5f) /* Back tab key */`

Definition at line 747 of file ncurses.h.

5.2.1.299 `#define KEY_C1 (KEY_OFFSET + 0xc7) /* lower left on Virt. keypad */`

Definition at line 856 of file ncurses.h.

5.2.1.300 `#define KEY_C2 (KEY_OFFSET + 0xc8) /* lower middle on Virt. keypad */`

Definition at line 857 of file ncurses.h.

5.2.1.301 `#define KEY_C3 (KEY_OFFSET + 0xc9) /* lower right on Vir. keypad */`

Definition at line 858 of file ncurses.h.

5.2.1.302 `#define KEY_CANCEL (KEY_OFFSET + 0x61) /* cancel key */`

Definition at line 749 of file ncurses.h.

5.2.1.303 `#define KEY_CATAB (KEY_OFFSET + 0x56) /* clear all tabs */`

Definition at line 738 of file ncurses.h.

5.2.1.304 `#define KEY_CBROWSER_BACK (KEY_OFFSET + 0x143)`

Definition at line 995 of file ncurses.h.

5.2.1.305 `#define KEY_CBROWSER_FWD (KEY_OFFSET + 0x147)`

Definition at line 999 of file ncurses.h.

5.2.1.306 `#define KEY_CBROWSER_HOME (KEY_OFFSET + 0x15B)`

Definition at line 1019 of file ncurses.h.

5.2.1.307 `#define KEY_CBROWSER_REF (KEY_OFFSET + 0x14B)`

Definition at line 1003 of file ncurses.h.

5.2.1.308 `#define KEY_CBROWSER_STOP (KEY_OFFSET + 0x14F)`

Definition at line 1007 of file ncurses.h.

5.2.1.309 `#define KEY_CFAVORITES (KEY_OFFSET + 0x157)`

Definition at line 1015 of file ncurses.h.

5.2.1.310 `#define KEY_CLAUNCH_APP1 (KEY_OFFSET + 0x183)`

Definition at line 1059 of file ncurses.h.

5.2.1.311 `#define KEY_CLAUNCH_APP2 (KEY_OFFSET + 0x187)`

Definition at line 1063 of file ncurses.h.

5.2.1.312 `#define KEY_CLAUNCH_MAIL (KEY_OFFSET + 0x17B)`

Definition at line 1051 of file ncurses.h.

5.2.1.313 `#define KEY_CLEAR (KEY_OFFSET + 0x4d) /* clear screen */`

Definition at line 729 of file ncurses.h.

5.2.1.314 `#define KEY_CLOSE (KEY_OFFSET + 0x62) /* close key */`

Definition at line 750 of file ncurses.h.

5.2.1.315 `#define KEY_CMEDIA_SELECT (KEY_OFFSET + 0x17F)`

Definition at line 1055 of file ncurses.h.

5.2.1.316 `#define KEY_CMEDIA_STOP (KEY_OFFSET + 0x173)`

Definition at line 1043 of file ncurses.h.

5.2.1.317 `#define KEY_CNEXT_TRACK (KEY_OFFSET + 0x16B)`

Definition at line 1035 of file ncurses.h.

5.2.1.318 `#define KEY_CODE_YES (KEY_OFFSET + 0x00) /* If get_wch() gives a key code */`

Definition at line 713 of file ncurses.h.

5.2.1.319 `#define KEY_COMMAND (KEY_OFFSET + 0x63) /* cmd (command) key */`

Definition at line 751 of file ncurses.h.

5.2.1.320 `#define KEY_CONTROL_L (KEY_OFFSET + 0x11e) /* Left-control */`

Definition at line 949 of file ncurses.h.

5.2.1.321 `#define KEY_CONTROL_R (KEY_OFFSET + 0x11f) /* Right-control */`

Definition at line 950 of file ncurses.h.

5.2.1.322 `#define KEY_COPY (KEY_OFFSET + 0x64) /* copy key */`

Definition at line 752 of file ncurses.h.

5.2.1.323 `#define KEY_CPLAY_PAUSE (KEY_OFFSET + 0x177)`

Definition at line 1047 of file ncurses.h.

5.2.1.324 `#define KEY_CPREV_TRACK (KEY_OFFSET + 0x16F)`

Definition at line 1039 of file ncurses.h.

5.2.1.325 `#define KEY_CREATE (KEY_OFFSET + 0x65) /* create key */`

Definition at line 753 of file ncurses.h.

5.2.1.326 `#define KEY_CSEARCH (KEY_OFFSET + 0x153)`

Definition at line 1011 of file ncurses.h.

5.2.1.327 `#define KEY_CTAB (KEY_OFFSET + 0x55) /* clear tab */`

Definition at line 737 of file ncurses.h.

5.2.1.328 `#define KEY_CVOLUME_DOWN (KEY_OFFSET + 0x163)`

Definition at line 1027 of file ncurses.h.

5.2.1.329 `#define KEY_CVOLUME_MUTE (KEY_OFFSET + 0x15F)`

Definition at line 1023 of file ncurses.h.

5.2.1.330 `#define KEY_CVOLUME_UP (KEY_OFFSET + 0x167)`

Definition at line 1031 of file ncurses.h.

5.2.1.331 `#define KEY_DC (KEY_OFFSET + 0x4a) /* delete character */`

Definition at line 726 of file ncurses.h.

5.2.1.332 `#define KEY_DL (KEY_OFFSET + 0x48) /* delete line */`

Definition at line 724 of file ncurses.h.

5.2.1.333 `#define KEY_DOWN (KEY_OFFSET + 0x02) /* Down arrow key */`

Definition at line 716 of file ncurses.h.

5.2.1.334 `#define KEY_EIC (KEY_OFFSET + 0x4c) /* exit insert char mode */`

Definition at line 728 of file ncurses.h.

5.2.1.335 `#define KEY_END (KEY_OFFSET + 0x66) /* end key */`

Definition at line 754 of file ncurses.h.

5.2.1.336 `#define KEY_ENTER (KEY_OFFSET + 0x57) /* enter or send (unreliable) */`

Definition at line 739 of file ncurses.h.

5.2.1.337 `#define KEY_EOL (KEY_OFFSET + 0x4f) /* clear to end of line */`

Definition at line 731 of file ncurses.h.

5.2.1.338 `#define KEY_EOS (KEY_OFFSET + 0x4e) /* clear to end of screen */`

Definition at line 730 of file ncurses.h.

5.2.1.339 `#define KEY_EXIT (KEY_OFFSET + 0x67) /* exit key */`

Definition at line 755 of file ncurses.h.

5.2.1.340 `#define KEY_F(n) (KEY_F0 + (n))`

Definition at line 1069 of file ncurses.h.

5.2.1.341 `#define KEY_F0 (KEY_OFFSET + 0x08) /* function keys; 64 reserved */`

Definition at line 722 of file ncurses.h.

5.2.1.342 `#define KEY_FAVORITES (KEY_OFFSET + 0x155)`

Definition at line 1013 of file ncurses.h.

5.2.1.343 `#define KEY_FIND (KEY_OFFSET + 0x68) /* find key */`

Definition at line 756 of file ncurses.h.

5.2.1.344 `#define KEY_HELP (KEY_OFFSET + 0x69) /* help key */`

Definition at line 757 of file ncurses.h.

5.2.1.345 `#define KEY_HOME (KEY_OFFSET + 0x06) /* home key */`

Definition at line 720 of file ncurses.h.

5.2.1.346 `#define KEY_IC (KEY_OFFSET + 0x4b) /* insert char or enter ins mode */`

Definition at line 727 of file ncurses.h.

5.2.1.347 `#define KEY_IL (KEY_OFFSET + 0x49) /* insert line */`

Definition at line 725 of file ncurses.h.

5.2.1.348 `#define KEY_LAUNCH_APP1 (KEY_OFFSET + 0x181)`

Definition at line 1057 of file ncurses.h.

5.2.1.349 `#define KEY_LAUNCH_APP2 (KEY_OFFSET + 0x185)`

Definition at line 1061 of file ncurses.h.

5.2.1.350 `#define KEY_LAUNCH_MAIL (KEY_OFFSET + 0x179)`

Definition at line 1049 of file ncurses.h.

5.2.1.351 `#define KEY_LEFT (KEY_OFFSET + 0x04) /* Left arrow key */`

Definition at line 718 of file ncurses.h.

5.2.1.352 `#define KEY_LHELP (KEY_OFFSET + 0x5e) /* long help */`

Definition at line 746 of file ncurses.h.

5.2.1.353 `#define KEY_LL (KEY_OFFSET + 0x5b) /* home down/bottom (lower left) */`

Definition at line 743 of file ncurses.h.

5.2.1.354 `#define KEY_MARK (KEY_OFFSET + 0x6a) /* mark key */`

Definition at line 758 of file ncurses.h.

5.2.1.355 `#define KEY_MAX KEY_ALAUNCH_APP2 /* Maximum curses key */`

Definition at line 1067 of file ncurses.h.

5.2.1.356 `#define KEY_MEDIA_SELECT (KEY_OFFSET + 0x17D)`

Definition at line 1053 of file ncurses.h.

5.2.1.357 `#define KEY_MEDIA_STOP (KEY_OFFSET + 0x171)`

Definition at line 1041 of file ncurses.h.

5.2.1.358 `#define KEY_MESSAGE (KEY_OFFSET + 0x6b) /* message key */`

Definition at line 759 of file ncurses.h.

5.2.1.359 `#define KEY_MIN KEY_BREAK /* Minimum curses key value */`

Definition at line 1066 of file ncurses.h.

5.2.1.360 `#define KEY_MOUSE (KEY_OFFSET + 0x11b) /* "mouse" key */`

Definition at line 946 of file ncurses.h.

5.2.1.361 `#define KEY_MOVE (KEY_OFFSET + 0x6c) /* move key */`

Definition at line 760 of file ncurses.h.

5.2.1.362 `#define KEY_NEXT (KEY_OFFSET + 0x6d) /* next object key */`

Definition at line 761 of file ncurses.h.

5.2.1.363 `#define KEY_NEXT_TRACK (KEY_OFFSET + 0x169)`

Definition at line 1033 of file ncurses.h.

5.2.1.364 `#define KEY_NPAGE (KEY_OFFSET + 0x52) /* next page */`

Definition at line 734 of file ncurses.h.

5.2.1.365 `#define KEY_OFFSET 0x100`

Definition at line 710 of file ncurses.h.

5.2.1.366 `#define KEY_OPEN (KEY_OFFSET + 0x6e) /* open key */`

Definition at line 762 of file ncurses.h.

5.2.1.367 `#define KEY_OPTIONS (KEY_OFFSET + 0x6f) /* options key */`

Definition at line 763 of file ncurses.h.

5.2.1.368 `#define KEY_PAUSE (KEY_OFFSET + 0x130)`

Definition at line 973 of file ncurses.h.

5.2.1.369 `#define KEY_PLAY_PAUSE (KEY_OFFSET + 0x175)`

Definition at line 1045 of file ncurses.h.

5.2.1.370 `#define KEY_PPAGE (KEY_OFFSET + 0x53) /* previous page */`

Definition at line 735 of file ncurses.h.

5.2.1.371 `#define KEY_PREV_TRACK (KEY_OFFSET + 0x16D)`

Definition at line 1037 of file ncurses.h.

5.2.1.372 `#define KEY_PREVIOUS (KEY_OFFSET + 0x70) /* previous object key */`

Definition at line 764 of file ncurses.h.

5.2.1.373 `#define KEY_PRINT (KEY_OFFSET + 0x5a) /* print/copy */`

Definition at line 742 of file ncurses.h.

5.2.1.374 `#define KEY_PRINTSCREEN (KEY_OFFSET + 0x133)`

Definition at line 977 of file ncurses.h.

5.2.1.375 `#define KEY_REDO (KEY_OFFSET + 0x71) /* redo key */`

Definition at line 765 of file ncurses.h.

5.2.1.376 `#define KEY_REFERENCE (KEY_OFFSET + 0x72) /* ref(erence) key */`

Definition at line 766 of file ncurses.h.

5.2.1.377 `#define KEY_REFRESH (KEY_OFFSET + 0x73) /* refresh key */`

Definition at line 767 of file ncurses.h.

5.2.1.378 **#define KEY_REPLACE (KEY_OFFSET + 0x74) /* replace key */**

Definition at line 768 of file ncurses.h.

5.2.1.379 **#define KEY_RESET (KEY_OFFSET + 0x59) /* reset/hard reset (unreliable) */**

Definition at line 741 of file ncurses.h.

5.2.1.380 **#define KEY_RESIZE (KEY_OFFSET + 0x122) /* Window resize */**

Definition at line 953 of file ncurses.h.

5.2.1.381 **#define KEY_RESTART (KEY_OFFSET + 0x75) /* restart key */**

Definition at line 769 of file ncurses.h.

5.2.1.382 **#define KEY_RESUME (KEY_OFFSET + 0x76) /* resume key */**

Definition at line 770 of file ncurses.h.

5.2.1.383 **#define KEY_RIGHT (KEY_OFFSET + 0x05) /* Right arrow key */**

Definition at line 719 of file ncurses.h.

5.2.1.384 **#define KEY_SAPPS (KEY_OFFSET + 0x12d)**

Definition at line 969 of file ncurses.h.

5.2.1.385 **#define KEY_SAVE (KEY_OFFSET + 0x77) /* save key */**

Definition at line 771 of file ncurses.h.

5.2.1.386 **#define KEY_SBEG (KEY_OFFSET + 0x78) /* shifted beginning key */**

Definition at line 772 of file ncurses.h.

5.2.1.387 **#define KEY_SBROWSER_BACK (KEY_OFFSET + 0x142)**

Definition at line 994 of file ncurses.h.

5.2.1.388 **#define KEY_SBROWSER_FWD (KEY_OFFSET + 0x146)**

Definition at line 998 of file ncurses.h.

5.2.1.389 **#define KEY_SBROWSER_HOME (KEY_OFFSET + 0x15A)**

Definition at line 1018 of file ncurses.h.

5.2.1.390 `#define KEY_SBROWSER_REF (KEY_OFFSET + 0x14A)`

Definition at line 1002 of file ncurses.h.

5.2.1.391 `#define KEY_SBROWSER_STOP (KEY_OFFSET + 0x14E)`

Definition at line 1006 of file ncurses.h.

5.2.1.392 `#define KEY_SCANCEL (KEY_OFFSET + 0x79) /* shifted cancel key */`

Definition at line 773 of file ncurses.h.

5.2.1.393 `#define KEY_SCOMMAND (KEY_OFFSET + 0x7a) /* shifted command key */`

Definition at line 774 of file ncurses.h.

5.2.1.394 `#define KEY_SCOPY (KEY_OFFSET + 0x7b) /* shifted copy key */`

Definition at line 775 of file ncurses.h.

5.2.1.395 `#define KEY_SCREATE (KEY_OFFSET + 0x7c) /* shifted create key */`

Definition at line 776 of file ncurses.h.

5.2.1.396 `#define KEY_SCROLLLOCK (KEY_OFFSET + 0x135)`

Definition at line 979 of file ncurses.h.

5.2.1.397 `#define KEY_SDC (KEY_OFFSET + 0x7d) /* shifted delete char key */`

Definition at line 777 of file ncurses.h.

5.2.1.398 `#define KEY_SDL (KEY_OFFSET + 0x7e) /* shifted delete line key */`

Definition at line 778 of file ncurses.h.

5.2.1.399 `#define KEY_SDOWN (KEY_OFFSET + 0x124) /* Shifted down arrow */`

Definition at line 955 of file ncurses.h.

5.2.1.400 `#define KEY_SEARCH (KEY_OFFSET + 0x151)`

Definition at line 1009 of file ncurses.h.

5.2.1.401 `#define KEY_SELECT (KEY_OFFSET + 0x7f) /* select key */`

Definition at line 779 of file ncurses.h.

5.2.1.402 `#define KEY_SEND (KEY_OFFSET + 0x80) /* shifted end key */`

Definition at line 780 of file ncurses.h.

5.2.1.403 `#define KEY_SEOL (KEY_OFFSET + 0x81) /* shifted clear line key */`

Definition at line 781 of file ncurses.h.

5.2.1.404 `#define KEY_SEXIT (KEY_OFFSET + 0x82) /* shifted exit key */`

Definition at line 782 of file ncurses.h.

5.2.1.405 `#define KEY_SF (KEY_OFFSET + 0x50) /* scroll 1 line forward */`

Definition at line 732 of file ncurses.h.

5.2.1.406 `#define KEY_SFAVORITES (KEY_OFFSET + 0x156)`

Definition at line 1014 of file ncurses.h.

5.2.1.407 `#define KEY_SFIND (KEY_OFFSET + 0x83) /* shifted find key */`

Definition at line 783 of file ncurses.h.

5.2.1.408 `#define KEY_SHELP (KEY_OFFSET + 0x5d) /* short help */`

Definition at line 745 of file ncurses.h.

5.2.1.409 `#define KEY_SHIFT_L (KEY_OFFSET + 0x11c) /* Left-shift */`

Definition at line 947 of file ncurses.h.

5.2.1.410 `#define KEY_SHIFT_R (KEY_OFFSET + 0x11d) /* Right-shift */`

Definition at line 948 of file ncurses.h.

5.2.1.411 `#define KEY_SHOME (KEY_OFFSET + 0x84) /* shifted home key */`

Definition at line 784 of file ncurses.h.

5.2.1.412 `#define KEY_SIC (KEY_OFFSET + 0x85) /* shifted input key */`

Definition at line 785 of file ncurses.h.

5.2.1.413 `#define KEY_SLAUNCH_APP1 (KEY_OFFSET + 0x182)`

Definition at line 1058 of file ncurses.h.

5.2.1.414 `#define KEY_SLAUNCH_APP2 (KEY_OFFSET + 0x186)`

Definition at line 1062 of file ncurses.h.

5.2.1.415 `#define KEY_SLAUNCH_MAIL (KEY_OFFSET + 0x17A)`

Definition at line 1050 of file ncurses.h.

5.2.1.416 `#define KEY_SLEFT (KEY_OFFSET + 0x87) /* shifted left arrow key */`

Definition at line 787 of file ncurses.h.

5.2.1.417 `#define KEY_SMEDIA_SELECT (KEY_OFFSET + 0x17E)`

Definition at line 1054 of file ncurses.h.

5.2.1.418 `#define KEY_SMEDIA_STOP (KEY_OFFSET + 0x172)`

Definition at line 1042 of file ncurses.h.

5.2.1.419 `#define KEY_SMESSAGE (KEY_OFFSET + 0x88) /* shifted message key */`

Definition at line 788 of file ncurses.h.

5.2.1.420 `#define KEY_SMOVE (KEY_OFFSET + 0x89) /* shifted move key */`

Definition at line 789 of file ncurses.h.

5.2.1.421 `#define KEY_SNEXT (KEY_OFFSET + 0x8a) /* shifted next key */`

Definition at line 790 of file ncurses.h.

5.2.1.422 `#define KEY_SNEXT_TRACK (KEY_OFFSET + 0x16A)`

Definition at line 1034 of file ncurses.h.

5.2.1.423 `#define KEY_SOPTIONS (KEY_OFFSET + 0x8b) /* shifted options key */`

Definition at line 791 of file ncurses.h.

5.2.1.424 `#define KEY_SPAUSE (KEY_OFFSET + 0x131)`

Definition at line 974 of file ncurses.h.

5.2.1.425 `#define KEY_SPLAY_PAUSE (KEY_OFFSET + 0x176)`

Definition at line 1046 of file ncurses.h.

5.2.1.426 **#define KEY_SPREV_TRACK (KEY_OFFSET + 0x16E)**

Definition at line 1038 of file ncurses.h.

5.2.1.427 **#define KEY_SPREVIOUS (KEY_OFFSET + 0x8c) /* shifted prev key */**

Definition at line 792 of file ncurses.h.

5.2.1.428 **#define KEY_SPRINT (KEY_OFFSET + 0x8d) /* shifted print key */**

Definition at line 793 of file ncurses.h.

5.2.1.429 **#define KEY_SR (KEY_OFFSET + 0x51) /* scroll 1 line back (reverse) */**

Definition at line 733 of file ncurses.h.

5.2.1.430 **#define KEY_SREDO (KEY_OFFSET + 0x8e) /* shifted redo key */**

Definition at line 794 of file ncurses.h.

5.2.1.431 **#define KEY_SREPLACE (KEY_OFFSET + 0x8f) /* shifted replace key */**

Definition at line 795 of file ncurses.h.

5.2.1.432 **#define KEY_SRESET (KEY_OFFSET + 0x58) /* soft/reset (partial/unreliable) */**

Definition at line 740 of file ncurses.h.

5.2.1.433 **#define KEY_SRIGHT (KEY_OFFSET + 0x90) /* shifted right arrow */**

Definition at line 796 of file ncurses.h.

5.2.1.434 **#define KEY_SRSUME (KEY_OFFSET + 0x91) /* shifted resume key */**

Definition at line 797 of file ncurses.h.

5.2.1.435 **#define KEY_SSAVE (KEY_OFFSET + 0x92) /* shifted save key */**

Definition at line 798 of file ncurses.h.

5.2.1.436 **#define KEY_SSEARCH (KEY_OFFSET + 0x152)**

Definition at line 1010 of file ncurses.h.

5.2.1.437 **#define KEY_SSUSPEND (KEY_OFFSET + 0x93) /* shifted suspend key */**

Definition at line 799 of file ncurses.h.

5.2.1.438 `#define KEY_STAB (KEY_OFFSET + 0x54) /* set tab */`

Definition at line 736 of file ncurses.h.

5.2.1.439 `#define KEY_SUNDO (KEY_OFFSET + 0x94) /* shifted undo key */`

Definition at line 800 of file ncurses.h.

5.2.1.440 `#define KEY_SUP (KEY_OFFSET + 0x123) /* Shifted up arrow */`

Definition at line 954 of file ncurses.h.

5.2.1.441 `#define KEY_SUSPEND (KEY_OFFSET + 0x95) /* suspend key */`

Definition at line 801 of file ncurses.h.

5.2.1.442 `#define KEY_SVOLUME_DOWN (KEY_OFFSET + 0x162)`

Definition at line 1026 of file ncurses.h.

5.2.1.443 `#define KEY_SVOLUME_MUTE (KEY_OFFSET + 0x15E)`

Definition at line 1022 of file ncurses.h.

5.2.1.444 `#define KEY_SVOLUME_UP (KEY_OFFSET + 0x166)`

Definition at line 1030 of file ncurses.h.

5.2.1.445 `#define KEY_UNDO (KEY_OFFSET + 0x96) /* undo key */`

Definition at line 802 of file ncurses.h.

5.2.1.446 `#define KEY_UP (KEY_OFFSET + 0x03) /* Up arrow key */`

Definition at line 717 of file ncurses.h.

5.2.1.447 `#define KEY_VOLUME_DOWN (KEY_OFFSET + 0x161)`

Definition at line 1025 of file ncurses.h.

5.2.1.448 `#define KEY_VOLUME_MUTE (KEY_OFFSET + 0x15D)`

Definition at line 1021 of file ncurses.h.

5.2.1.449 `#define KEY_VOLUME_UP (KEY_OFFSET + 0x165)`

Definition at line 1029 of file ncurses.h.

5.2.1.450 `#define MOUSE_MOVED (Mouse_status.changes & PDC_MOUSE_MOVED)`

Definition at line 183 of file ncurses.h.

5.2.1.451 `#define MOUSE_POS_REPORT (Mouse_status.changes & PDC_MOUSE_POSITION)`

Definition at line 184 of file ncurses.h.

5.2.1.452 `#define MOUSE_WHEEL_DOWN (Mouse_status.changes & PDC_MOUSE_WHEEL_DOWN)`

Definition at line 188 of file ncurses.h.

5.2.1.453 `#define MOUSE_WHEEL_LEFT (Mouse_status.changes & PDC_MOUSE_WHEEL_LEFT)`

Definition at line 189 of file ncurses.h.

5.2.1.454 `#define MOUSE_WHEEL_RIGHT (Mouse_status.changes & PDC_MOUSE_WHEEL_RIGHT)`

Definition at line 190 of file ncurses.h.

5.2.1.455 `#define MOUSE_WHEEL_SCROLL 0x02000000L /* PDCurses */`

Definition at line 232 of file ncurses.h.

5.2.1.456 `#define MOUSE_WHEEL_UP (Mouse_status.changes & PDC_MOUSE_WHEEL_UP)`

Definition at line 187 of file ncurses.h.

5.2.1.457 `#define MOUSE_X_POS (Mouse_status.x)`

Definition at line 151 of file ncurses.h.

5.2.1.458 `#define MOUSE_Y_POS (Mouse_status.y)`

Definition at line 152 of file ncurses.h.

5.2.1.459 `#define NULL (void *)0`

Definition at line 77 of file ncurses.h.

5.2.1.460 `#define OK 0`

Definition at line 83 of file ncurses.h.

5.2.1.461 `#define PAD0 (KEY_OFFSET + 0xfa) /* keypad 0 */`

Definition at line 908 of file ncurses.h.

5.2.1.462 `#define PADENTER (KEY_OFFSET + 0xcb) /* enter on keypad */`

Definition at line 861 of file ncurses.h.

5.2.1.463 `#define PADMINUS (KEY_OFFSET + 0xd0) /* minus on keypad */`

Definition at line 866 of file ncurses.h.

5.2.1.464 `#define PADPLUS (KEY_OFFSET + 0xd1) /* plus on keypad */`

Definition at line 867 of file ncurses.h.

5.2.1.465 `#define PADSLASH (KEY_OFFSET + 0xca) /* slash on keypad */`

Definition at line 860 of file ncurses.h.

5.2.1.466 `#define PADSTAR (KEY_OFFSET + 0xcf) /* star on keypad */`

Definition at line 865 of file ncurses.h.

5.2.1.467 `#define PADSTOP (KEY_OFFSET + 0xce) /* stop on keypad */`

Definition at line 864 of file ncurses.h.

5.2.1.468 `#define PAIR_NUMBER(n) (((n) & A_COLOR) >> PDC_COLOR_SHIFT) & 0xff`

Definition at line 1553 of file ncurses.h.

5.2.1.469 `#define PDC_BUILD 3401`

Definition at line 34 of file ncurses.h.

5.2.1.470 `#define PDC_BUTTON_ALT 0x0020 /* PDCurses */`

Definition at line 148 of file ncurses.h.

5.2.1.471 `#define PDC_BUTTON_CONTROL 0x0010 /* PDCurses */`

Definition at line 147 of file ncurses.h.

5.2.1.472 `#define PDC_BUTTON_SHIFT 0x0008 /* PDCurses */`

Definition at line 146 of file ncurses.h.

5.2.1.473 `#define PDC_CHARTEXT_BITS 21`

Definition at line 450 of file ncurses.h.

5.2.1.474 `#define PDC_CLIP_ACCESS_ERROR 1`

Definition at line 1572 of file ncurses.h.

5.2.1.475 `#define PDC_CLIP_EMPTY 2`

Definition at line 1573 of file ncurses.h.

5.2.1.476 `#define PDC_CLIP_MEMORY_ERROR 3`

Definition at line 1574 of file ncurses.h.

5.2.1.477 `#define PDC_CLIP_SUCCESS 0`

Definition at line 1571 of file ncurses.h.

5.2.1.478 `#define PDC_COLOR_SHIFT (PDC_CHARTEXT_BITS + 12)`

Definition at line 468 of file ncurses.h.

5.2.1.479 `#define PDC_KEY_MODIFIER_ALT 4`

Definition at line 1580 of file ncurses.h.

5.2.1.480 `#define PDC_KEY_MODIFIER_CONTROL 2`

Definition at line 1579 of file ncurses.h.

5.2.1.481 `#define PDC_KEY_MODIFIER_NUMLOCK 8`

Definition at line 1581 of file ncurses.h.

5.2.1.482 `#define PDC_KEY_MODIFIER_SHIFT 1`

Definition at line 1578 of file ncurses.h.

5.2.1.483 `#define PDC_MAX_MOUSE_BUTTONS 9`

Definition at line 125 of file ncurses.h.

5.2.1.484 `#define PDC_MOUSE_MOVED 0x0008`

Definition at line 175 of file ncurses.h.

5.2.1.485 `#define PDC_MOUSE_POSITION 0x0010`

Definition at line 176 of file ncurses.h.

5.2.1.486 `#define PDC_MOUSE_WHEEL_DOWN 0x0040`

Definition at line 178 of file ncurses.h.

5.2.1.487 `#define PDC_MOUSE_WHEEL_LEFT 0x0080`

Definition at line 179 of file ncurses.h.

5.2.1.488 `#define PDC_MOUSE_WHEEL_RIGHT 0x0100`

Definition at line 180 of file ncurses.h.

5.2.1.489 `#define PDC_MOUSE_WHEEL_UP 0x0020`

Definition at line 177 of file ncurses.h.

5.2.1.490 `#define PDC_N_EXTENDED_MOUSE_BUTTONS 6`

Definition at line 126 of file ncurses.h.

5.2.1.491 `#define PDCEX extern`

Definition at line 368 of file ncurses.h.

5.2.1.492 `#define PDCURSES 1 /* PDCurses-only routines */`

Definition at line 35 of file ncurses.h.

5.2.1.493 `#define REPORT_MOUSE_POSITION 0x20000000L`

Definition at line 238 of file ncurses.h.

5.2.1.494 `#define SHF_DC (KEY_OFFSET + 0x11a) /* shift-delete on keypad */`

Definition at line 944 of file ncurses.h.

5.2.1.495 `#define SHF_DOWN (KEY_OFFSET + 0x118) /* shift-down on keypad */`

Definition at line 942 of file ncurses.h.

5.2.1.496 `#define SHF_IC (KEY_OFFSET + 0x119) /* shift-insert on keypad */`

Definition at line 943 of file ncurses.h.

5.2.1.497 `#define SHF_PADENTER (KEY_OFFSET + 0x112) /* shift-enter on keypad */`

Definition at line 936 of file ncurses.h.

5.2.1.498 #define SHF_PADMINUS (KEY_OFFSET + 0x116) /* shift-minus on keypad */

Definition at line 940 of file ncurses.h.

5.2.1.499 #define SHF_PADPLUS (KEY_OFFSET + 0x115) /* shift-plus on keypad */

Definition at line 939 of file ncurses.h.

5.2.1.500 #define SHF_PADSLASH (KEY_OFFSET + 0x113) /* shift-slash on keypad */

Definition at line 937 of file ncurses.h.

5.2.1.501 #define SHF_PADSTAR (KEY_OFFSET + 0x114) /* shift-star on keypad */

Definition at line 938 of file ncurses.h.

5.2.1.502 #define SHF_UP (KEY_OFFSET + 0x117) /* shift-up on keypad */

Definition at line 941 of file ncurses.h.

5.2.1.503 #define SYSVcurses 1 /* System V Curses routines */

Definition at line 37 of file ncurses.h.

5.2.1.504 #define TRUE 1

Definition at line 74 of file ncurses.h.

5.2.1.505 #define ungetch(ch) PDC_ungetch(ch)

Definition at line 1550 of file ncurses.h.

5.2.1.506 #define WA_ALTCHARSET A_ALTCHARSET

Definition at line 536 of file ncurses.h.

5.2.1.507 #define WA_ATTRIBUTES A_ATTRIBUTES

Definition at line 553 of file ncurses.h.

5.2.1.508 #define WA_BLINK A_BLINK

Definition at line 537 of file ncurses.h.

5.2.1.509 #define WA_BOLD A_BOLD

Definition at line 538 of file ncurses.h.

5.2.1.510 #define WA_DIM A_DIM

Definition at line 539 of file ncurses.h.

5.2.1.511 #define WA_HORIZONTAL A_NORMAL

Definition at line 548 of file ncurses.h.

5.2.1.512 #define WA_INVIS A_INVIS

Definition at line 540 of file ncurses.h.

5.2.1.513 #define WA_LEFT A_LEFTLINE

Definition at line 541 of file ncurses.h.

5.2.1.514 #define WA_LOW A_NORMAL

Definition at line 549 of file ncurses.h.

5.2.1.515 #define WA_NORMAL A_NORMAL

Definition at line 534 of file ncurses.h.

5.2.1.516 #define WA_PROTECT A_PROTECT

Definition at line 542 of file ncurses.h.

5.2.1.517 #define WA_REVERSE A_REVERSE

Definition at line 543 of file ncurses.h.

5.2.1.518 #define WA_RIGHT A_RIGHTLINE

Definition at line 544 of file ncurses.h.

5.2.1.519 #define WA_STANDOUT A_STANDOUT

Definition at line 545 of file ncurses.h.

5.2.1.520 #define WA_TOP A_NORMAL

Definition at line 550 of file ncurses.h.

5.2.1.521 #define WA_UNDERLINE A_UNDERLINE

Definition at line 546 of file ncurses.h.

5.2.1.522 #define WA_VERTICAL A_NORMAL

Definition at line 551 of file ncurses.h.

5.2.1.523 #define WHEEL_SCROLLED 0x0006 /* PDCurses */

Definition at line 143 of file ncurses.h.

5.2.1.524 #define XOPEN 1 /* X/Open Curses routines */

Definition at line 36 of file ncurses.h.

5.2.2 Typedef Documentation**5.2.2.1 typedef chtype attr_t**

Definition at line 110 of file ncurses.h.

5.2.2.2 typedef unsigned char bool

Definition at line 92 of file ncurses.h.

5.2.2.3 typedef chtype cchar_t

Definition at line 108 of file ncurses.h.

5.2.2.4 typedef uint64_t chtype

Definition at line 96 of file ncurses.h.

5.2.2.5 typedef unsigned long mmask_t

Definition at line 242 of file ncurses.h.

5.2.2.6 typedef struct _win WINDOW**5.2.3 Function Documentation****5.2.3.1 int addch (const chtype)****5.2.3.2 int addchnstr (const chtype *, int)****5.2.3.3 int addchstr (const chtype *)****5.2.3.4 int addnstr (const char *, int)****5.2.3.5 int adddrawch (chtype)****5.2.3.6 int addstr (const char *)**

5.2.3.7 `int assume_default_colors (int , int)`

5.2.3.8 `int attr_get (attr_t *, short *, void *)`

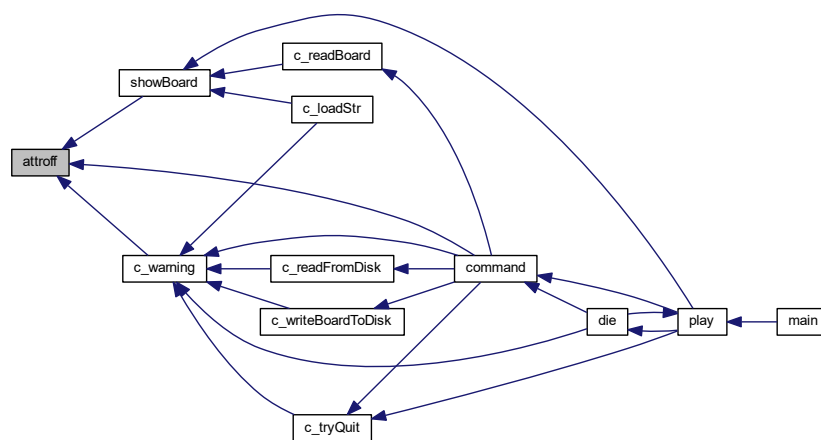
5.2.3.9 `int attr_off (attr_t , void *)`

5.2.3.10 `int attr_on (attr_t , void *)`

5.2.3.11 `int attr_set (attr_t , short , void *)`

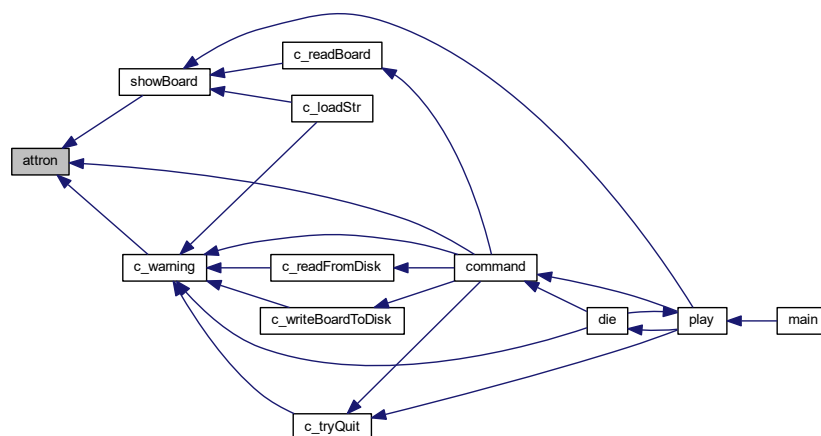
5.2.3.12 `int attroff (chtype)`

Here is the caller graph for this function:



5.2.3.13 `int attron (chtype)`

Here is the caller graph for this function:



5.2.3.14 `int attrset (chtype)`

5.2.3.15 `int baudrate (void)`

5.2.3.16 `int beep (void)`

5.2.3.17 `int bkgd (chtype)`

5.2.3.18 `void bkgdset (chtype)`

5.2.3.19 `int border (chtype , chtype , chtype , chtype , chtype , chtype , chtype , chtype)`

5.2.3.20 `int box (WINDOW *, chtype , chtype)`

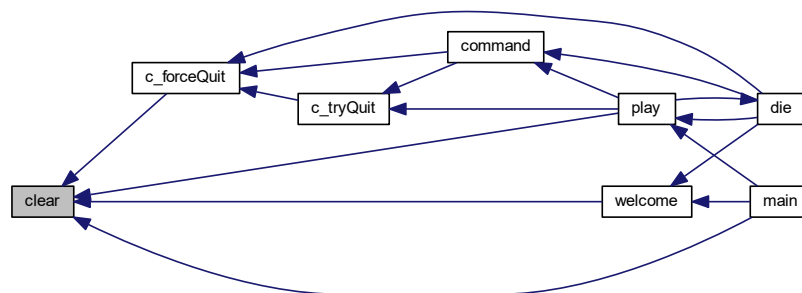
5.2.3.21 `bool can_change_color (void)`

5.2.3.22 `int cbreak (void)`

5.2.3.23 `int chgat (int , attr_t , short , const void *)`

5.2.3.24 `int clear (void)`

Here is the caller graph for this function:

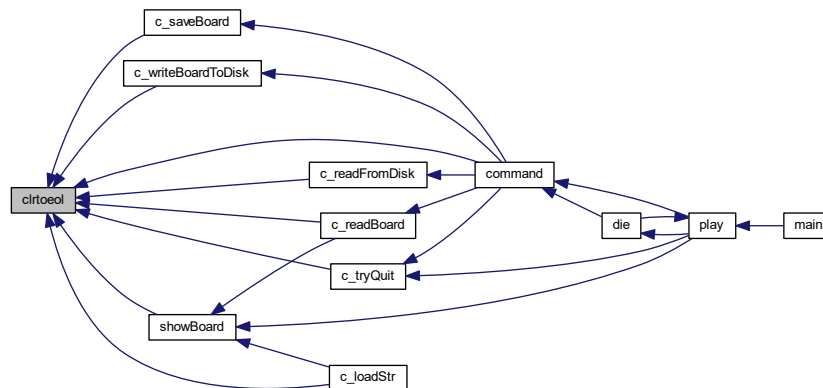


5.2.3.25 `int clearok (WINDOW *, bool)`

5.2.3.26 `int clrbot (void)`

5.2.3.27 int clrtoeol (void)

Here is the caller graph for this function:



5.2.3.28 int color_content (short, short *, short *, short *)

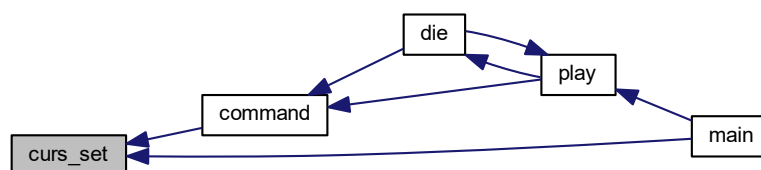
5.2.3.29 int color_set (short, void *)

5.2.3.30 int copywin (const WINDOW *, WINDOW *, int, int, int, int, int, int, int)

5.2.3.31 int crmode (void)

5.2.3.32 int curs_set (int)

Here is the caller graph for this function:



5.2.3.33 const char* curses_version (void)

5.2.3.34 int def_prog_mode (void)

5.2.3.35 int def_shell_mode (void)

5.2.3.36 int delay_output (int)

5.2.3.37 `int delch (void)`

5.2.3.38 `int deleteln (void)`

5.2.3.39 `void delscreen (SCREEN *)`

5.2.3.40 `int delwin (WINDOW *)`

5.2.3.41 `WINDOW* derwin (WINDOW *, int, int, int, int)`

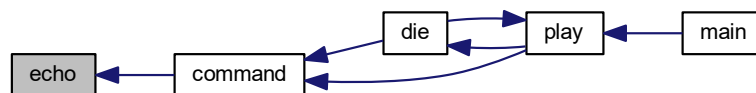
5.2.3.42 `int doupdate (void)`

5.2.3.43 `int draino (int)`

5.2.3.44 `WINDOW* dupwin (WINDOW *)`

5.2.3.45 `int echo (void)`

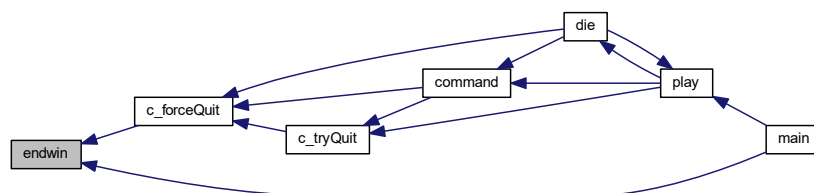
Here is the caller graph for this function:



5.2.3.46 `int echochar (const chtype)`

5.2.3.47 `int endwin (void)`

Here is the caller graph for this function:



5.2.3.48 `int erase (void)`

5.2.3.49 `char erasechar (void)`

5.2.3.50 `void filter (void)`

- 5.2.3.51 `int fixterm (void)`
- 5.2.3.52 `int flash (void)`
- 5.2.3.53 `int flushinp (void)`
- 5.2.3.54 `chtype getattrs (WINDOW *)`
- 5.2.3.55 `int getbegx (WINDOW *)`
- 5.2.3.56 `int getbegy (WINDOW *)`
- 5.2.3.57 `chtype getbkgd (WINDOW *)`
- 5.2.3.58 `unsigned long getbmap (void)`
- 5.2.3.59 `int getcurx (WINDOW *)`
- 5.2.3.60 `int getcury (WINDOW *)`
- 5.2.3.61 `int getmaxx (WINDOW *)`
- 5.2.3.62 `int getmaxy (WINDOW *)`
- 5.2.3.63 `unsigned long getmouse (void)`
- 5.2.3.64 `int getnstr (char *, int)`
- 5.2.3.65 `int getparx (WINDOW *)`
- 5.2.3.66 `int getpary (WINDOW *)`
- 5.2.3.67 `int getstr (char *)`
- 5.2.3.68 `WINDOW* getwin (FILE *)`
- 5.2.3.69 `int halfdelay (int)`
- 5.2.3.70 `bool has_colors (void)`

Here is the caller graph for this function:



- 5.2.3.71 `bool has_ic (void)`

5.2.3.72 **bool** has_il (void)

5.2.3.73 **bool** has_key (int)

5.2.3.74 **int** hline (**chtype** , int)

5.2.3.75 **void** idcok (**WINDOW** * , **bool**)

5.2.3.76 **int** idlok (**WINDOW** * , **bool**)

5.2.3.77 **void** immedok (**WINDOW** * , **bool**)

5.2.3.78 **chtype** inch (void)

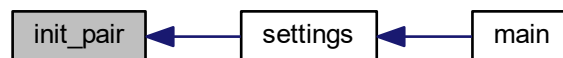
5.2.3.79 **int** inchnstr (**chtype** * , int)

5.2.3.80 **int** inchstr (**chtype** *)

5.2.3.81 **int** init_color (short , short , short , short)

5.2.3.82 **int** init_pair (short , short , short)

Here is the caller graph for this function:



5.2.3.83 **WINDOW*** initscr (void)

Here is the caller graph for this function:



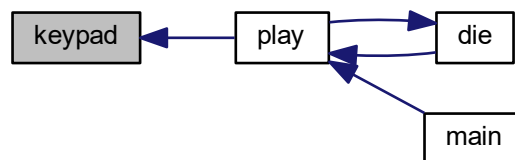
5.2.3.84 **int** innstr (**char** * , int)

5.2.3.85 **int** insch (**chtype**)

5.2.3.86 **int** insdelln (int)

- 5.2.3.87 `int insertln (void)`
- 5.2.3.88 `int insnstr (const char *, int)`
- 5.2.3.89 `int insrawch (chtype)`
- 5.2.3.90 `int insstr (const char *)`
- 5.2.3.91 `int instr (char *)`
- 5.2.3.92 `int intrflush (WINDOW *, bool)`
- 5.2.3.93 `bool is_linetouched (WINDOW *, int)`
- 5.2.3.94 `bool is_termresized (void)`
- 5.2.3.95 `bool is_wintouched (WINDOW *)`
- 5.2.3.96 `bool isendwin (void)`
- 5.2.3.97 `char* keyname (int)`
- 5.2.3.98 `int keypad (WINDOW *, bool)`

Here is the caller graph for this function:



- 5.2.3.99 `char killchar (void)`
- 5.2.3.100 `int leaveok (WINDOW *, bool)`
- 5.2.3.101 `char* longname (void)`
- 5.2.3.102 `int map_button (unsigned long)`
- 5.2.3.103 `int meta (WINDOW *, bool)`
- 5.2.3.104 `int mouse_off (unsigned long)`
- 5.2.3.105 `int mouse_on (unsigned long)`
- 5.2.3.106 `int mouse_set (unsigned long)`

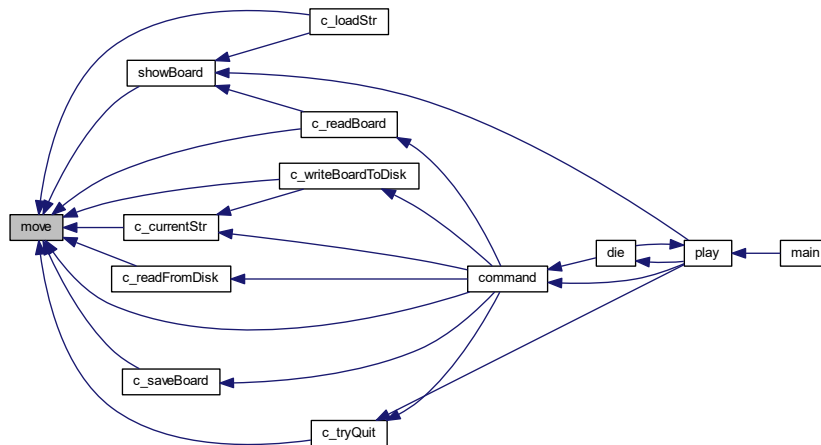
5.2.3.107 **bool** mouse_trafo (int *, int *, bool)

5.2.3.108 **int** mouseinterval (int)

5.2.3.109 **mmask_t** mousemask (**mmask_t**, **mmask_t** *)

5.2.3.110 **int** move (int, int)

Here is the caller graph for this function:



5.2.3.111 **int** mvaddch (int, int, const **chtype**)

5.2.3.112 **int** mvaddchnstr (int, int, const **chtype** *, int)

5.2.3.113 **int** mvaddchstr (int, int, const **chtype** *)

5.2.3.114 **int** mvaddnstr (int, int, const char *, int)

5.2.3.115 **int** mvaddrawch (int, int, **chtype**)

5.2.3.116 **int** mvaddstr (int, int, const char *)

5.2.3.117 **int** mvchgat (int, int, int, **attr_t**, short, const void *)

5.2.3.118 **int** mvcur (int, int, int, int)

5.2.3.119 **int** mvdelch (int, int)

5.2.3.120 **int** mvdeleteln (int, int)

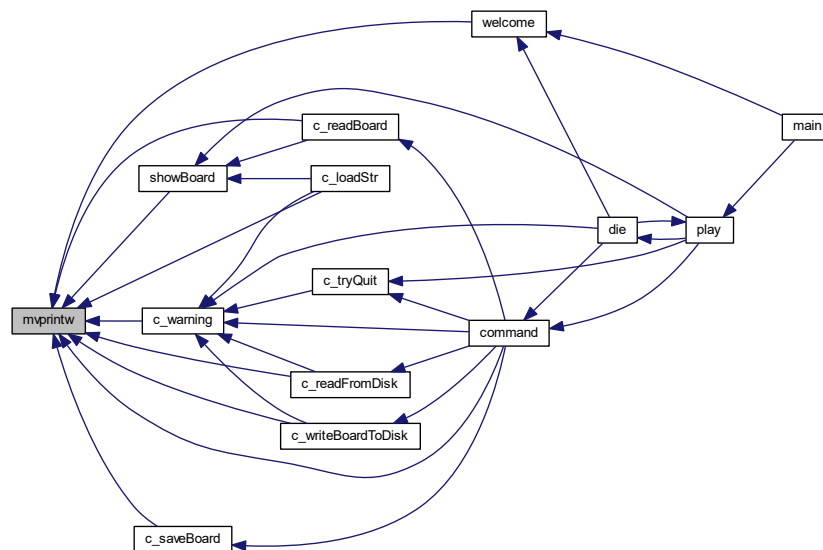
5.2.3.121 **int** mvderwin (**WINDOW** *, int, int)

5.2.3.122 **int** mvgetch (int, int)

5.2.3.123 **int** mvgetnstr (int, int, char *, int)

- 5.2.3.124 `int mvgetstr (int , int , char *)`
- 5.2.3.125 `int mvhline (int , int , chtype , int)`
- 5.2.3.126 `chtype mvinch (int , int)`
- 5.2.3.127 `int mvinchnstr (int , int , chtype * , int)`
- 5.2.3.128 `int mvinchstr (int , int , chtype *)`
- 5.2.3.129 `int mvinnstr (int , int , char * , int)`
- 5.2.3.130 `int mvinsch (int , int , chtype)`
- 5.2.3.131 `int mvinsertln (int , int)`
- 5.2.3.132 `int mvinsnstr (int , int , const char * , int)`
- 5.2.3.133 `int mvinsrawch (int , int , chtype)`
- 5.2.3.134 `int mvinsstr (int , int , const char *)`
- 5.2.3.135 `int mvinstr (int , int , char *)`
- 5.2.3.136 `int mvprintw (int , int , const char * , ...)`

Here is the caller graph for this function:



- 5.2.3.137 `int mvscanw (int , int , const char * , ...)`
- 5.2.3.138 `int mvvline (int , int , chtype , int)`
- 5.2.3.139 `int mvwaddch (WINDOW * , int , int , const chtype)`

5.2.3.140 int mvwaddchnstr (WINDOW *, int, int, const chtype *, int)

5.2.3.141 int mvwaddchstr (WINDOW *, int, int, const chtype *)

5.2.3.142 int mvwaddnstr (WINDOW *, int, int, const char *, int)

5.2.3.143 int mvwaddrawch (WINDOW *, int, int, chtype)

5.2.3.144 int mvwaddstr (WINDOW *, int, int, const char *)

5.2.3.145 int mvwchgat (WINDOW *, int, int, int, attr_t, short, const void *)

5.2.3.146 int mvwdelch (WINDOW *, int, int)

5.2.3.147 int mvwdeleteln (WINDOW *, int, int)

5.2.3.148 int mvwgetch (WINDOW *, int, int)

5.2.3.149 int mvwgetnstr (WINDOW *, int, int, char *, int)

5.2.3.150 int mvwgetstr (WINDOW *, int, int, char *)

5.2.3.151 int mvwhline (WINDOW *, int, int, chtype, int)

5.2.3.152 int mvwin (WINDOW *, int, int)

5.2.3.153 chtype mvwinch (WINDOW *, int, int)

5.2.3.154 int mvwinchnstr (WINDOW *, int, int, chtype *, int)

5.2.3.155 int mvwinchstr (WINDOW *, int, int, chtype *)

5.2.3.156 int mvwinnstr (WINDOW *, int, int, char *, int)

5.2.3.157 int mvwinsch (WINDOW *, int, int, chtype)

5.2.3.158 int mvwinsertln (WINDOW *, int, int)

5.2.3.159 int mvwinsnstr (WINDOW *, int, int, const char *, int)

5.2.3.160 int mvwinsrawch (WINDOW *, int, int, chtype)

5.2.3.161 int mvwinsstr (WINDOW *, int, int, const char *)

5.2.3.162 int mvwinstr (WINDOW *, int, int, char *)

5.2.3.163 int mvwprintw (WINDOW *, int, int, const char *, ...)

5.2.3.164 int mvwscanw (WINDOW *, int, int, const char *, ...)

5.2.3.165 int mvwvline (WINDOW *, int, int, chtype, int)

5.2.3.166 int napms (int)

5.2.3.167 int nc_getmouse (MEVENT *)

5.2.3.168 **WINDOW*** newpad (int , int)

5.2.3.169 **SCREEN*** newterm (const char * , FILE * , FILE *)

5.2.3.170 **WINDOW*** newwin (int , int , int , int)

5.2.3.171 int nl (void)

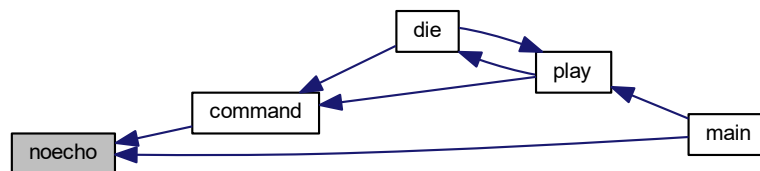
5.2.3.172 int nocbreak (void)

5.2.3.173 int nocrmode (void)

5.2.3.174 int nodelay (**WINDOW *** , bool)

5.2.3.175 int noecho (void)

Here is the caller graph for this function:



5.2.3.176 int nonl (void)

5.2.3.177 void noqiflush (void)

5.2.3.178 int noraw (void)

5.2.3.179 int notimeout (**WINDOW *** , bool)

5.2.3.180 int overlay (const **WINDOW *** , **WINDOW ***)

5.2.3.181 int overwrite (const **WINDOW *** , **WINDOW ***)

5.2.3.182 int pair_content (short , short * , short *)

5.2.3.183 int PDC_clearclipboard (void)

5.2.3.184 void PDC_debug (const char * , ...)

5.2.3.185 int PDC_freeclipboard (char *)

5.2.3.186 unsigned long PDC_get_input_fd (void)

5.2.3.187 unsigned long PDC_get_key_modifiers (void)

5.2.3.188 int PDC_getclipboard (char **, long *)

5.2.3.189 int PDC_return_key_modifiers (bool)

5.2.3.190 int PDC_save_key_modifiers (bool)

5.2.3.191 int PDC_set_blink (bool)

5.2.3.192 int PDC_set_line_color (short)

5.2.3.193 void PDC_set_resize_limits (const int *new_min_lines*, const int *new_max_lines*, const int *new_min_cols*, const int *new_max_cols*)

5.2.3.194 void PDC_set_title (const char *)

5.2.3.195 int PDC_setclipboard (const char *, long)

5.2.3.196 int PDC_ungetch (int)

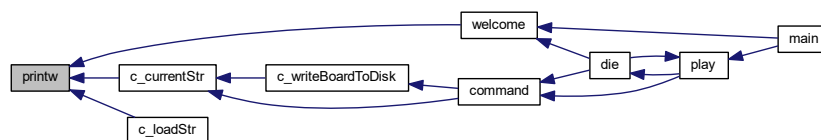
5.2.3.197 int pechochar (WINDOW *, chtype)

5.2.3.198 int pnoutrefresh (WINDOW *, int, int, int, int, int, int, int)

5.2.3.199 int prefresh (WINDOW *, int, int, int, int, int, int, int)

5.2.3.200 int printw (const char *, ...)

Here is the caller graph for this function:



5.2.3.201 int putwin (WINDOW *, FILE *)

5.2.3.202 void qiflush (void)

5.2.3.203 int raw (void)

Here is the caller graph for this function:

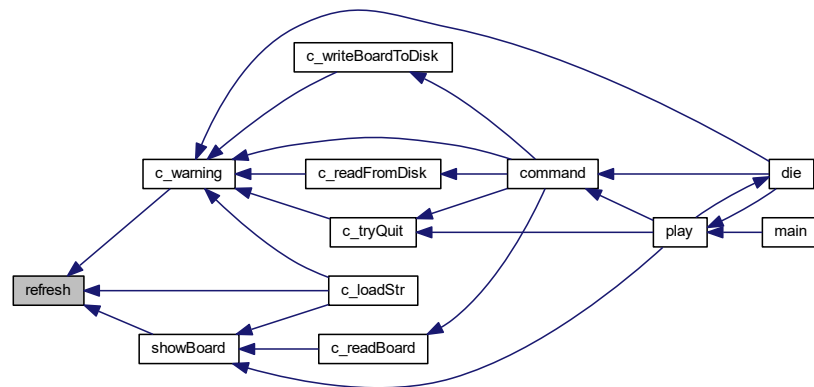


5.2.3.204 int raw_output (bool)

5.2.3.205 int redrawwin (WINDOW *)

5.2.3.206 int refresh (void)

Here is the caller graph for this function:



5.2.3.207 int request_mouse_pos (void)

5.2.3.208 int reset_prog_mode (void)

5.2.3.209 int reset_shell_mode (void)

5.2.3.210 int resetterm (void)

5.2.3.211 int resetty (void)

5.2.3.212 int resize_term (int , int)

5.2.3.213 WINDOW* resize_window (WINDOW * , int , int)

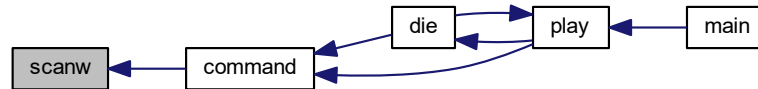
5.2.3.214 int ripoffline (int , int(*) (WINDOW *, int))

5.2.3.215 int saveterm (void)

5.2.3.216 int savetty (void)

5.2.3.217 `int scanw (const char *, ...)`

Here is the caller graph for this function:



5.2.3.218 `int scr_dump (const char *)`

5.2.3.219 `int scr_init (const char *)`

5.2.3.220 `int scr_restore (const char *)`

5.2.3.221 `int scr_set (const char *)`

5.2.3.222 `int scr1 (int)`

5.2.3.223 `int scroll (WINDOW *)`

5.2.3.224 `int scrollok (WINDOW *, bool)`

5.2.3.225 `SCREEN* set_term (SCREEN *)`

5.2.3.226 `int setscreg (int , int)`

5.2.3.227 `int setsyx (int , int)`

5.2.3.228 `int slk_attr_off (const attr_t , void *)`

5.2.3.229 `int slk_attr_on (const attr_t , void *)`

5.2.3.230 `int slk_attr_set (const attr_t , short , void *)`

5.2.3.231 `int slk_attroff (const chtype)`

5.2.3.232 `int slk_attron (const chtype)`

5.2.3.233 `int slk_attrset (const chtype)`

5.2.3.234 `int slk_clear (void)`

5.2.3.235 `int slk_color (short)`

5.2.3.236 `int slk_init (int)`

5.2.3.237 `char* slk_label (int)`

5.2.3.238 `int slk_noutrefresh (void)`

5.2.3.239 int slk_refresh (void)

5.2.3.240 int slk_restore (void)

5.2.3.241 int slk_set (int , const char * , int)

5.2.3.242 int slk_touch (void)

5.2.3.243 int standend (void)

5.2.3.244 int standout (void)

5.2.3.245 int start_color (void)

Here is the caller graph for this function:



5.2.3.246 WINDOW* subpad (WINDOW * , int , int , int , int)

5.2.3.247 WINDOW* subwin (WINDOW * , int , int , int , int)

5.2.3.248 int syncok (WINDOW * , bool)

5.2.3.249 attr_t term_attrs (void)

5.2.3.250 chtype termattrs (void)

5.2.3.251 char* termname (void)

5.2.3.252 void timeout (int)

5.2.3.253 int touchline (WINDOW * , int , int)

5.2.3.254 int touchwin (WINDOW *)

5.2.3.255 void traceoff (void)

5.2.3.256 void traceon (void)

5.2.3.257 int typeahead (int)

5.2.3.258 char* unctrl (chtype)

5.2.3.259 int ungetmouse (MEVENT *)

- 5.2.3.260 int untouchwin (WINDOW *)
- 5.2.3.261 int use_default_colors (void)
- 5.2.3.262 void use_env (bool)
- 5.2.3.263 int vid_attr (attr_t , short , void *)
- 5.2.3.264 int vid_puts (attr_t , short , void * , int(*)(int))
- 5.2.3.265 int vidattr (chtype)
- 5.2.3.266 int vidputs (chtype , int(*)(int))
- 5.2.3.267 int vline (chtype , int)
- 5.2.3.268 int vw_printw (WINDOW * , const char * , va_list)
- 5.2.3.269 int vw_scanw (WINDOW * , const char * , va_list)
- 5.2.3.270 int vwprintw (WINDOW * , const char * , va_list)
- 5.2.3.271 int vwscanw (WINDOW * , const char * , va_list)
- 5.2.3.272 int waddch (WINDOW * , const chtype)
- 5.2.3.273 int waddchnstr (WINDOW * , const chtype * , int)
- 5.2.3.274 int waddchstr (WINDOW * , const chtype *)
- 5.2.3.275 int waddnstr (WINDOW * , const char * , int)
- 5.2.3.276 int waddrawch (WINDOW * , chtype)
- 5.2.3.277 int waddstr (WINDOW * , const char *)
- 5.2.3.278 int wattr_get (WINDOW * , attr_t * , short * , void *)
- 5.2.3.279 int wattr_off (WINDOW * , attr_t , void *)
- 5.2.3.280 int wattr_on (WINDOW * , attr_t , void *)
- 5.2.3.281 int wattr_set (WINDOW * , attr_t , short , void *)
- 5.2.3.282 int wattroff (WINDOW * , chtype)
- 5.2.3.283 int wattron (WINDOW * , chtype)
- 5.2.3.284 int wattrset (WINDOW * , chtype)
- 5.2.3.285 int wbkgd (WINDOW * , chtype)
- 5.2.3.286 void wbkgdset (WINDOW * , chtype)
- 5.2.3.287 int wborder (WINDOW * , chtype , chtype , chtype , chtype , chtype , chtype , chtype , chtype)

- 5.2.3.288 int wchgat (WINDOW *, int, attr_t, short, const void *)
- 5.2.3.289 int wclear (WINDOW *)
- 5.2.3.290 int wclrtoebot (WINDOW *)
- 5.2.3.291 int wclrtoeol (WINDOW *)
- 5.2.3.292 int wcolor_set (WINDOW *, short, void *)
- 5.2.3.293 void wcursyncup (WINDOW *)
- 5.2.3.294 int wdelch (WINDOW *)
- 5.2.3.295 int wdeleteln (WINDOW *)
- 5.2.3.296 int wechochar (WINDOW *, const chtype)
- 5.2.3.297 bool wenclose (const WINDOW *, int, int)
- 5.2.3.298 int werase (WINDOW *)
- 5.2.3.299 int wgetch (WINDOW *)
- 5.2.3.300 int wgetnstr (WINDOW *, char *, int)
- 5.2.3.301 int wgetstr (WINDOW *, char *)
- 5.2.3.302 int whline (WINDOW *, chtype, int)
- 5.2.3.303 chtype winch (WINDOW *)
- 5.2.3.304 int winchnstr (WINDOW *, chtype *, int)
- 5.2.3.305 int winchstr (WINDOW *, chtype *)
- 5.2.3.306 int winnstr (WINDOW *, char *, int)
- 5.2.3.307 int winsch (WINDOW *, chtype)
- 5.2.3.308 int winsdelln (WINDOW *, int)
- 5.2.3.309 int winsertln (WINDOW *)
- 5.2.3.310 int winsnstr (WINDOW *, const char *, int)
- 5.2.3.311 int winsrawch (WINDOW *, chtype)
- 5.2.3.312 int winsstr (WINDOW *, const char *)
- 5.2.3.313 int winstr (WINDOW *, char *)
- 5.2.3.314 void wmouse_position (WINDOW *, int *, int *)
- 5.2.3.315 bool wmouse_trafo (const WINDOW *, int *, int *, bool)

- 5.2.3.316 int wmove (WINDOW *, int , int)
- 5.2.3.317 int wnoutrefresh (WINDOW *)
- 5.2.3.318 char wordchar (void)
- 5.2.3.319 int wprintw (WINDOW *, const char *, ...)
- 5.2.3.320 int wredrawln (WINDOW *, int , int)
- 5.2.3.321 int wrefresh (WINDOW *)
- 5.2.3.322 int wresize (WINDOW *, int , int)
- 5.2.3.323 int wscanw (WINDOW *, const char *, ...)
- 5.2.3.324 int wscr1 (WINDOW *, int)
- 5.2.3.325 int wsetscrreg (WINDOW *, int , int)
- 5.2.3.326 int wstandend (WINDOW *)
- 5.2.3.327 int wstandout (WINDOW *)
- 5.2.3.328 void wsyncdown (WINDOW *)
- 5.2.3.329 void wsyncup (WINDOW *)
- 5.2.3.330 void wtimeout (WINDOW *, int)
- 5.2.3.331 int wtouchln (WINDOW *, int , int , int)
- 5.2.3.332 int wvline (WINDOW *, chtype , int)
- 5.2.3.333 WINDOW* Xinitscr (int , char **)

5.2.4 Variable Documentation

5.2.4.1 PDCEX chtype acs_map[]

Definition at line 380 of file ncurses.h.

5.2.4.2 PDCEX int COLOR_PAIRS

Definition at line 378 of file ncurses.h.

5.2.4.3 PDCEX int COLORS

Definition at line 377 of file ncurses.h.

5.2.4.4 PDCEX int COLS

Definition at line 372 of file ncurses.h.

5.2.4.5 PDCEX WINDOW* curscr

Definition at line 374 of file ncurses.h.

5.2.4.6 PDCEX int LINES

Definition at line 371 of file ncurses.h.

5.2.4.7 PDCEX MOUSE_STATUS Mouse_status

Definition at line 376 of file ncurses.h.

5.2.4.8 PDCEX SCREEN* SP

Definition at line 375 of file ncurses.h.

5.2.4.9 PDCEX WINDOW* stdscr

Definition at line 373 of file ncurses.h.

5.2.4.10 PDCEX int TABSIZE

Definition at line 379 of file ncurses.h.

5.2.4.11 PDCEX char ttytype[]

Definition at line 381 of file ncurses.h.

5.3 README.md File Reference

Index

- [_attrs](#)
 - [_win](#), [8](#)
- [_begx](#)
 - [_win](#), [8](#)
- [_begy](#)
 - [_win](#), [8](#)
- [_bkgd](#)
 - [_win](#), [8](#)
- [_bmarg](#)
 - [_win](#), [8](#)
- [_clear](#)
 - [_win](#), [8](#)
- [_curx](#)
 - [_win](#), [8](#)
- [_cury](#)
 - [_win](#), [8](#)
- [_delayms](#)
 - [_win](#), [8](#)
- [_firstch](#)
 - [_win](#), [8](#)
- [_flags](#)
 - [_win](#), [9](#)
- [_immed](#)
 - [_win](#), [9](#)
- [_lastch](#)
 - [_win](#), [9](#)
- [_leaveit](#)
 - [_win](#), [9](#)
- [_map_mbe_to_key](#)
 - [SCREEN](#), [13](#)
- [_maxx](#)
 - [_win](#), [9](#)
- [_maxy](#)
 - [_win](#), [9](#)
- [_nodelay](#)
 - [_win](#), [9](#)
- [_parent](#)
 - [_win](#), [9](#)
- [_parx](#)
 - [_win](#), [9](#)
- [_pary](#)
 - [_win](#), [9](#)
- [_preserve](#)
 - [SCREEN](#), [13](#)
- [_restore](#)
 - [SCREEN](#), [13](#)
- [_scroll](#)
 - [_win](#), [9](#)
- [_sync](#)
 - [_win](#), [9](#)

- [_tmarg](#)
 - [_win](#), [10](#)
- [_trap_mbe](#)
 - [SCREEN](#), [13](#)
- [_use_keypad](#)
 - [_win](#), [10](#)
- [_win](#), [7](#)
 - [_attrs](#), [8](#)
 - [_begx](#), [8](#)
 - [_begy](#), [8](#)
 - [_bkgd](#), [8](#)
 - [_bmarg](#), [8](#)
 - [_clear](#), [8](#)
 - [_curx](#), [8](#)
 - [_cury](#), [8](#)
 - [_delayms](#), [8](#)
 - [_firstch](#), [8](#)
 - [_flags](#), [9](#)
 - [_immed](#), [9](#)
 - [_lastch](#), [9](#)
 - [_leaveit](#), [9](#)
 - [_maxx](#), [9](#)
 - [_maxy](#), [9](#)
 - [_nodelay](#), [9](#)
 - [_parent](#), [9](#)
 - [_parx](#), [9](#)
 - [_pary](#), [9](#)
 - [_scroll](#), [9](#)
 - [_sync](#), [9](#)
 - [_tmarg](#), [10](#)
 - [_use_keypad](#), [10](#)
 - [_y](#), [10](#)
- [_y](#)
 - [_win](#), [10](#)

- [2048.c](#), [17](#)
 - [AlignCol](#), [20](#)
 - [AlignLine](#), [21](#)
 - [board](#), [45](#)
 - [boardseed](#), [45](#)
 - [boardstr](#), [45](#)
 - [c_checksum](#), [21](#)
 - [c_currentStr](#), [22](#)
 - [c_forceQuit](#), [22](#)
 - [c_loadStr](#), [23](#)
 - [c_readBoard](#), [24](#)
 - [c_readFromDisk](#), [25](#)
 - [c_saveBoard](#), [26](#)
 - [c_tryQuit](#), [27](#)

- c_warning, 28
- c_writeBoardToDisk, 29
- CheckEat, 30
- Clrboard, 31
- col, 45
- command, 31
- cs_pwd, 46
- curs, 46
- die, 33
- Display, 35
- display, 46
- EDOWN, 19
- ELEFT, 19
- ERIGHT, 19
- EUP, 19
- Eat, 35
- eat, 46
- EatCol, 36
- EatLine, 37
- GetRandNums, 38
- MAX_BOARD_NUM, 19
- MAX_BOARD_SIZE, 20
- MAX_RANDOM, 46
- MENU_POSITION_X, 20
- MENU_POSITION_Y, 20
- main, 39
- N, 46
- NA, 46
- P_RANDOM, 46
- PWD, 20
- PWD_LEN, 20
- play, 40
- point, 46
- Rando, 42
- row, 47
- settings, 42
- showBoard, 43
- WARNING_POSITION_X, 20
- WARNING_POSITION_Y, 20
- welcome, 44
- A_ALTCHARSET
 - ncurses.h, 64
- A_ATTRIBUTES
 - ncurses.h, 64
- A_BLINK
 - ncurses.h, 64
- A_BOLD
 - ncurses.h, 64
- A_BUTTON_CHANGED
 - ncurses.h, 64
- A_CHARTEXT
 - ncurses.h, 64
- A_COLOR
 - ncurses.h, 64
- A_DIM
 - ncurses.h, 64
- A_INVIS
 - ncurses.h, 65
- A_ITALIC
 - ncurses.h, 65
- A_LEFTLINE
 - ncurses.h, 65
- A_NORMAL
 - ncurses.h, 65
- A_OVERLINE
 - ncurses.h, 65
- A_PROTECT
 - ncurses.h, 65
- A_REVERSE
 - ncurses.h, 65
- A_RGB
 - ncurses.h, 65
- A_RGB_COLOR
 - ncurses.h, 65
- A_RIGHTLINE
 - ncurses.h, 65
- A_STANDOUT
 - ncurses.h, 65
- A_STRIKEOUT
 - ncurses.h, 66
- A_UNDERLINE
 - ncurses.h, 66
- ACS_BBSS
 - ncurses.h, 66
- ACS_BLOCK
 - ncurses.h, 66
- ACS_BOARD
 - ncurses.h, 66
- ACS_BSBS
 - ncurses.h, 66
- ACS_BSSB
 - ncurses.h, 66
- ACS_BSSS
 - ncurses.h, 66
- ACS_BTEE
 - ncurses.h, 66
- ACS_BULLET
 - ncurses.h, 66
- ACS_CKBOARD
 - ncurses.h, 66
- ACS_DARROW
 - ncurses.h, 66
- ACS_DEGREE
 - ncurses.h, 67
- ACS_DIAMOND
 - ncurses.h, 67
- ACS_GEQUAL
 - ncurses.h, 67
- ACS_HLINE
 - ncurses.h, 67
- ACS_LANTERN
 - ncurses.h, 67
- ACS_LARROW
 - ncurses.h, 67
- ACS_LEQUAL
 - ncurses.h, 67

ACS_LLCORNER
 ncurses.h, [67](#)
ACS_LRCORNER
 ncurses.h, [67](#)
ACS_LTEE
 ncurses.h, [67](#)
ACS_NEQUAL
 ncurses.h, [67](#)
ACS_PI
 ncurses.h, [67](#)
ACS_PICK
 ncurses.h, [68](#)
ACS_PLMINUS
 ncurses.h, [68](#)
ACS_PLUS
 ncurses.h, [68](#)
ACS_RARROW
 ncurses.h, [68](#)
ACS_RTEE
 ncurses.h, [68](#)
ACS_S1
 ncurses.h, [68](#)
ACS_S3
 ncurses.h, [68](#)
ACS_S7
 ncurses.h, [68](#)
ACS_S9
 ncurses.h, [68](#)
ACS_SBBS
 ncurses.h, [68](#)
ACS_SBSB
 ncurses.h, [68](#)
ACS_SBSS
 ncurses.h, [68](#)
ACS_SSBB
 ncurses.h, [69](#)
ACS_SSBS
 ncurses.h, [69](#)
ACS_SSSB
 ncurses.h, [69](#)
ACS_SSSS
 ncurses.h, [69](#)
ACS_STERLING
 ncurses.h, [69](#)
ACS_TTEE
 ncurses.h, [69](#)
ACS_UARROW
 ncurses.h, [69](#)
ACS_ULCORNER
 ncurses.h, [69](#)
ACS_URCORNER
 ncurses.h, [69](#)
ACS_VLINE
 ncurses.h, [69](#)
ALL_MOUSE_EVENTS
 ncurses.h, [69](#)
ALT_0
 ncurses.h, [69](#)
ALT_1
 ncurses.h, [70](#)
ALT_2
 ncurses.h, [70](#)
ALT_3
 ncurses.h, [70](#)
ALT_4
 ncurses.h, [70](#)
ALT_5
 ncurses.h, [70](#)
ALT_6
 ncurses.h, [70](#)
ALT_7
 ncurses.h, [70](#)
ALT_8
 ncurses.h, [70](#)
ALT_9
 ncurses.h, [70](#)
ALT_A
 ncurses.h, [70](#)
ALT_APPS
 ncurses.h, [70](#)
ALT_B
 ncurses.h, [70](#)
ALT_BKSP
 ncurses.h, [71](#)
ALT_BQUOTE
 ncurses.h, [71](#)
ALT_BSLASH
 ncurses.h, [71](#)
ALT_C
 ncurses.h, [71](#)
ALT_COMMA
 ncurses.h, [71](#)
ALT_D
 ncurses.h, [71](#)
ALT_DEL
 ncurses.h, [71](#)
ALT_DOWN
 ncurses.h, [71](#)
ALT_E
 ncurses.h, [71](#)
ALT_END
 ncurses.h, [71](#)
ALT_ENTER
 ncurses.h, [71](#)
ALT_EQUAL
 ncurses.h, [71](#)
ALT_ESC
 ncurses.h, [72](#)
ALT_F
 ncurses.h, [72](#)
ALT_FQUOTE
 ncurses.h, [72](#)
ALT_FSLASH
 ncurses.h, [72](#)
ALT_G
 ncurses.h, [72](#)

ALT_H
 ncurses.h, 72

ALT_HOME
 ncurses.h, 72

ALT_I
 ncurses.h, 72

ALT_INS
 ncurses.h, 72

ALT_J
 ncurses.h, 72

ALT_K
 ncurses.h, 72

ALT_L
 ncurses.h, 72

ALT_LBRACKET
 ncurses.h, 73

ALT_LEFT
 ncurses.h, 73

ALT_M
 ncurses.h, 73

ALT_MINUS
 ncurses.h, 73

ALT_N
 ncurses.h, 73

ALT_O
 ncurses.h, 73

ALT_P
 ncurses.h, 73

ALT_PAD0
 ncurses.h, 73

ALT_PAD1
 ncurses.h, 73

ALT_PAD2
 ncurses.h, 73

ALT_PAD3
 ncurses.h, 73

ALT_PAD4
 ncurses.h, 73

ALT_PAD5
 ncurses.h, 74

ALT_PAD6
 ncurses.h, 74

ALT_PAD7
 ncurses.h, 74

ALT_PAD8
 ncurses.h, 74

ALT_PAD9
 ncurses.h, 74

ALT_PADENTER
 ncurses.h, 74

ALT_PADMINUS
 ncurses.h, 74

ALT_PADPLUS
 ncurses.h, 74

ALT_PADSLASH
 ncurses.h, 74

ALT_PADSTAR
 ncurses.h, 74

ALT_PADSTOP
 ncurses.h, 74

ALT_PGDN
 ncurses.h, 74

ALT_PGUP
 ncurses.h, 75

ALT_PRINTSCREEN
 ncurses.h, 75

ALT_Q
 ncurses.h, 75

ALT_R
 ncurses.h, 75

ALT_RBRACKET
 ncurses.h, 75

ALT_RIGHT
 ncurses.h, 75

ALT_S
 ncurses.h, 75

ALT_SCROLLLOCK
 ncurses.h, 75

ALT_SEMICOLON
 ncurses.h, 75

ALT_STOP
 ncurses.h, 75

ALT_T
 ncurses.h, 75

ALT_TAB
 ncurses.h, 75

ALT_U
 ncurses.h, 76

ALT_UP
 ncurses.h, 76

ALT_V
 ncurses.h, 76

ALT_W
 ncurses.h, 76

ALT_X
 ncurses.h, 76

ALT_Y
 ncurses.h, 76

ALT_Z
 ncurses.h, 76

ATR_MSK
 ncurses.h, 76

ATR_NRM
 ncurses.h, 76

acs_map
 ncurses.h, 126

addch
 ncurses.h, 108

addchnstr
 ncurses.h, 108

addchstr
 ncurses.h, 108

addnstr
 ncurses.h, 108

addrdrawch
 ncurses.h, 108

addstr
 ncurses.h, 108
AlignCol
 2048.c, 20
AlignLine
 2048.c, 21
alive
 SCREEN, 13
assume_default_colors
 ncurses.h, 108
attr_get
 ncurses.h, 109
attr_off
 ncurses.h, 109
attr_on
 ncurses.h, 109
attr_set
 ncurses.h, 109
attr_t
 ncurses.h, 108
attroff
 ncurses.h, 109
attron
 ncurses.h, 109
attrset
 ncurses.h, 109
audible
 SCREEN, 13
autocr
 SCREEN, 13

BSDcurses
 ncurses.h, 76
BUTTON1_CLICKED
 ncurses.h, 76
BUTTON1_DOUBLE_CLICKED
 ncurses.h, 76
BUTTON1_MOVED
 ncurses.h, 77
BUTTON1_PRESSED
 ncurses.h, 77
BUTTON1_RELEASED
 ncurses.h, 77
BUTTON1_TRIPLE_CLICKED
 ncurses.h, 77
BUTTON2_CLICKED
 ncurses.h, 77
BUTTON2_DOUBLE_CLICKED
 ncurses.h, 77
BUTTON2_MOVED
 ncurses.h, 77
BUTTON2_PRESSED
 ncurses.h, 77
BUTTON2_RELEASED
 ncurses.h, 77
BUTTON2_TRIPLE_CLICKED
 ncurses.h, 77
BUTTON3_CLICKED
 ncurses.h, 77
BUTTON3_DOUBLE_CLICKED
 ncurses.h, 77
BUTTON3_MOVED
 ncurses.h, 78
BUTTON3_PRESSED
 ncurses.h, 78
BUTTON3_RELEASED
 ncurses.h, 78
BUTTON3_TRIPLE_CLICKED
 ncurses.h, 78
BUTTON4_CLICKED
 ncurses.h, 78
BUTTON4_DOUBLE_CLICKED
 ncurses.h, 78
BUTTON4_PRESSED
 ncurses.h, 78
BUTTON4_RELEASED
 ncurses.h, 78
BUTTON4_TRIPLE_CLICKED
 ncurses.h, 78
BUTTON5_CLICKED
 ncurses.h, 78
BUTTON5_DOUBLE_CLICKED
 ncurses.h, 78
BUTTON5_PRESSED
 ncurses.h, 78
BUTTON5_RELEASED
 ncurses.h, 79
BUTTON5_TRIPLE_CLICKED
 ncurses.h, 79
BUTTON_ACTION_MASK
 ncurses.h, 79
BUTTON_ALT
 ncurses.h, 79
BUTTON_CHANGED
 ncurses.h, 79
BUTTON_CLICKED
 ncurses.h, 79
BUTTON_CONTROL
 ncurses.h, 79
BUTTON_DOUBLE_CLICKED
 ncurses.h, 79
BUTTON_MODIFIER_ALT
 ncurses.h, 79
BUTTON_MODIFIER_CONTROL
 ncurses.h, 79
BUTTON_MODIFIER_MASK
 ncurses.h, 79
BUTTON_MODIFIER_SHIFT
 ncurses.h, 79
BUTTON_MOVED
 ncurses.h, 80
BUTTON_PRESSED
 ncurses.h, 80
BUTTON_RELEASED
 ncurses.h, 80
BUTTON_SHIFT
 ncurses.h, 80

- BUTTON_STATUS
 - ncurses.h, [80](#)
- BUTTON_TRIPLE_CLICKED
 - ncurses.h, [80](#)
- baudrate
 - ncurses.h, [110](#)
- beep
 - ncurses.h, [110](#)
- bkgd
 - ncurses.h, [110](#)
- bkgdset
 - ncurses.h, [110](#)
- board
 - 2048.c, [45](#)
- boardseed
 - 2048.c, [45](#)
- boardstr
 - 2048.c, [45](#)
- bool
 - ncurses.h, [108](#)
- border
 - ncurses.h, [110](#)
- box
 - ncurses.h, [110](#)
- bstate
 - MEVENT, [10](#)
- button
 - MOUSE_STATUS, [11](#)
- c_checksum
 - 2048.c, [21](#)
- c_currentStr
 - 2048.c, [22](#)
- c_forceQuit
 - 2048.c, [22](#)
- c_loadStr
 - 2048.c, [23](#)
- c_readBoard
 - 2048.c, [24](#)
- c_readFromDisk
 - 2048.c, [25](#)
- c_saveBoard
 - 2048.c, [26](#)
- c_tryQuit
 - 2048.c, [27](#)
- c_warning
 - 2048.c, [28](#)
- c_writeBoardToDisk
 - 2048.c, [29](#)
- CHR_MSK
 - ncurses.h, [80](#)
- CHTYPE_LONG
 - ncurses.h, [80](#)
- COLOR_BLACK
 - ncurses.h, [80](#)
- COLOR_BLUE
 - ncurses.h, [80](#)
- COLOR_CYAN
 - ncurses.h, [80](#)
- COLOR_GREEN
 - ncurses.h, [80](#)
- COLOR_MAGENTA
 - ncurses.h, [81](#)
- COLOR_PAIR
 - ncurses.h, [81](#)
- COLOR_PAIRS
 - ncurses.h, [126](#)
- COLOR_RED
 - ncurses.h, [81](#)
- COLOR_WHITE
 - ncurses.h, [81](#)
- COLOR_YELLOW
 - ncurses.h, [81](#)
- COLORS
 - ncurses.h, [126](#)
- COLS
 - ncurses.h, [126](#)
- CTL_0
 - ncurses.h, [81](#)
- CTL_1
 - ncurses.h, [81](#)
- CTL_2
 - ncurses.h, [81](#)
- CTL_3
 - ncurses.h, [81](#)
- CTL_4
 - ncurses.h, [81](#)
- CTL_5
 - ncurses.h, [81](#)
- CTL_6
 - ncurses.h, [81](#)
- CTL_7
 - ncurses.h, [82](#)
- CTL_8
 - ncurses.h, [82](#)
- CTL_9
 - ncurses.h, [82](#)
- CTL_APPS
 - ncurses.h, [82](#)
- CTL_BKSP
 - ncurses.h, [82](#)
- CTL_BQUOTE
 - ncurses.h, [82](#)
- CTL_COMMA
 - ncurses.h, [82](#)
- CTL_DEL
 - ncurses.h, [82](#)
- CTL_DOWN
 - ncurses.h, [82](#)
- CTL_END
 - ncurses.h, [82](#)
- CTL_ENTER
 - ncurses.h, [82](#)
- CTL_EQUAL
 - ncurses.h, [82](#)
- CTL_FSLASH
 - ncurses.h, [83](#)

- CTL_HOME
 - ncurses.h, [83](#)
- CTL_INS
 - ncurses.h, [83](#)
- CTL_LEFT
 - ncurses.h, [83](#)
- CTL_MINUS
 - ncurses.h, [83](#)
- CTL_PAD0
 - ncurses.h, [83](#)
- CTL_PAD1
 - ncurses.h, [83](#)
- CTL_PAD2
 - ncurses.h, [83](#)
- CTL_PAD3
 - ncurses.h, [83](#)
- CTL_PAD4
 - ncurses.h, [83](#)
- CTL_PAD5
 - ncurses.h, [83](#)
- CTL_PAD6
 - ncurses.h, [83](#)
- CTL_PAD7
 - ncurses.h, [84](#)
- CTL_PAD8
 - ncurses.h, [84](#)
- CTL_PAD9
 - ncurses.h, [84](#)
- CTL_PADCENTER
 - ncurses.h, [84](#)
- CTL_PADENTER
 - ncurses.h, [84](#)
- CTL_PADMINUS
 - ncurses.h, [84](#)
- CTL_PADPLUS
 - ncurses.h, [84](#)
- CTL_PADSLASH
 - ncurses.h, [84](#)
- CTL_PADSTAR
 - ncurses.h, [84](#)
- CTL_PADSTOP
 - ncurses.h, [84](#)
- CTL_PAUSE
 - ncurses.h, [84](#)
- CTL_PGDN
 - ncurses.h, [84](#)
- CTL_PGUP
 - ncurses.h, [85](#)
- CTL_RIGHT
 - ncurses.h, [85](#)
- CTL_SEMICOLON
 - ncurses.h, [85](#)
- CTL_STOP
 - ncurses.h, [85](#)
- CTL_TAB
 - ncurses.h, [85](#)
- CTL_UP
 - ncurses.h, [85](#)
- can_change_color
 - ncurses.h, [110](#)
- cbreak
 - ncurses.h, [110](#)
- SCREEN, [13](#)
- cchar_t
 - ncurses.h, [108](#)
- changes
 - MOUSE_STATUS, [11](#)
- CheckEat
 - 2048.c, [30](#)
- chgat
 - ncurses.h, [110](#)
- chtype
 - ncurses.h, [108](#)
- clear
 - ncurses.h, [110](#)
- clearok
 - ncurses.h, [110](#)
- Clrboard
 - 2048.c, [31](#)
- clrtobot
 - ncurses.h, [110](#)
- clrtoeol
 - ncurses.h, [110](#)
- col
 - 2048.c, [45](#)
- color_content
 - ncurses.h, [111](#)
- color_set
 - ncurses.h, [111](#)
- cols
 - SCREEN, [13](#)
- command
 - 2048.c, [31](#)
- copywin
 - ncurses.h, [111](#)
- crmode
 - ncurses.h, [111](#)
- cs_pwd
 - 2048.c, [46](#)
- curs
 - 2048.c, [46](#)
- curs_set
 - ncurses.h, [111](#)
- curscol
 - SCREEN, [14](#)
- curscr
 - ncurses.h, [126](#)
- curses_version
 - ncurses.h, [111](#)
- cursrow
 - SCREEN, [14](#)
- def_prog_mode
 - ncurses.h, [111](#)
- def_shell_mode
 - ncurses.h, [111](#)
- delay_output

- ncurses.h, 111
- delaytenths
 - SCREEN, 14
- delch
 - ncurses.h, 111
- deleteln
 - ncurses.h, 112
- delscreen
 - ncurses.h, 112
- delwin
 - ncurses.h, 112
- derwin
 - ncurses.h, 112
- die
 - 2048.c, 33
- Display
 - 2048.c, 35
- display
 - 2048.c, 46
- doupdate
 - ncurses.h, 112
- draino
 - ncurses.h, 112
- dupwin
 - ncurses.h, 112
- EDOWN
 - 2048.c, 19
- ELEFT
 - 2048.c, 19
- ERIGHT
 - 2048.c, 19
- ERR
 - ncurses.h, 85
- EUP
 - 2048.c, 19
- Eat
 - 2048.c, 35
- eat
 - 2048.c, 46
- EatCol
 - 2048.c, 36
- EatLine
 - 2048.c, 37
- echo
 - ncurses.h, 112
 - SCREEN, 14
- echochar
 - ncurses.h, 112
- endwin
 - ncurses.h, 112
- erase
 - ncurses.h, 112
- erasechar
 - ncurses.h, 112
- FALSE
 - ncurses.h, 85
- filter
 - ncurses.h, 112
- fixterm
 - ncurses.h, 112
- flash
 - ncurses.h, 113
- flushinp
 - ncurses.h, 113
- GetRandNums
 - 2048.c, 38
- getattrs
 - ncurses.h, 113
- getbegx
 - ncurses.h, 113
- getbegy
 - ncurses.h, 113
- getbegyx
 - ncurses.h, 85
- getbkgd
 - ncurses.h, 113
- getbmap
 - ncurses.h, 113
- getch
 - ncurses.h, 85
- getcurx
 - ncurses.h, 113
- getcury
 - ncurses.h, 113
- getmaxx
 - ncurses.h, 113
- getmaxy
 - ncurses.h, 113
- getmaxyx
 - ncurses.h, 85
- getmouse
 - ncurses.h, 113
- getnstr
 - ncurses.h, 113
- getparx
 - ncurses.h, 113
- getpary
 - ncurses.h, 113
- getparyx
 - ncurses.h, 85
- getstr
 - ncurses.h, 113
- getsyx
 - ncurses.h, 86
- getwin
 - ncurses.h, 113
- getyx
 - ncurses.h, 86
- halfdelay
 - ncurses.h, 113
- has_colors
 - ncurses.h, 113
- has_ic
 - ncurses.h, 113

has_il
 ncurses.h, 113

has_key
 ncurses.h, 114

hline
 ncurses.h, 114

id
 MEVENT, 10

idcok
 ncurses.h, 114

idlok
 ncurses.h, 114

immedok
 ncurses.h, 114

inch
 ncurses.h, 114

inchnstr
 ncurses.h, 114

inchstr
 ncurses.h, 114

init_color
 ncurses.h, 114

init_pair
 ncurses.h, 114

initscr
 ncurses.h, 114

innstr
 ncurses.h, 114

insch
 ncurses.h, 114

insdelln
 ncurses.h, 114

insertln
 ncurses.h, 114

insnstr
 ncurses.h, 115

insrawch
 ncurses.h, 115

insstr
 ncurses.h, 115

instr
 ncurses.h, 115

intrflush
 ncurses.h, 115

is_linetouched
 ncurses.h, 115

is_termresized
 ncurses.h, 115

is_wintouched
 ncurses.h, 115

isendwin
 ncurses.h, 115

KEY_A1
 ncurses.h, 86

KEY_A2
 ncurses.h, 86

KEY_A3
 ncurses.h, 86

KEY_ABORT
 ncurses.h, 86

KEY_ABROWSER_BACK
 ncurses.h, 86

KEY_ABROWSER_FWD
 ncurses.h, 86

KEY_ABROWSER_HOME
 ncurses.h, 86

KEY_ABROWSER_REF
 ncurses.h, 86

KEY_ABROWSER_STOP
 ncurses.h, 86

KEY_AFAVORITES
 ncurses.h, 87

KEY_ALAUNCH_APP1
 ncurses.h, 87

KEY_ALAUNCH_APP2
 ncurses.h, 87

KEY_ALAUNCH_MAIL
 ncurses.h, 87

KEY_ALT_L
 ncurses.h, 87

KEY_ALT_R
 ncurses.h, 87

KEY_AMEDIA_SELECT
 ncurses.h, 87

KEY_AMEDIA_STOP
 ncurses.h, 87

KEY_ANEXT_TRACK
 ncurses.h, 87

KEY_APLAY_PAUSE
 ncurses.h, 87

KEY_APPS
 ncurses.h, 87

KEY_APREV_TRACK
 ncurses.h, 87

KEY_ASEARCH
 ncurses.h, 88

KEY_AVOLUME_DOWN
 ncurses.h, 88

KEY_AVOLUME_MUTE
 ncurses.h, 88

KEY_AVOLUME_UP
 ncurses.h, 88

KEY_B1
 ncurses.h, 88

KEY_B2
 ncurses.h, 88

KEY_B3
 ncurses.h, 88

KEY_BACKSPACE
 ncurses.h, 88

KEY_BEG
 ncurses.h, 88

KEY_BREAK
 ncurses.h, 88

KEY_BROWSER_BACK

ncurses.h, [88](#)
KEY_BROWSER_FWD
ncurses.h, [88](#)
KEY_BROWSER_HOME
ncurses.h, [89](#)
KEY_BROWSER_REF
ncurses.h, [89](#)
KEY_BROWSER_STOP
ncurses.h, [89](#)
KEY_BTAB
ncurses.h, [89](#)
KEY_C1
ncurses.h, [89](#)
KEY_C2
ncurses.h, [89](#)
KEY_C3
ncurses.h, [89](#)
KEY_CANCEL
ncurses.h, [89](#)
KEY_CATAB
ncurses.h, [89](#)
KEY_CBROWSER_BACK
ncurses.h, [89](#)
KEY_CBROWSER_FWD
ncurses.h, [89](#)
KEY_CBROWSER_HOME
ncurses.h, [89](#)
KEY_CBROWSER_REF
ncurses.h, [90](#)
KEY_CBROWSER_STOP
ncurses.h, [90](#)
KEY_CFAVORITES
ncurses.h, [90](#)
KEY_CLAUNCH_APP1
ncurses.h, [90](#)
KEY_CLAUNCH_APP2
ncurses.h, [90](#)
KEY_CLAUNCH_MAIL
ncurses.h, [90](#)
KEY_CLEAR
ncurses.h, [90](#)
KEY_CLOSE
ncurses.h, [90](#)
KEY_CMEDIA_SELECT
ncurses.h, [90](#)
KEY_CMEDIA_STOP
ncurses.h, [90](#)
KEY_CNEXT_TRACK
ncurses.h, [90](#)
KEY_CODE_YES
ncurses.h, [90](#)
KEY_COMMAND
ncurses.h, [91](#)
KEY_CONTROL_L
ncurses.h, [91](#)
KEY_CONTROL_R
ncurses.h, [91](#)
KEY_COPY
ncurses.h, [91](#)
KEY_CPLAY_PAUSE
ncurses.h, [91](#)
KEY_CPREV_TRACK
ncurses.h, [91](#)
KEY_CREATE
ncurses.h, [91](#)
KEY_CSEARCH
ncurses.h, [91](#)
KEY_CTAB
ncurses.h, [91](#)
KEY_CVOLUME_DOWN
ncurses.h, [91](#)
KEY_CVOLUME_MUTE
ncurses.h, [91](#)
KEY_CVOLUME_UP
ncurses.h, [91](#)
KEY_DC
ncurses.h, [92](#)
KEY_DL
ncurses.h, [92](#)
KEY_DOWN
ncurses.h, [92](#)
KEY_EIC
ncurses.h, [92](#)
KEY_END
ncurses.h, [92](#)
KEY_ENTER
ncurses.h, [92](#)
KEY_EOL
ncurses.h, [92](#)
KEY_EOS
ncurses.h, [92](#)
KEY_EXIT
ncurses.h, [92](#)
KEY_F
ncurses.h, [92](#)
KEY_F0
ncurses.h, [92](#)
KEY_FAVORITES
ncurses.h, [92](#)
KEY_FIND
ncurses.h, [93](#)
KEY_HELP
ncurses.h, [93](#)
KEY_HOME
ncurses.h, [93](#)
KEY_IC
ncurses.h, [93](#)
KEY_IL
ncurses.h, [93](#)
KEY_LAUNCH_APP1
ncurses.h, [93](#)
KEY_LAUNCH_APP2
ncurses.h, [93](#)
KEY_LAUNCH_MAIL
ncurses.h, [93](#)
KEY_LEFT

ncurses.h, 93
KEY_LHELP
ncurses.h, 93
KEY_LL
ncurses.h, 93
KEY_MARK
ncurses.h, 93
KEY_MAX
ncurses.h, 94
KEY_MEDIA_SELECT
ncurses.h, 94
KEY_MEDIA_STOP
ncurses.h, 94
KEY_MESSAGE
ncurses.h, 94
KEY_MIN
ncurses.h, 94
KEY_MOUSE
ncurses.h, 94
KEY_MOVE
ncurses.h, 94
KEY_NEXT
ncurses.h, 94
KEY_NEXT_TRACK
ncurses.h, 94
KEY_NPAGE
ncurses.h, 94
KEY_OFFSET
ncurses.h, 94
KEY_OPEN
ncurses.h, 94
KEY_OPTIONS
ncurses.h, 95
KEY_PAUSE
ncurses.h, 95
KEY_PLAY_PAUSE
ncurses.h, 95
KEY_PPAGE
ncurses.h, 95
KEY_PREV_TRACK
ncurses.h, 95
KEY_PREVIOUS
ncurses.h, 95
KEY_PRINT
ncurses.h, 95
KEY_PRINTSCREEN
ncurses.h, 95
KEY_REDO
ncurses.h, 95
KEY_REFERENCE
ncurses.h, 95
KEY_REFRESH
ncurses.h, 95
KEY_REPLACE
ncurses.h, 95
KEY_RESET
ncurses.h, 96
KEY_RESIZE
ncurses.h, 96
KEY_RESTART
ncurses.h, 96
KEY_RESUME
ncurses.h, 96
KEY_RIGHT
ncurses.h, 96
KEY_SAPPS
ncurses.h, 96
KEY_SAVE
ncurses.h, 96
KEY_SBEG
ncurses.h, 96
KEY_SBROWSER_BACK
ncurses.h, 96
KEY_SBROWSER_FWD
ncurses.h, 96
KEY_SBROWSER_HOME
ncurses.h, 96
KEY_SBROWSER_REF
ncurses.h, 96
KEY_SBROWSER_STOP
ncurses.h, 97
KEY_SCANCEL
ncurses.h, 97
KEY_SCOMMAND
ncurses.h, 97
KEY_SCOPY
ncurses.h, 97
KEY_SCREATE
ncurses.h, 97
KEY_SCROLLLOCK
ncurses.h, 97
KEY_SDC
ncurses.h, 97
KEY_SDL
ncurses.h, 97
KEY_SDOWN
ncurses.h, 97
KEY_SEARCH
ncurses.h, 97
KEY_SELECT
ncurses.h, 97
KEY_SEND
ncurses.h, 97
KEY_SEOL
ncurses.h, 98
KEY_SEXIT
ncurses.h, 98
KEY_SF
ncurses.h, 98
KEY_SFAVORITES
ncurses.h, 98
KEY_SFIND
ncurses.h, 98
KEY_SHELP
ncurses.h, 98
KEY_SHIFT_L

- ncurses.h, 98
- KEY_SHIFT_R
 - ncurses.h, 98
- KEY_SHOME
 - ncurses.h, 98
- KEY_SIC
 - ncurses.h, 98
- KEY_SLAUNCH_APP1
 - ncurses.h, 98
- KEY_SLAUNCH_APP2
 - ncurses.h, 98
- KEY_SLAUNCH_MAIL
 - ncurses.h, 99
- KEY_SLEFT
 - ncurses.h, 99
- KEY_SMEDIA_SELECT
 - ncurses.h, 99
- KEY_SMEDIA_STOP
 - ncurses.h, 99
- KEY_SMESSAGE
 - ncurses.h, 99
- KEY_SMOVE
 - ncurses.h, 99
- KEY_SNEXT
 - ncurses.h, 99
- KEY_SNEXT_TRACK
 - ncurses.h, 99
- KEY_SOPTIONS
 - ncurses.h, 99
- KEY_SPAUSE
 - ncurses.h, 99
- KEY_SPLAY_PAUSE
 - ncurses.h, 99
- KEY_SPREV_TRACK
 - ncurses.h, 99
- KEY_SPREVIOUS
 - ncurses.h, 100
- KEY_SPRINT
 - ncurses.h, 100
- KEY_SR
 - ncurses.h, 100
- KEY_SREDO
 - ncurses.h, 100
- KEY_SREPLACE
 - ncurses.h, 100
- KEY_SRESET
 - ncurses.h, 100
- KEY_SRIGHT
 - ncurses.h, 100
- KEY_SRSUME
 - ncurses.h, 100
- KEY_SSAVE
 - ncurses.h, 100
- KEY_SSEARCH
 - ncurses.h, 100
- KEY_SSUSPEND
 - ncurses.h, 100
- KEY_STAB
 - ncurses.h, 100
- KEY_SUNDO
 - ncurses.h, 101
- KEY_SUP
 - ncurses.h, 101
- KEY_SUSPEND
 - ncurses.h, 101
- KEY_SVOLUME_DOWN
 - ncurses.h, 101
- KEY_SVOLUME_MUTE
 - ncurses.h, 101
- KEY_SVOLUME_UP
 - ncurses.h, 101
- KEY_UNDO
 - ncurses.h, 101
- KEY_UP
 - ncurses.h, 101
- KEY_VOLUME_DOWN
 - ncurses.h, 101
- KEY_VOLUME_MUTE
 - ncurses.h, 101
- KEY_VOLUME_UP
 - ncurses.h, 101
- key_code
 - SCREEN, 14
- keyname
 - ncurses.h, 115
- keypad
 - ncurses.h, 115
- killchar
 - ncurses.h, 115
- LINES
 - ncurses.h, 127
- leaveok
 - ncurses.h, 115
- line_color
 - SCREEN, 14
- lines
 - SCREEN, 14
- linesrippedoff
 - SCREEN, 14
- linesrippedoffontop
 - SCREEN, 14
- longname
 - ncurses.h, 115
- MAX_BOARD_NUM
 - 2048.c, 19
- MAX_BOARD_SIZE
 - 2048.c, 20
- MAX_RANDOM
 - 2048.c, 46
- MENU_POSITION_X
 - 2048.c, 20
- MENU_POSITION_Y
 - 2048.c, 20
- MEVENT, 10
 - bstate, 10

- id, [10](#)
- x, [10](#)
- y, [10](#)
- z, [11](#)
- MOUSE_MOVED
 - [ncurses.h](#), [101](#)
- MOUSE_POS_REPORT
 - [ncurses.h](#), [102](#)
- MOUSE_STATUS, [11](#)
 - button, [11](#)
 - changes, [11](#)
 - x, [11](#)
 - xbutton, [11](#)
 - y, [11](#)
- MOUSE_WHEEL_DOWN
 - [ncurses.h](#), [102](#)
- MOUSE_WHEEL_LEFT
 - [ncurses.h](#), [102](#)
- MOUSE_WHEEL_RIGHT
 - [ncurses.h](#), [102](#)
- MOUSE_WHEEL_SCROLL
 - [ncurses.h](#), [102](#)
- MOUSE_WHEEL_UP
 - [ncurses.h](#), [102](#)
- MOUSE_X_POS
 - [ncurses.h](#), [102](#)
- MOUSE_Y_POS
 - [ncurses.h](#), [102](#)
- main
 - 2048.c, [39](#)
- map_button
 - [ncurses.h](#), [115](#)
- meta
 - [ncurses.h](#), [115](#)
- mmask_t
 - [ncurses.h](#), [108](#)
- mono
 - SCREEN, [14](#)
- mouse_off
 - [ncurses.h](#), [115](#)
- mouse_on
 - [ncurses.h](#), [115](#)
- mouse_set
 - [ncurses.h](#), [115](#)
- Mouse_status
 - [ncurses.h](#), [127](#)
- mouse_trafo
 - [ncurses.h](#), [115](#)
- mouse_wait
 - SCREEN, [14](#)
- mouseinterval
 - [ncurses.h](#), [116](#)
- mousemask
 - [ncurses.h](#), [116](#)
- move
 - [ncurses.h](#), [116](#)
- mvaddch
 - [ncurses.h](#), [116](#)
- mvaddchnstr
 - [ncurses.h](#), [116](#)
- mvaddchstr
 - [ncurses.h](#), [116](#)
- mvaddnstr
 - [ncurses.h](#), [116](#)
- mvaddrawch
 - [ncurses.h](#), [116](#)
- mvaddstr
 - [ncurses.h](#), [116](#)
- mvchgat
 - [ncurses.h](#), [116](#)
- mvcur
 - [ncurses.h](#), [116](#)
- mvdelch
 - [ncurses.h](#), [116](#)
- mvdeleteln
 - [ncurses.h](#), [116](#)
- mvderwin
 - [ncurses.h](#), [116](#)
- mvgetch
 - [ncurses.h](#), [116](#)
- mvgetnstr
 - [ncurses.h](#), [116](#)
- mvgetstr
 - [ncurses.h](#), [116](#)
- mvhline
 - [ncurses.h](#), [117](#)
- mvinch
 - [ncurses.h](#), [117](#)
- mvinchnstr
 - [ncurses.h](#), [117](#)
- mvinchstr
 - [ncurses.h](#), [117](#)
- mvinnstr
 - [ncurses.h](#), [117](#)
- mvinsch
 - [ncurses.h](#), [117](#)
- mvinsertln
 - [ncurses.h](#), [117](#)
- mvinsnstr
 - [ncurses.h](#), [117](#)
- mvinsrawch
 - [ncurses.h](#), [117](#)
- mvinsstr
 - [ncurses.h](#), [117](#)
- mvinstr
 - [ncurses.h](#), [117](#)
- mvprintw
 - [ncurses.h](#), [117](#)
- mvscanw
 - [ncurses.h](#), [117](#)
- mvvline
 - [ncurses.h](#), [117](#)
- mvwaddch
 - [ncurses.h](#), [117](#)
- mvwaddchnstr
 - [ncurses.h](#), [117](#)

- mvwaddchstr
 - ncurses.h, [118](#)
- mvwaddnstr
 - ncurses.h, [118](#)
- mvwaddrawch
 - ncurses.h, [118](#)
- mvwaddstr
 - ncurses.h, [118](#)
- mvwchgat
 - ncurses.h, [118](#)
- mvwdelch
 - ncurses.h, [118](#)
- mvwdeleteln
 - ncurses.h, [118](#)
- mvwgetch
 - ncurses.h, [118](#)
- mvwgetnstr
 - ncurses.h, [118](#)
- mvwgetstr
 - ncurses.h, [118](#)
- mvwhline
 - ncurses.h, [118](#)
- mvwin
 - ncurses.h, [118](#)
- mvwinch
 - ncurses.h, [118](#)
- mvwinchnstr
 - ncurses.h, [118](#)
- mvwinchstr
 - ncurses.h, [118](#)
- mvwinnstr
 - ncurses.h, [118](#)
- mvwinsch
 - ncurses.h, [118](#)
- mvwinserln
 - ncurses.h, [118](#)
- mvwinsnstr
 - ncurses.h, [118](#)
- mvwinsrawch
 - ncurses.h, [118](#)
- mvwinsstr
 - ncurses.h, [118](#)
- mvwinstr
 - ncurses.h, [118](#)
- mvwprintw
 - ncurses.h, [118](#)
- mvwscanw
 - ncurses.h, [118](#)
- mvwvline
 - ncurses.h, [118](#)
- N
 - 2048.c, [46](#)
- NA
 - 2048.c, [46](#)
- NULL
 - ncurses.h, [102](#)
- napms
 - ncurses.h, [118](#)
- nc_getmouse
 - ncurses.h, [118](#)
- ncurses.h, [47](#)
 - A_ALTCHARSET, [64](#)
 - A_ATTRIBUTES, [64](#)
 - A_BLINK, [64](#)
 - A_BOLD, [64](#)
 - A_BUTTON_CHANGED, [64](#)
 - A_CHARTTEXT, [64](#)
 - A_COLOR, [64](#)
 - A_DIM, [64](#)
 - A_INVIS, [65](#)
 - A_ITALIC, [65](#)
 - A_LEFTLINE, [65](#)
 - A_NORMAL, [65](#)
 - A_OVERLINE, [65](#)
 - A_PROTECT, [65](#)
 - A_REVERSE, [65](#)
 - A_RGB, [65](#)
 - A_RGB_COLOR, [65](#)
 - A_RIGHTLINE, [65](#)
 - A_STANDOUT, [65](#)
 - A_STRIKEOUT, [66](#)
 - A_UNDERLINE, [66](#)
 - ACS_BBSS, [66](#)
 - ACS_BLOCK, [66](#)
 - ACS_BOARD, [66](#)
 - ACS_BSBS, [66](#)
 - ACS_BSSB, [66](#)
 - ACS_BSSS, [66](#)
 - ACS_BTEE, [66](#)
 - ACS_BULLET, [66](#)
 - ACS_CKBOARD, [66](#)
 - ACS_DARROW, [66](#)
 - ACS_DEGREE, [67](#)
 - ACS_DIAMOND, [67](#)
 - ACS_GEQUAL, [67](#)
 - ACS_HLINE, [67](#)
 - ACS_LANTERN, [67](#)
 - ACS_LARROW, [67](#)
 - ACS_LEQUAL, [67](#)
 - ACS_LLCORNER, [67](#)
 - ACS_LRCORNER, [67](#)
 - ACS_LTEE, [67](#)
 - ACS_NEQUAL, [67](#)
 - ACS_PI, [67](#)
 - ACS_PICK, [68](#)
 - ACS_PLMINUS, [68](#)
 - ACS_PLUS, [68](#)
 - ACS_RARROW, [68](#)
 - ACS_RTEE, [68](#)
 - ACS_S1, [68](#)
 - ACS_S3, [68](#)
 - ACS_S7, [68](#)
 - ACS_S9, [68](#)
 - ACS_SBBS, [68](#)
 - ACS_SBSB, [68](#)
 - ACS_SBSS, [68](#)

ACS_SSBB, 69
ACS_SSBS, 69
ACS_SSSB, 69
ACS_SSSS, 69
ACS_STERLING, 69
ACS_TTEE, 69
ACS_UARROW, 69
ACS_ULCORNER, 69
ACS_URCORNER, 69
ACS_VLINE, 69
ALL_MOUSE_EVENTS, 69
ALT_0, 69
ALT_1, 70
ALT_2, 70
ALT_3, 70
ALT_4, 70
ALT_5, 70
ALT_6, 70
ALT_7, 70
ALT_8, 70
ALT_9, 70
ALT_A, 70
ALT_APPS, 70
ALT_B, 70
ALT_BKSP, 71
ALT_BQUOTE, 71
ALT_BSLASH, 71
ALT_C, 71
ALT_COMMA, 71
ALT_D, 71
ALT_DEL, 71
ALT_DOWN, 71
ALT_E, 71
ALT_END, 71
ALT_ENTER, 71
ALT_EQUAL, 71
ALT_ESC, 72
ALT_F, 72
ALT_FQUOTE, 72
ALT_FSLASH, 72
ALT_G, 72
ALT_H, 72
ALT_HOME, 72
ALT_I, 72
ALT_INS, 72
ALT_J, 72
ALT_K, 72
ALT_L, 72
ALT_LBRACKET, 73
ALT_LEFT, 73
ALT_M, 73
ALT_MINUS, 73
ALT_N, 73
ALT_O, 73
ALT_P, 73
ALT_PAD0, 73
ALT_PAD1, 73
ALT_PAD2, 73
ALT_PAD3, 73
ALT_PAD4, 73
ALT_PAD5, 74
ALT_PAD6, 74
ALT_PAD7, 74
ALT_PAD8, 74
ALT_PAD9, 74
ALT_PADENTER, 74
ALT_PADMINUS, 74
ALT_PADPLUS, 74
ALT_PADSLASH, 74
ALT_PADSTAR, 74
ALT_PADSTOP, 74
ALT_PGDN, 74
ALT_PGUP, 75
ALT_PRINTSCREEN, 75
ALT_Q, 75
ALT_R, 75
ALT_RBRACKET, 75
ALT_RIGHT, 75
ALT_S, 75
ALT_SCROLLLOCK, 75
ALT_SEMICOLON, 75
ALT_STOP, 75
ALT_T, 75
ALT_TAB, 75
ALT_U, 76
ALT_UP, 76
ALT_V, 76
ALT_W, 76
ALT_X, 76
ALT_Y, 76
ALT_Z, 76
ATR_MSK, 76
ATR_NRM, 76
acs_map, 126
addch, 108
addchnstr, 108
addchstr, 108
addnstr, 108
addrawch, 108
addstr, 108
assume_default_colors, 108
attr_get, 109
attr_off, 109
attr_on, 109
attr_set, 109
attr_t, 108
attroff, 109
attron, 109
attrset, 109
BSDcurses, 76
BUTTON1_CLICKED, 76
BUTTON1_DOUBLE_CLICKED, 76
BUTTON1_MOVED, 77
BUTTON1_PRESSED, 77
BUTTON1_RELEASED, 77
BUTTON1_TRIPLE_CLICKED, 77

BUTTON2_CLICKED, 77
BUTTON2_DOUBLE_CLICKED, 77
BUTTON2_MOVED, 77
BUTTON2_PRESSED, 77
BUTTON2_RELEASED, 77
BUTTON2_TRIPLE_CLICKED, 77
BUTTON3_CLICKED, 77
BUTTON3_DOUBLE_CLICKED, 77
BUTTON3_MOVED, 78
BUTTON3_PRESSED, 78
BUTTON3_RELEASED, 78
BUTTON3_TRIPLE_CLICKED, 78
BUTTON4_CLICKED, 78
BUTTON4_DOUBLE_CLICKED, 78
BUTTON4_PRESSED, 78
BUTTON4_RELEASED, 78
BUTTON4_TRIPLE_CLICKED, 78
BUTTON5_CLICKED, 78
BUTTON5_DOUBLE_CLICKED, 78
BUTTON5_PRESSED, 78
BUTTON5_RELEASED, 79
BUTTON5_TRIPLE_CLICKED, 79
BUTTON_ACTION_MASK, 79
BUTTON_ALT, 79
BUTTON_CHANGED, 79
BUTTON_CLICKED, 79
BUTTON_CONTROL, 79
BUTTON_DOUBLE_CLICKED, 79
BUTTON_MODIFIER_ALT, 79
BUTTON_MODIFIER_CONTROL, 79
BUTTON_MODIFIER_MASK, 79
BUTTON_MODIFIER_SHIFT, 79
BUTTON_MOVED, 80
BUTTON_PRESSED, 80
BUTTON_RELEASED, 80
BUTTON_SHIFT, 80
BUTTON_STATUS, 80
BUTTON_TRIPLE_CLICKED, 80
baudrate, 110
beep, 110
bkgd, 110
bkgdset, 110
bool, 108
border, 110
box, 110
CHR_MSK, 80
CHTYPE_LONG, 80
COLOR_BLACK, 80
COLOR_BLUE, 80
COLOR_CYAN, 80
COLOR_GREEN, 80
COLOR_MAGENTA, 81
COLOR_PAIR, 81
COLOR_PAIRS, 126
COLOR_RED, 81
COLOR_WHITE, 81
COLOR_YELLOW, 81
COLORS, 126
COLS, 126
CTL_0, 81
CTL_1, 81
CTL_2, 81
CTL_3, 81
CTL_4, 81
CTL_5, 81
CTL_6, 81
CTL_7, 82
CTL_8, 82
CTL_9, 82
CTL_APPS, 82
CTL_BKSP, 82
CTL_BQUOTE, 82
CTL_COMMA, 82
CTL_DEL, 82
CTL_DOWN, 82
CTL_END, 82
CTL_ENTER, 82
CTL_EQUAL, 82
CTL_FSLASH, 83
CTL_HOME, 83
CTL_INS, 83
CTL_LEFT, 83
CTL_MINUS, 83
CTL_PAD0, 83
CTL_PAD1, 83
CTL_PAD2, 83
CTL_PAD3, 83
CTL_PAD4, 83
CTL_PAD5, 83
CTL_PAD6, 83
CTL_PAD7, 84
CTL_PAD8, 84
CTL_PAD9, 84
CTL_PADCENTER, 84
CTL_PADENTER, 84
CTL_PADMINUS, 84
CTL_PADPLUS, 84
CTL_PADSLASH, 84
CTL_PADSTAR, 84
CTL_PADSTOP, 84
CTL_PAUSE, 84
CTL_PGDN, 84
CTL_PGUP, 85
CTL_RIGHT, 85
CTL_SEMICOLON, 85
CTL_STOP, 85
CTL_TAB, 85
CTL_UP, 85
can_change_color, 110
cbreak, 110
cchar_t, 108
chgat, 110
chtype, 108
clear, 110
clearok, 110
clrtoeb, 110

clrtoeol, 110
color_content, 111
color_set, 111
copywin, 111
crmode, 111
curs_set, 111
curscr, 126
curses_version, 111
def_prog_mode, 111
def_shell_mode, 111
delay_output, 111
delch, 111
deleteln, 112
delscreen, 112
delwin, 112
derwin, 112
doupdate, 112
draino, 112
dupwin, 112
ERR, 85
echo, 112
echochar, 112
endwin, 112
erase, 112
erasechar, 112
FALSE, 85
filter, 112
fixterm, 112
flash, 113
flushinp, 113
getattrs, 113
getbegx, 113
getbegy, 113
getbegyx, 85
getbkgd, 113
getbmap, 113
getch, 85
getcurx, 113
getcury, 113
getmaxx, 113
getmaxy, 113
getmaxyx, 85
getmouse, 113
getnstr, 113
getparx, 113
getpary, 113
getparyx, 85
getstr, 113
getsyx, 86
getwin, 113
getyx, 86
halfdelay, 113
has_colors, 113
has_ic, 113
has_il, 113
has_key, 114
hline, 114
idcok, 114
idlok, 114
immedok, 114
inch, 114
inchnstr, 114
inchstr, 114
init_color, 114
init_pair, 114
initscr, 114
innstr, 114
insch, 114
insdelln, 114
insertln, 114
insnstr, 115
insrawch, 115
insstr, 115
instr, 115
intrflush, 115
is_linetouched, 115
is_termresized, 115
is_wintouched, 115
isendwin, 115
KEY_A1, 86
KEY_A2, 86
KEY_A3, 86
KEY_ABORT, 86
KEY_ABROWSER_BACK, 86
KEY_ABROWSER_FWD, 86
KEY_ABROWSER_HOME, 86
KEY_ABROWSER_REF, 86
KEY_ABROWSER_STOP, 86
KEY_AFAVORITES, 87
KEY_ALAUNCH_APP1, 87
KEY_ALAUNCH_APP2, 87
KEY_ALAUNCH_MAIL, 87
KEY_ALT_L, 87
KEY_ALT_R, 87
KEY_AMEDIA_SELECT, 87
KEY_AMEDIA_STOP, 87
KEY_ANEXT_TRACK, 87
KEY_APLAY_PAUSE, 87
KEY_APPS, 87
KEY_APREV_TRACK, 87
KEY_ASEARCH, 88
KEY_AVOLUME_DOWN, 88
KEY_AVOLUME_MUTE, 88
KEY_AVOLUME_UP, 88
KEY_B1, 88
KEY_B2, 88
KEY_B3, 88
KEY_BACKSPACE, 88
KEY_BEG, 88
KEY_BREAK, 88
KEY_BROWSER_BACK, 88
KEY_BROWSER_FWD, 88
KEY_BROWSER_HOME, 89
KEY_BROWSER_REF, 89
KEY_BROWSER_STOP, 89
KEY_BTAB, 89

KEY_C1, [89](#)
KEY_C2, [89](#)
KEY_C3, [89](#)
KEY_CANCEL, [89](#)
KEY_CATAB, [89](#)
KEY_CBROWSER_BACK, [89](#)
KEY_CBROWSER_FWD, [89](#)
KEY_CBROWSER_HOME, [89](#)
KEY_CBROWSER_REF, [90](#)
KEY_CBROWSER_STOP, [90](#)
KEY_CFAVORITES, [90](#)
KEY_CLAUNCH_APP1, [90](#)
KEY_CLAUNCH_APP2, [90](#)
KEY_CLAUNCH_MAIL, [90](#)
KEY_CLEAR, [90](#)
KEY_CLOSE, [90](#)
KEY_CMEDIA_SELECT, [90](#)
KEY_CMEDIA_STOP, [90](#)
KEY_CNEXT_TRACK, [90](#)
KEY_CODE_YES, [90](#)
KEY_COMMAND, [91](#)
KEY_CONTROL_L, [91](#)
KEY_CONTROL_R, [91](#)
KEY_COPY, [91](#)
KEY_CPLAY_PAUSE, [91](#)
KEY_CPREV_TRACK, [91](#)
KEY_CREATE, [91](#)
KEY_CSEARCH, [91](#)
KEY_CTAB, [91](#)
KEY_CVOLUME_DOWN, [91](#)
KEY_CVOLUME_MUTE, [91](#)
KEY_CVOLUME_UP, [91](#)
KEY_DC, [92](#)
KEY_DL, [92](#)
KEY_DOWN, [92](#)
KEY_EIC, [92](#)
KEY_END, [92](#)
KEY_ENTER, [92](#)
KEY_EOL, [92](#)
KEY_EOS, [92](#)
KEY_EXIT, [92](#)
KEY_F, [92](#)
KEY_F0, [92](#)
KEY_FAVORITES, [92](#)
KEY_FIND, [93](#)
KEY_HELP, [93](#)
KEY_HOME, [93](#)
KEY_IC, [93](#)
KEY_IL, [93](#)
KEY_LAUNCH_APP1, [93](#)
KEY_LAUNCH_APP2, [93](#)
KEY_LAUNCH_MAIL, [93](#)
KEY_LEFT, [93](#)
KEY_LHELP, [93](#)
KEY_LL, [93](#)
KEY_MARK, [93](#)
KEY_MAX, [94](#)
KEY_MEDIA_SELECT, [94](#)
KEY_MEDIA_STOP, [94](#)
KEY_MESSAGE, [94](#)
KEY_MIN, [94](#)
KEY_MOUSE, [94](#)
KEY_MOVE, [94](#)
KEY_NEXT, [94](#)
KEY_NEXT_TRACK, [94](#)
KEY_NPAGE, [94](#)
KEY_OFFSET, [94](#)
KEY_OPEN, [94](#)
KEY_OPTIONS, [95](#)
KEY_PAUSE, [95](#)
KEY_PLAY_PAUSE, [95](#)
KEY_PPAGE, [95](#)
KEY_PREV_TRACK, [95](#)
KEY_PREVIOUS, [95](#)
KEY_PRINT, [95](#)
KEY_PRINTSCREEN, [95](#)
KEY_REDO, [95](#)
KEY_REFERENCE, [95](#)
KEY_REFRESH, [95](#)
KEY_REPLACE, [95](#)
KEY_RESET, [96](#)
KEY_RESIZE, [96](#)
KEY_RESTART, [96](#)
KEY_RESUME, [96](#)
KEY_RIGHT, [96](#)
KEY_SAPPS, [96](#)
KEY_SAVE, [96](#)
KEY_SBEG, [96](#)
KEY_SBROWSER_BACK, [96](#)
KEY_SBROWSER_FWD, [96](#)
KEY_SBROWSER_HOME, [96](#)
KEY_SBROWSER_REF, [96](#)
KEY_SBROWSER_STOP, [97](#)
KEY_SCANCEL, [97](#)
KEY_SCOMMAND, [97](#)
KEY_SCOPY, [97](#)
KEY_SCREATE, [97](#)
KEY_SCROLLLOCK, [97](#)
KEY_SDC, [97](#)
KEY_SDL, [97](#)
KEY_SDOWN, [97](#)
KEY_SEARCH, [97](#)
KEY_SELECT, [97](#)
KEY_SEND, [97](#)
KEY_SEOL, [98](#)
KEY_SEXIT, [98](#)
KEY_SF, [98](#)
KEY_SFAVORITES, [98](#)
KEY_SFIND, [98](#)
KEY_SHELP, [98](#)
KEY_SHIFT_L, [98](#)
KEY_SHIFT_R, [98](#)
KEY_SHOME, [98](#)
KEY_SIC, [98](#)
KEY_SLAUNCH_APP1, [98](#)
KEY_SLAUNCH_APP2, [98](#)

KEY_SLAUNCH_MAIL, 99
KEY_SLEFT, 99
KEY_SMEDIA_SELECT, 99
KEY_SMEDIA_STOP, 99
KEY_SMESSAGE, 99
KEY_SMOVE, 99
KEY_SNEXT, 99
KEY_SNEXT_TRACK, 99
KEY_SOPTIONS, 99
KEY_SPAUSE, 99
KEY_SPLAY_PAUSE, 99
KEY_SPREV_TRACK, 99
KEY_SPREVIOUS, 100
KEY_SPRINT, 100
KEY_SR, 100
KEY_SREDO, 100
KEY_SREPLACE, 100
KEY_SRESET, 100
KEY_SRIGHT, 100
KEY_SRSUME, 100
KEY_SSAVE, 100
KEY_SSEARCH, 100
KEY_SSUSPEND, 100
KEY_STAB, 100
KEY_SUNDO, 101
KEY_SUP, 101
KEY_SUSPEND, 101
KEY_SVOLUME_DOWN, 101
KEY_SVOLUME_MUTE, 101
KEY_SVOLUME_UP, 101
KEY_UNDO, 101
KEY_UP, 101
KEY_VOLUME_DOWN, 101
KEY_VOLUME_MUTE, 101
KEY_VOLUME_UP, 101
keyname, 115
keypad, 115
killchar, 115
LINES, 127
leaveok, 115
longname, 115
MOUSE_MOVED, 101
MOUSE_POS_REPORT, 102
MOUSE_WHEEL_DOWN, 102
MOUSE_WHEEL_LEFT, 102
MOUSE_WHEEL_RIGHT, 102
MOUSE_WHEEL_SCROLL, 102
MOUSE_WHEEL_UP, 102
MOUSE_X_POS, 102
MOUSE_Y_POS, 102
map_button, 115
meta, 115
mmask_t, 108
mouse_off, 115
mouse_on, 115
mouse_set, 115
Mouse_status, 127
mouse_trafo, 115
mouseinterval, 116
mousemask, 116
move, 116
mvaddch, 116
mvaddchnstr, 116
mvaddchstr, 116
mvaddnstr, 116
mvaddrawch, 116
mvaddstr, 116
mvchgat, 116
mvcur, 116
mvdelch, 116
mvdeleteln, 116
mvderwin, 116
mvgetch, 116
mvgetnstr, 116
mvgetstr, 116
mvhline, 117
mvinch, 117
mvinchnstr, 117
mvinchstr, 117
mvinnstr, 117
mvinsch, 117
mvinsertln, 117
mvinsnstr, 117
mvinsrawch, 117
mvinsstr, 117
mvinstr, 117
mvprintw, 117
mvscanw, 117
mvvline, 117
mvwaddch, 117
mvwaddchnstr, 117
mvwaddchstr, 118
mvwaddnstr, 118
mvwaddrawch, 118
mvwaddstr, 118
mvwchgat, 118
mvwdelch, 118
mvwdeleteln, 118
mvwgetch, 118
mvwgetnstr, 118
mvwgetstr, 118
mvwhline, 118
mvwin, 118
mvwinch, 118
mvwinchnstr, 118
mvwinchstr, 118
mvwinnstr, 118
mvwinsch, 118
mvwinsetln, 118
mvwinsnstr, 118
mvwinsrawch, 118
mvwinsstr, 118
mvwinstr, 118
mvwprintw, 118
mvwscanw, 118
mvwvline, 118

NULL, 102
napms, 118
nc_getmouse, 118
newpad, 118
newterm, 119
newwin, 119
nl, 119
nocbreak, 119
nocrmode, 119
nodelay, 119
noecho, 119
nonl, 119
noqiflush, 119
noraw, 119
notimeout, 119
OK, 102
overlay, 119
overwrite, 119
PAD0, 102
PADENTER, 102
PADMINUS, 103
PADPLUS, 103
PADSLASH, 103
PADSTAR, 103
PADSTOP, 103
PAIR_NUMBER, 103
PDC_BUILD, 103
PDC_BUTTON_ALT, 103
PDC_BUTTON_CONTROL, 103
PDC_BUTTON_SHIFT, 103
PDC_CHARTEXT_BITS, 103
PDC_CLIP_ACCESS_ERROR, 103
PDC_CLIP_EMPTY, 104
PDC_CLIP_MEMORY_ERROR, 104
PDC_CLIP_SUCCESS, 104
PDC_COLOR_SHIFT, 104
PDC_KEY_MODIFIER_ALT, 104
PDC_KEY_MODIFIER_CONTROL, 104
PDC_KEY_MODIFIER_NUMLOCK, 104
PDC_KEY_MODIFIER_SHIFT, 104
PDC_MAX_MOUSE_BUTTONS, 104
PDC_MOUSE_MOVED, 104
PDC_MOUSE_POSITION, 104
PDC_MOUSE_WHEEL_DOWN, 104
PDC_MOUSE_WHEEL_LEFT, 105
PDC_MOUSE_WHEEL_RIGHT, 105
PDC_MOUSE_WHEEL_UP, 105
PDC_N_EXTENDED_MOUSE_BUTTONS, 105
PDC_clearclipboard, 119
PDC_debug, 119
PDC_freeclipboard, 119
PDC_get_input_fd, 119
PDC_get_key_modifiers, 119
PDC_getclipboard, 119
PDC_return_key_modifiers, 120
PDC_save_key_modifiers, 120
PDC_set_blink, 120
PDC_set_line_color, 120
PDC_set_resize_limits, 120
PDC_set_title, 120
PDC_setclipboard, 120
PDC_ungetch, 120
PDCEX, 105
PDCURSES, 105
pair_content, 119
pechochar, 120
pnoutrefresh, 120
prefresh, 120
printw, 120
putwin, 120
qiflush, 120
REPORT_MOUSE_POSITION, 105
raw, 120
raw_output, 121
redrawwin, 121
refresh, 121
request_mouse_pos, 121
reset_prog_mode, 121
reset_shell_mode, 121
resetterm, 121
resetty, 121
resize_term, 121
resize_window, 121
ripline, 121
SHF_DC, 105
SHF_DOWN, 105
SHF_IC, 105
SHF_PADENTER, 105
SHF_PADMINUS, 105
SHF_PADPLUS, 106
SHF_PADSLASH, 106
SHF_PADSTAR, 106
SHF_UP, 106
SP, 127
SYSVcurses, 106
saveterm, 121
savetty, 121
scanw, 121
scr_dump, 122
scr_init, 122
scr_restore, 122
scr_set, 122
scr1, 122
scroll, 122
scrollok, 122
set_term, 122
setscrreg, 122
setsyx, 122
slk_attr_off, 122
slk_attr_on, 122
slk_attr_set, 122
slk_attrtff, 122
slk_attron, 122
slk_attrset, 122
slk_clear, 122
slk_color, 122

slk_init, 122
slk_label, 122
slk_noutrefresh, 122
slk_refresh, 122
slk_restore, 123
slk_set, 123
slk_touch, 123
standend, 123
standout, 123
start_color, 123
stdscr, 127
subpad, 123
subwin, 123
syncok, 123
TABSIZ, 127
TRUE, 106
term_attrs, 123
termattrs, 123
termname, 123
timeout, 123
touchline, 123
touchwin, 123
traceoff, 123
traceon, 123
ttytype, 127
typeahead, 123
unctrl, 123
ungetch, 106
ungetmouse, 123
untouchwin, 123
use_default_colors, 124
use_env, 124
vid_attr, 124
vid_puts, 124
vidattr, 124
vidputs, 124
vline, 124
vw_printw, 124
vw_scanw, 124
vwprintw, 124
vwscanw, 124
WA_ALTCHARSET, 106
WA_ATTRIBUTES, 106
WA_BLINK, 106
WA_BOLD, 106
WA_DIM, 106
WA_HORIZONTAL, 107
WA_INVIS, 107
WA_LEFT, 107
WA_LOW, 107
WA_NORMAL, 107
WA_PROTECT, 107
WA_REVERSE, 107
WA_RIGHT, 107
WA_STANDOUT, 107
WA_TOP, 107
WA_UNDERLINE, 107
WA_VERTICAL, 107
WHEEL_SCROLLED, 108
WINDOW, 108
waddch, 124
waddchnstr, 124
waddchstr, 124
waddnstr, 124
waddrawch, 124
waddstr, 124
wattr_get, 124
wattr_off, 124
wattr_on, 124
wattr_set, 124
wattroff, 124
watron, 124
wattrset, 124
wbkgd, 124
wbkgdset, 124
wborder, 124
wchgat, 124
wclear, 125
wclrtobot, 125
wclrtoeol, 125
wcolor_set, 125
wcursyncup, 125
wdelch, 125
wdeleteln, 125
wechochar, 125
wenclose, 125
werase, 125
wgetch, 125
wgetnstr, 125
wgetstr, 125
whline, 125
winch, 125
winchnstr, 125
winchstr, 125
winnstr, 125
winsch, 125
winsdelln, 125
winsertln, 125
winsnstr, 125
winsrawch, 125
winsstr, 125
winstr, 125
wmouse_position, 125
wmouse_trafo, 125
wmove, 125
wnoutrefresh, 126
wordchar, 126
wprintw, 126
wredrawln, 126
wrefresh, 126
wresize, 126
wscanw, 126
wscl, 126
wsetscreg, 126
wstandend, 126
wstandout, 126

- wsyncdown, 126
- wsyncup, 126
- wtimeout, 126
- wtouchln, 126
- wvline, 126
- XOPEN, 108
- Xinitscr, 126
- newpad
 - ncurses.h, 118
- newterm
 - ncurses.h, 119
- newwin
 - ncurses.h, 119
- nl
 - ncurses.h, 119
- nocbreak
 - ncurses.h, 119
- nocrmode
 - ncurses.h, 119
- nodelay
 - ncurses.h, 119
- noecho
 - ncurses.h, 119
- nonl
 - ncurses.h, 119
- noqiflush
 - ncurses.h, 119
- noraw
 - ncurses.h, 119
- notimeout
 - ncurses.h, 119
- OK
 - ncurses.h, 102
- orig_attr
 - SCREEN, 14
- orig_back
 - SCREEN, 15
- orig_cursor
 - SCREEN, 15
- orig_fore
 - SCREEN, 15
- overlay
 - ncurses.h, 119
- overwrite
 - ncurses.h, 119
- P_RANDNUM
 - 2048.c, 46
- PAD0
 - ncurses.h, 102
- PADENTER
 - ncurses.h, 102
- PADMINUS
 - ncurses.h, 103
- PADPLUS
 - ncurses.h, 103
- PADSLASH
 - ncurses.h, 103
- PADSTAR
 - ncurses.h, 103
- PADSTOP
 - ncurses.h, 103
- PAIR_NUMBER
 - ncurses.h, 103
- PDC_BUILD
 - ncurses.h, 103
- PDC_BUTTON_ALT
 - ncurses.h, 103
- PDC_BUTTON_CONTROL
 - ncurses.h, 103
- PDC_BUTTON_SHIFT
 - ncurses.h, 103
- PDC_CHARTEXT_BITS
 - ncurses.h, 103
- PDC_CLIP_ACCESS_ERROR
 - ncurses.h, 103
- PDC_CLIP_EMPTY
 - ncurses.h, 104
- PDC_CLIP_MEMORY_ERROR
 - ncurses.h, 104
- PDC_CLIP_SUCCESS
 - ncurses.h, 104
- PDC_COLOR_SHIFT
 - ncurses.h, 104
- PDC_KEY_MODIFIER_ALT
 - ncurses.h, 104
- PDC_KEY_MODIFIER_CONTROL
 - ncurses.h, 104
- PDC_KEY_MODIFIER_NUMLOCK
 - ncurses.h, 104
- PDC_KEY_MODIFIER_SHIFT
 - ncurses.h, 104
- PDC_MAX_MOUSE_BUTTONS
 - ncurses.h, 104
- PDC_MOUSE_MOVED
 - ncurses.h, 104
- PDC_MOUSE_POSITION
 - ncurses.h, 104
- PDC_MOUSE_WHEEL_DOWN
 - ncurses.h, 104
- PDC_MOUSE_WHEEL_LEFT
 - ncurses.h, 105
- PDC_MOUSE_WHEEL_RIGHT
 - ncurses.h, 105
- PDC_MOUSE_WHEEL_UP
 - ncurses.h, 105
- PDC_N_EXTENDED_MOUSE_BUTTONS
 - ncurses.h, 105
- PDC_clearclipboard
 - ncurses.h, 119
- PDC_debug
 - ncurses.h, 119
- PDC_freeclipboard
 - ncurses.h, 119
- PDC_get_input_fd
 - ncurses.h, 119

PDC_get_key_modifiers
 ncurses.h, 119
 PDC_getclipboard
 ncurses.h, 119
 PDC_return_key_modifiers
 ncurses.h, 120
 PDC_save_key_modifiers
 ncurses.h, 120
 PDC_set_blink
 ncurses.h, 120
 PDC_set_line_color
 ncurses.h, 120
 PDC_set_resize_limits
 ncurses.h, 120
 PDC_set_title
 ncurses.h, 120
 PDC_setclipboard
 ncurses.h, 120
 PDC_ungetch
 ncurses.h, 120
 PDCEX
 ncurses.h, 105
 PDCURSES
 ncurses.h, 105
 PWD
 2048.c, 20
 PWD_LEN
 2048.c, 20
 pair_content
 ncurses.h, 119
 pechochar
 ncurses.h, 120
 play
 2048.c, 40
 pnoutrefresh
 ncurses.h, 120
 point
 2048.c, 46
 prefresh
 ncurses.h, 120
 printw
 ncurses.h, 120
 putwin
 ncurses.h, 120

 qiflush
 ncurses.h, 120

 README.md, 127
 REPORT_MOUSE_POSITION
 ncurses.h, 105
 Rando
 2048.c, 42
 raw
 ncurses.h, 120
 raw_inp
 SCREEN, 15
 raw_out
 SCREEN, 15

 raw_output
 ncurses.h, 121
 redrawwin
 ncurses.h, 121
 refresh
 ncurses.h, 121
 request_mouse_pos
 ncurses.h, 121
 reset_prog_mode
 ncurses.h, 121
 reset_shell_mode
 ncurses.h, 121
 resetterm
 ncurses.h, 121
 resetty
 ncurses.h, 121
 resize_term
 ncurses.h, 121
 resize_window
 ncurses.h, 121
 resized
 SCREEN, 15
 return_key_modifiers
 SCREEN, 15
 ripoffline
 ncurses.h, 121
 row
 2048.c, 47

 SCREEN, 12
 _map_mbe_to_key, 13
 _preserve, 13
 _restore, 13
 _trap_mbe, 13
 alive, 13
 audible, 13
 autocr, 13
 cbreak, 13
 cols, 13
 curscol, 14
 cursrow, 14
 delaytenths, 14
 echo, 14
 key_code, 14
 line_color, 14
 lines, 14
 linesrippedoff, 14
 linesrippedoffontop, 14
 mono, 14
 mouse_wait, 14
 orig_attr, 14
 orig_back, 15
 orig_cursor, 15
 orig_fore, 15
 raw_inp, 15
 raw_out, 15
 resized, 15
 return_key_modifiers, 15
 save_key_modifiers, 15

- slk_winptr, 15
- slklines, 15
- visibility, 15
- SHF_DC
 - ncurses.h, 105
- SHF_DOWN
 - ncurses.h, 105
- SHF_IC
 - ncurses.h, 105
- SHF_PADENTER
 - ncurses.h, 105
- SHF_PADMINUS
 - ncurses.h, 105
- SHF_PADPLUS
 - ncurses.h, 106
- SHF_PADSLASH
 - ncurses.h, 106
- SHF_PADSTAR
 - ncurses.h, 106
- SHF_UP
 - ncurses.h, 106
- SP
 - ncurses.h, 127
- SYSVcurses
 - ncurses.h, 106
- save_key_modifiers
 - SCREEN, 15
- saveterm
 - ncurses.h, 121
- savetty
 - ncurses.h, 121
- scanw
 - ncurses.h, 121
- scr_dump
 - ncurses.h, 122
- scr_init
 - ncurses.h, 122
- scr_restore
 - ncurses.h, 122
- scr_set
 - ncurses.h, 122
- scr1
 - ncurses.h, 122
- scroll
 - ncurses.h, 122
- scrollok
 - ncurses.h, 122
- set_term
 - ncurses.h, 122
- setscreg
 - ncurses.h, 122
- setsyx
 - ncurses.h, 122
- settings
 - 2048.c, 42
- showBoard
 - 2048.c, 43
- slk_attr_off
 - ncurses.h, 122
- slk_attr_on
 - ncurses.h, 122
- slk_attr_set
 - ncurses.h, 122
- slk_attroff
 - ncurses.h, 122
- slk_attron
 - ncurses.h, 122
- slk_attrset
 - ncurses.h, 122
- slk_clear
 - ncurses.h, 122
- slk_color
 - ncurses.h, 122
- slk_init
 - ncurses.h, 122
- slk_label
 - ncurses.h, 122
- slk_noutrefresh
 - ncurses.h, 122
- slk_refresh
 - ncurses.h, 122
- slk_restore
 - ncurses.h, 123
- slk_set
 - ncurses.h, 123
- slk_touch
 - ncurses.h, 123
- slk_winptr
 - SCREEN, 15
- slklines
 - SCREEN, 15
- standend
 - ncurses.h, 123
- standout
 - ncurses.h, 123
- start_color
 - ncurses.h, 123
- stdscr
 - ncurses.h, 127
- subpad
 - ncurses.h, 123
- subwin
 - ncurses.h, 123
- syncok
 - ncurses.h, 123
- TABSIZE
 - ncurses.h, 127
- TRUE
 - ncurses.h, 106
- term_attrs
 - ncurses.h, 123
- termattrs
 - ncurses.h, 123
- termname
 - ncurses.h, 123
- timeout

- ncurses.h, [123](#)
- touchline
 - ncurses.h, [123](#)
- touchwin
 - ncurses.h, [123](#)
- traceoff
 - ncurses.h, [123](#)
- traceon
 - ncurses.h, [123](#)
- ttytype
 - ncurses.h, [127](#)
- typeahead
 - ncurses.h, [123](#)
- unctrl
 - ncurses.h, [123](#)
- ungetch
 - ncurses.h, [106](#)
- ungetmouse
 - ncurses.h, [123](#)
- untouchwin
 - ncurses.h, [123](#)
- use_default_colors
 - ncurses.h, [124](#)
- use_env
 - ncurses.h, [124](#)
- vid_attr
 - ncurses.h, [124](#)
- vid_puts
 - ncurses.h, [124](#)
- vidattr
 - ncurses.h, [124](#)
- vidputs
 - ncurses.h, [124](#)
- visibility
 - SCREEN, [15](#)
- vline
 - ncurses.h, [124](#)
- vw_printw
 - ncurses.h, [124](#)
- vw_scanw
 - ncurses.h, [124](#)
- vwprintw
 - ncurses.h, [124](#)
- vwscanw
 - ncurses.h, [124](#)
- WA_ALTCHARSET
 - ncurses.h, [106](#)
- WA_ATTRIBUTES
 - ncurses.h, [106](#)
- WA_BLINK
 - ncurses.h, [106](#)
- WA_BOLD
 - ncurses.h, [106](#)
- WA_DIM
 - ncurses.h, [106](#)
- WA_HORIZONTAL
 - ncurses.h, [107](#)
- WA_INVIS
 - ncurses.h, [107](#)
- WA_LEFT
 - ncurses.h, [107](#)
- WA_LOW
 - ncurses.h, [107](#)
- WA_NORMAL
 - ncurses.h, [107](#)
- WA_PROTECT
 - ncurses.h, [107](#)
- WA_REVERSE
 - ncurses.h, [107](#)
- WA_RIGHT
 - ncurses.h, [107](#)
- WA_STANDOUT
 - ncurses.h, [107](#)
- WA_TOP
 - ncurses.h, [107](#)
- WA_UNDERLINE
 - ncurses.h, [107](#)
- WA_VERTICAL
 - ncurses.h, [107](#)
- WARNING_POSITION_X
 - 2048.c, [20](#)
- WARNING_POSITION_Y
 - 2048.c, [20](#)
- WHEEL_SCROLLED
 - ncurses.h, [108](#)
- WINDOW
 - ncurses.h, [108](#)
- waddch
 - ncurses.h, [124](#)
- waddchnstr
 - ncurses.h, [124](#)
- waddchstr
 - ncurses.h, [124](#)
- waddnstr
 - ncurses.h, [124](#)
- waddrawch
 - ncurses.h, [124](#)
- waddstr
 - ncurses.h, [124](#)
- wattr_get
 - ncurses.h, [124](#)
- wattr_off
 - ncurses.h, [124](#)
- wattr_on
 - ncurses.h, [124](#)
- wattr_set
 - ncurses.h, [124](#)
- wattroff
 - ncurses.h, [124](#)
- wattron
 - ncurses.h, [124](#)
- wattrset
 - ncurses.h, [124](#)
- wbkgd

- ncurses.h, 124
- wbkgdset
 - ncurses.h, 124
- wborder
 - ncurses.h, 124
- wchgat
 - ncurses.h, 124
- wclear
 - ncurses.h, 125
- wclrtoobot
 - ncurses.h, 125
- wclrtoeol
 - ncurses.h, 125
- wcolor_set
 - ncurses.h, 125
- wcursyncup
 - ncurses.h, 125
- wdelch
 - ncurses.h, 125
- wdeleteln
 - ncurses.h, 125
- wechochar
 - ncurses.h, 125
- welcome
 - 2048.c, 44
- wenclose
 - ncurses.h, 125
- werase
 - ncurses.h, 125
- wgetch
 - ncurses.h, 125
- wgetnstr
 - ncurses.h, 125
- wgetstr
 - ncurses.h, 125
- whline
 - ncurses.h, 125
- winch
 - ncurses.h, 125
- winchnstr
 - ncurses.h, 125
- winchstr
 - ncurses.h, 125
- winnstr
 - ncurses.h, 125
- winsch
 - ncurses.h, 125
- winsdelln
 - ncurses.h, 125
- winsertln
 - ncurses.h, 125
- winsnstr
 - ncurses.h, 125
- winsrawch
 - ncurses.h, 125
- winsstr
 - ncurses.h, 125
- winstr
 - ncurses.h, 125
- wmouse_position
 - ncurses.h, 125
- wmouse_trafo
 - ncurses.h, 125
- wmove
 - ncurses.h, 125
- wnoutrefresh
 - ncurses.h, 126
- wordchar
 - ncurses.h, 126
- wprintw
 - ncurses.h, 126
- wredrawln
 - ncurses.h, 126
- wrefresh
 - ncurses.h, 126
- wresize
 - ncurses.h, 126
- wscanw
 - ncurses.h, 126
- wscrl
 - ncurses.h, 126
- wsetscreg
 - ncurses.h, 126
- wstandend
 - ncurses.h, 126
- wstandout
 - ncurses.h, 126
- wsyncdown
 - ncurses.h, 126
- wsyncup
 - ncurses.h, 126
- wtimeout
 - ncurses.h, 126
- wtouchln
 - ncurses.h, 126
- wvline
 - ncurses.h, 126
- x
 - MEVENT, 10
 - MOUSE_STATUS, 11
- XOPEN
 - ncurses.h, 108
- xbutton
 - MOUSE_STATUS, 11
- Xinitscr
 - ncurses.h, 126
- y
 - MEVENT, 10
 - MOUSE_STATUS, 11
- z
 - MEVENT, 11