Returns A pointer to the string associated with the matching name. Returns a null pointer if no match is found. *26.2*

---

gets *Read String* `<stdio.h>`

```
char *gets(char *s);
```

Reads characters from the `stdin` stream and stores them in the array pointed to by `s`. Reading stops at the first new-line character (which is discarded) or at end-of-file. `gets` appends a null character to the string.

Returns `s` (a pointer to the array in which the input is stored). Returns a null pointer if a read error occurs or `gets` encounters the end of the stream before it has stored any characters. *13.3, 22.5*

---

getwc *Read Wide Character from File (C99)* `<wchar.h>`

```
wint_t getwc(FILE *stream);
```

Wide-character version of `getc`. *25.5*

---

getwchar *Read Wide Character (C99)* `<wchar.h>`

```
wint_t getwchar(void);
```

Wide-character version of `getchar`. *25.5*

---

gmtime *Convert Calendar Time to Broken-Down UTC Time* `<time.h>`

```
struct tm *gmtime(const time_t *timer);
```

Returns A pointer to a structure containing a broken-down UTC time equivalent to the calendar time pointed to by `timer`. Returns a null pointer if the calendar time can't be converted to UTC. *26.3*

---

hypot *Hypotenuse (C99)* `<math.h>`

```
double hypot(double x, double y);
hypotf  float hypotf(float x, float y);
hypotl  long double hypotl(long double x, long double y);
```

Returns $\sqrt{x^2 + y^2}$ (the hypotenuse of a right triangle with legs `x` and `y`). A range error may occur. *23.4*

---

ilogb *Unbiased Exponent (C99)* `<math.h>`

```
int ilogb(double x);
ilogbf  int ilogbf(float x);
ilogbl  int ilogbl(long double x);
```

Returns Exponent of `x` as a signed integer; equivalent to calling the corresponding `logb` function and casting the returned value to type `int`. Returns `FP_ILOGB0` if `x` is zero, `INT_MAX` if `x` is infinite, and `FP_ILOGBNAN` if `x` is a NaN; a domain error or range error may occur in these cases. *23.4*

---

imaxabs *Greatest-Width Integer Absolute Value (C99)* `<inttypes.h>`

```
intmax_t imaxabs(intmax_t j);
```