Simple Numeric Access

Proposed Wording

Add the following to [lib.string.classes] at the end of the "Header <string> synopsis":

```
int stoi(string& str, int base = 10);
long stol(string& str, int base = 10);
unsigned long stoul(string& str, int base = 10);
long long stoll(string& str, int base = 10);
unsigned long long stoull(string& str, int base = 10);
float stof(string& str);
double stod(string& str);
long double stold(string& str);
string to_string(long long val);
string to_string(unsigned long long val);
string to_string(long double val);
```

Add the following as a new subclause 21.3.7.10, "Numeric conversions":

```
int stoi(string& str, int base = 10);
long stol(string& str, int base = 10);
unsigned long stoul(string& str, int base = 10);
long long stoll(string& str, int base = 10);
unsigned long long stoull(string& str, int base = 10);
```

Effects: the first two functions call `strtol(str.c_str(), 0, base)`, and the last three functions call `strtoul(str.c_str(), 0, base)`, `strtoll(str.c_str(), 0, base)`, and `strtoull(str.c_str(), 0, base)`, respectively. Each returns the converted result, if any, and erases the characters from the front of `str` that were converted to get the result.

Returns: the converted result.

Throws: `invalid_argument` if `strol`, `strtoul`, `strtoll`, or `strtoull` reports that no conversion could be performed. Throws `out_of_range` if the converted value is outside the range of representable values for the return type.

```
float stof(string& str);
double stod(string& str);
long double stold(string& str);
```

Effects: the functions call `strtof(str.c_str(), 0)`, `strtod(str.c_str(), 0)`, and `strtold(str.c_str(), 0)`, respectively, and return the converted result, if any. They erase the characters from the front of `str` that were converted to get the result.

Returns: the converted result.

Throws: `invalid_argument` if `strof`, `strtod`, or `strtold` reports that no conversion could be performed. Throws `out_of_range` if `strtof`, `strtod`, or `strtold` sets `errno` to `ERANGE`.

```
string to_string(long long val);
string to_string(unsigned long long val);
string to_string(long double val);
```

Returns: each function returns a `string` object holding the character representation of the value of its argument that would be generated by calling `sprintf(buf, fmt, val)` with a format specifier of `"%lld"`, `"%ulld"`, or `"%f"`, respectively.

Throws: nothing.