Attempted to pass an invalid type of variable into a variadic function.

Erroneous code example:

```
# use std::os::raw::{c_char, c_int};
extern "C" {
    fn printf(format: *const c_char, ...) -> c_int;
}

unsafe {
    printf("%f\n\0".as_ptr() as _, 0f32);
    // error: cannot pass an `f32` to variadic function, cast to `c_double`
}
```

Certain Rust types must be cast before passing them to a variadic function, because of arcane ABI rules dictated by the C standard. To fix the error, cast the value to the type specified by the error message (which you may need to import from `std::os::raw`).

In this case, `c_double` has the same size as `f64` so we can use it directly:

```
# use std::os::raw::{c_char, c_int};
# extern "C" {
#     fn printf(format: *const c_char, ...) -> c_int;
# }

unsafe {
    printf("%f\n\0".as_ptr() as _, 0f64); // ok!
}
```