第五章

5.19

```
void *advanced_memset(void *s, int c, size_t n)
{
  size_t k = sizeof(unsigned long);
  unsigned char *schar = s;
  unsigned long mask = (~((1UL<<k) - 1));
  unsigned long *slong_s = (schar + k - 1) & mask;
  unsigned long *slong_e = (schar + n) & mask;
  unsigned char *schar_s = (unsigned char*) slong_s;
  unsigned char *schar_e = (unsigned char*) slong_e;
  unsigned char *schar_end = schar + n;

  unsigned char cc = (unsigned char) c;
  unsigned long val = 0 | cc;

  if (schar_end < schar_e) {
    while (schar < schar_end)
      *schar++ = cc;

    return s;
  }

  for (size_t i = 1; i < k; i *= 2)
    val |= val << (i * 8);

  while (schar < schar_s)
    *schar++ = cc;

  while (slong_s < slong_e)
    *slong_s++ = val;

  while (schar_e < schar_end)
    *schar_e++ = cc;

  return s;
}
```

5.20

```
double poly(double a[], double x, int degree)
{
  long int i;
  double result = 0;
  double xpwr0 = 1;
  double xpwr1 = x;
  double xpwr2 = x*x;
  double xpwr3 = xpwr2 * x;
  double xpwr4 = xpwr3 * x;
```

```
  double x5 = xpwr4 * x;

  for (i = 0; i <= degree - 4; i += 5) {
    result += (a[i] * xpwr0 + a[i+1] * xpwr1 +\
               a[i+2] * xpwr2 + a[i+3] * xpwr3 +\
               a[i+4] * xpwr4);
    xpwr0 *= x5;
    xpwr1 *= x5;
    xpwr2 *= x5;
    xpwr3 *= x5;
    xpwr4 *= x5;
  }

  for (; i <= degree; ++i) {
    result += a[i] * xpwr0;
    xpwr0 *= x;
  }

  return result;
}

double poly2(double a[], double x, int degree)
{
  long i;
  double ret = 0;
  double x2 = x * x;
  double x3 = x2 * x;
  double x4 = x3 * x;
  double x5 = x4 * x;
  double x6 = x5 * x;
  double x7 = x6 * x;
  double x8 = x7 * x;

  for (i = degree; i >= 7; i -= 8) {
    ret = ret * x8 + (a[i] * x7 + a[i-1] * x6 +\
                      a[i-2] * x5 + a[i-3] * x4 +\
                      a[i-4] * x3 + a[i-5] * x2 +\
                      a[i-6] * x + a[i-7]);
  }

  for (; i >= 0; --i) {
    ret = ret * x + a[i];
  }

  return ret;
 }
```