61. Let $B = \left\{(x, y) \in \mathbb{R}^2 \mid \|(x, y) - (\frac{1}{2}, \frac{1}{2})\|_2 \leq \frac{1}{2}\right\}$.
   (a) Geometrically, what is B?
   (b) What is the area of B (using only elementary geometry)?
   (c) What follows for the integral $\int_B 1\ d(x, y) = \int_{I_2} \chi_B(x, y)\ d(x, y)$ (where $I_2 = [0, 1] \times [0, 1]$, the unit square)? (Note that $B \subseteq I_2$.)

   In the following exercises, we will consider several ways to compute this integral numerically.

62. (a) Write a program in MATLAB or Octave which takes as input parameters:
   - a function $f : [a, b] \subseteq \mathbb{R} \to \mathbb{R}$ (to approximate the integral of),
   - boundaries $a, b \in \mathbb{R}$ and
   - an even number $n \geq 2$ of subdivisions (such that the step length is $h = \frac{b-a}{n}$)

   and approximates the (one-dimensional) integral $\int_a^b f(x)\ dx$ using **Simpson's rule** (i.e. the Newton-Cotes formula of degree 2).

   (b) Green's theorem states that if, for a region $B \subseteq \mathbb{R}^2$, $(x, y) = (f(t), g(t))$ for $t \in [a, b]$ parametrizes ∂B, the boundary curve of B, then

$$\int_B d(x, y) = \int_a^b f(t)\ g'(t)\ dt.$$

   Motivate geometrically why the boundary curve ∂B of the B in example 61 can be parametrized by $f(t) = \frac{1+\cos t}{2}$, $g(t) = \frac{1+\sin t}{2}$, $t \in [0, 2\pi]$.

   (c) Show (using the above) that the integral in example 61 is equal to

$$\int_0^{2\pi} \frac{\cos t + \cos^2 t}{4}\ dt.$$

   (d) Compute the above integral using your program from part (a). (Use $n = 10$.)

63. (a) Write a program in MATLAB or Octave which takes as input parameters:
   - a dimension (an integer) $s \geq 1$,
   - a function $f : I_s \to \mathbb{R}$ (to approximate the integral of), where $I_s = [0, 1]^s$, and
   - a number of points (an integer) $n \geq 1$

   and approximates the s-dimensional integral $\int_{I_s} f(\vec{x})\ d\vec{x}$ using the **Monte Carlo method** for integration. Use the builtin `rand` function as your source of (pseudo)random numbers.

   (b) Compute the integral $\int_{I_2} \chi_B(x, y)\ d(x, y)$ from example 61 using your program from part (a). (Use $n = 10000$.) What happens if you run the program more than once? Why?
   Hint: Booleans in MATLAB and Octave are the numbers 0 and 1, so $\chi_{\{\alpha \leq \beta\}}$ can be written simply as $\alpha \leq \beta$, for any expressions α and β.

64. (a) Write a program in MATLAB or Octave which takes as input parameters:
    - a dimension (an integer) $s \geq 1$,
    - a function $f : I_s \to \mathbb{R}$ (to approximate the integral of), where $I_s = [0,1]^s$, and
    - a number of points (an integer) $n \geq 1$

    and approximates the s-dimensional integral $\int_{I_s} f(\vec{x})\ d\vec{x}$ using the **Quasi Monte Carlo method** for integration. For the quasirandom numbers, use the NIEDERREITER2 MATLAB implementation from `http://people.sc.fsu.edu/~jburkardt/m_src/niederreiter2/niederreiter2.html`. (You will have to download all the files.) A modified (debugged) version tested with Octave 3.4.3 and MATLAB 7.12.0 can be downloaded from `http://www.tigen.org/kevin.kofler/nummat2/niederreiter2.zip`.

    (b) Compute the integral $\int_{I_2} \chi_B(x,y)\ d(x,y)$ from example 61 using your program from part (a). (Use again $n = 10000$. This may take a few minutes.) What happens if you run this algorithm more than once? Why?

65. Let $B = \left\{\mathbf{x} = (x_1, \ldots, x_{10}) \in \mathbb{R}^{10} \mid \|\mathbf{x} - (\frac{1}{2}, \ldots, \frac{1}{2})\|_2 \leq \frac{1}{2}\right\}$. Compute (a numerical approximation of) the volume $\int_B 1\ d\mathbf{x} = \int_{I_{10}} \chi_B(\mathbf{x})\ d\mathbf{x}$ of B using
    (a) your Monte Carlo method from example 63 and
    (b) your Quasi Monte Carlo method from example 64.

    Which method converges more rapidly (i.e. with a smaller number of points n)?

66. (a) Write a program in MATLAB or Octave which takes as input parameters the integers:
    - $X_0 \geq 0$, the starting value,
    - $a \geq 0$, the multiplier,
    - $c \geq 0$, the increment and
    - m such that $m > X_0$, $m > a$ and $m > c$, the modulus,

    and implements a **linear congruence pseudorandom number generator**.

    (b) Write a program in MATLAB or Octave which takes as input parameters the integers:
    - X_0, a, c and m as above and
    - d such that $0 < d < m$, the coefficient of the quadratic term,

    and implements a **quadratic congruence pseudorandom number generator**.

    (c) Experiment with different parameters and check the quality of the pseudorandom numbers you get. Some useful tests:
    - Do the numbers “look” random?
    - What is the cycle length?
    - You can try some statistical tests on the pseudorandom numbers.