(a) Write a function that takes as input two Euclidean vectors $\vec{u} = \{u_1, u_2, ..., u_N\} \in \mathbb{R}^N$ and $\vec{v} = \{v_1, v_2, ..., v_N\} \in \mathbb{R}^N$ (of type `valarray<long double>`) and returns their inner product (also known as inner product or Hadamard product)

$$\vec{u} \cdot \vec{v} = \sum_{i=1}^{N} u_i v_i \tag{1}$$

as a `long double` number. Your function may use `(u*v).sum()` to compute the dot product of the `valarrays` `u` and `v`. Create a constant `valarray<long double>` equal to `u={0.1,0.1,...,0.1}` with $N = 10^6$ elements. Demonstrate that your program works by computing the dot product $\vec{u} \cdot \vec{u}$ for this constant vector. Display the difference $\vec{u} \cdot \vec{u} - 10^4$ on the screen.

(b) Repeat Question 2a using Kahan compensated summation to compute the sum.

(c) Write code for a function object that has a member variable m of type `int`, a suitable constructor, and a member function of the form
`double operator()(const valarray<double> u) const {`
which returns the weighted norm

$$\ell_m(\vec{v}) = \sqrt[m]{\sum_{i=0}^{N} |v_i|^m} \tag{2}$$

Use this function object to calculate the norm $\ell_2(\vec{u})$ for the vector in Question 2a. Does the quantity $\ell_2(\vec{u})^2$ equal the inner product $\vec{u} \cdot \vec{u}$ that you obtained above? [*Note*: half marks awarded if you use a regular function instead of a function object.] [