Doubles (cont)

- doubles can represent only a very small subset of the real numbers
- some real values cannot be represented exactly as a double
- arithmetic on doubles is approximate

Writing C Programs (cont)

Another problem to solve in C:

- precision check:
  - the expression

    1.0 - (0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1)

    should have the value 0
  - write a C program to check this

Admin

- no more C Lecture Stream after today
  - attend the A Stream lectures in CLB7
    (same timeslots … Tue 1-3, Wed 2-4)
  - given by Andrew Taylor, COMP1511 LiC
- VLab: don’t use the web one, use via VNC
- too many lab exercises each week?
  - just do as many as possible

Writing C Programs (cont)

Another problem to solve in C:

- pythagorean identity
  - geometry tells us that $\sin^2(t) + \cos^2(t) = 1.0$
  - write a C program to check this
  - read a value for *t*
  - check the identity (e.g. $1.0 - \sin^2(t) + \cos^2(t)$ is zero)
  - write out whether the identity holds

Recap

- **int** type for counters, indexes, …
  - read int values using e.g. scanf(“%d”, &x);
  - write int values using e.g. printf(“%5d”, x);
- **double** type for measurements, …
  - read double values using e.g. scanf(“%lf”, &y);
  - write int values using e.g. printf(“%6.2lf”, y);
- scanf() returns how many %*X* were *satisfied*

#define

- #define allows us to give meaningful names to expressions and constants
- usage: #define *Name* *Expression*
- effect:
  - everywhere *Name* appears in the program
  - *Name* is replaced by *Expression*