Design with Functions

Now that we have some notion of how functions work, let's look at how we can tackle a problem by dividing it into smaller pieces (*divide and conquer*)

Functional Decomposition

1. Divide the problem into independent sub-problems that are easier to solve.
2. Start with a *high-level* solution to the problem—many steps may be *abstract* (i.e., some of the implementation details remain unspecified)
3. Gradually refine solution into an concrete solution (i.e., algorithm) by adding detail.
4. a.k.a. *top-down design* and *stepwise refinement*

Example

Problem Statement

Write a program to find the total weight of a batch of washers. where every washer in the batch is identical.

Analysis

A washer is a doughnut shaped disc of metal of uniform thickness whose weight is a product of its volume and its density. The weight of a batch would simply be the weight of an individual washer times the number of washers in the batch.

Decomposition

1. Find the weight, w, of an individual washer
2. batch weight = n x w

Now we *refine* the first step

Find the weight, w, of an individual washer--->

1. Find the washer's area, a
2. Find the washer's volume, v = a x thickness, t.
3. weight: w = v x d (the density)

We should also refine the first step of this refined step

Find the washer's area, a-->

1. Find the area of the outer circle, A_o
2. Find the are of the inner circle (the doughnut hole) A_i
3. $a = A_o - A_i$

Seems like there's still another refinement

Find the area of a circle-->

1. area = pi x r^2

Note that each *refinement* is a candidate to be a *function*

It looks like we're done, but let's just think a minute

We've worked out the core problem, but what else should we do to make it a good program?

Since it's a complex program perhaps we should inform the user of what we're doing and gather all the data we need at once (why?).

This changes the top level steps to—

1. Print out an explanation of what the program is doing
2. Prompt and get values for n, t, d, and the inner and outer radii
3. Find the weight, w, of an individual washer
4. batch weight = n x w

Notice that we couldn't really do step 2 until we had worked out the core because we didn't know what data we needed *a priori*

Now let's go to code!

This page last updated on Monday, February 2, 2004