

CECS 174 – Lecture 17 – Structure Charts

Top-Down Design – is the step-wise process of breaking down problems into smaller tasks. Rather than diving straight into the code, plan out the components you will need to accomplish your task. This process will save you time and frustration.

The Steps of Top-Down Design:

1. Read the problem completely.
 - a. The better you understand the problem, the easier it is to solve it. Work through examples, draw pictures, list knowns and unknowns, analyze possible inputs and outputs and the connections between them. Plan out exactly how you want your program to work. Visualize the output.
2. Work out the problem manually with multiple sets of sample data.
 - a. Nothing can be automated that cannot be done manually. Working out the problem can help you further understand the problem and will make it easier to come up with the steps in order to solve it. Recognize exactly what steps you are applying and write them down so that you can easily adapt them into your program.
 - b. If you have trouble working out the problem as it is, try to represent the problem in another way. Figure out ways to simplify the problem, look for similarities to problems you have solved before, or break down the problem in to smaller tasks. Then come up with a general solution.
3. Break down and optimize the manual steps.
 - a. Break down tasks into smaller tasks using a structure chart. Keep breaking things down until you only have individual steps left.
 - b. Figure out what steps could be removed, combined, simplified, repeated, or generalized
4. Turn the optimized manual steps into code (which should be extremely easy if you have done a thorough job on the previous steps).
5. Test and optimize your code.
 - a. Make sure your program runs correctly. Test multiple inputs. Know what to expect for the results.
 - b. Is there any redundancy: Extra variables not being used? Values that could be stored rather than re-calculated? Chunks of code that could be made into a loop or method?
 - c. Is there a way to make the code cleaner and easier to read: Make readable variable names, add comments for blocks of code, fix spacing.

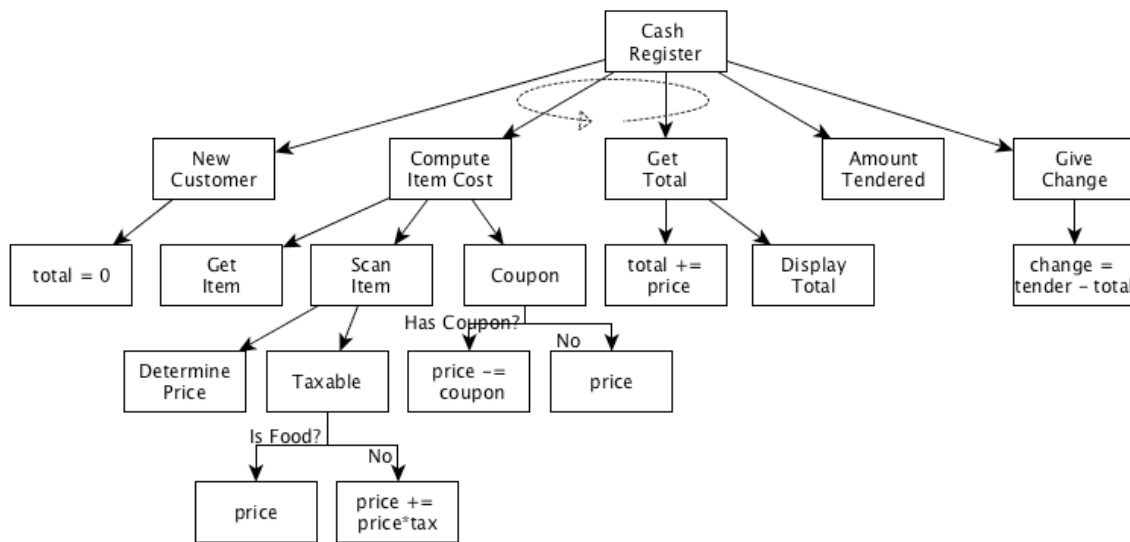
Structure Charts –

A structure chart is a diagram of the decomposition of a task. Each task is represented as one or more new tasks or steps at the next level down. This process is repeated for each new task until only indivisible steps remain. The resulting leaves of the tree now represent the final set of steps for your problem, which are read from left to right.

Representing Decisions or Repetition in a Structure Chart –

Decisions are represented by a line that diverges into two or more possibilities and labeled with the decision. Repetition is represented by a loop extending over any steps that are repeated, this loop may be labeled with the condition on which it repeats.

Examples – Cash Register at a grocery store.



Pseudocode –

```
Cash Register
total = 0
Repeat While More items
    Get item
    Get price
    If Is Taxable
        price += price * tax
    If Has Coupon
        price -= coupon
    total += price
    Display total
Get tendered
change = tendered - total
```

CECS 174 – Lecture 17 Worksheet -

Structure Charts –

Go through the process of Top-Down Design for this problem. Then create a structure chart and write out the resulting pseudocode for the following problem.

A delivery company wants you to write a program that will tell its workers which packages should be loaded on the trucks. Packages enter the delivery area on a conveyer belt. Each package that arrives is then loaded onto a truck. Each truck can only hold a maximum payload of 750 pounds. Each package can range in weight from 20 to 120 pounds. Each package is weighed before it is put on the truck. A package is loaded if the total weight loaded so far is less than or equal to 750 pounds. Assume that the packages are very small, but are quite heavy, so it is never a possibility that the truck will run out of space before it reaches its maximum payload. If a package being loaded puts the truck over its maximum payload, then the truck leaves and that package is loaded on the next truck. This process continues until there are no more packages to deliver for the day.