University of Warwick
Department of Computer Science

CS132: Computer Organisaton & Architecture

Coursework 2

2108182
Cem Yilmaz

December 30, 2021

Contents

1 Introduction

1.1 Problem 1

1.2 Problem 2

2 Problem 1

2.1 Power set

The first problem of coursework 2 introduces the implementation of a "Power set" in code. However, in order for this code to be written, it is first important to break down as to what requires to actually be defined.

Definition 2.1. Power Set

The power set of a finite set S, denoted as 2^S, is the set that contains all subsets of S as its elements. Formally,

$$2^S = \{X : X \subseteq S\}$$

The cardinality of the set (number of elements denoted as $|S|$), is then, as a corollary:

$$\begin{aligned} |2^S| &= 2^{|S|} \\ &= \sum_{i=0}^{|S|} {}^{|S|}C_k \end{aligned}$$

Note that for the sake of this paper, we will not be discussing if S is infinite, as the code will be implemented with the assumption that the input is also finite.

And the intuition behind this corollary is important for our implementation, as it in fact gives us a big hint as to how Problem 1 could be implemented as code. Consider the sets $S = \{x, y\}$, $S' = \{x, y, z\}$, 2^S and $2^{S'}$. In terms of decision trees for their power sets, it would be visualised as the following:

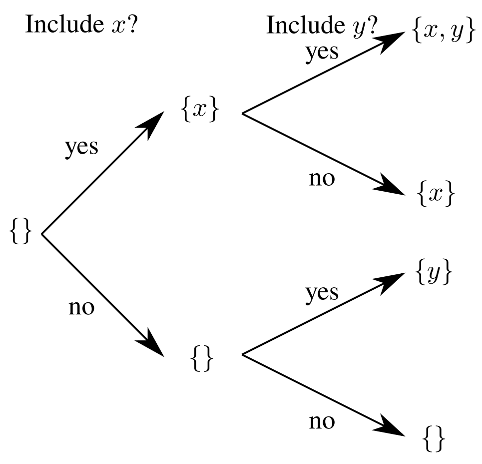

Figure 1: Visualisation of the power set 2^S

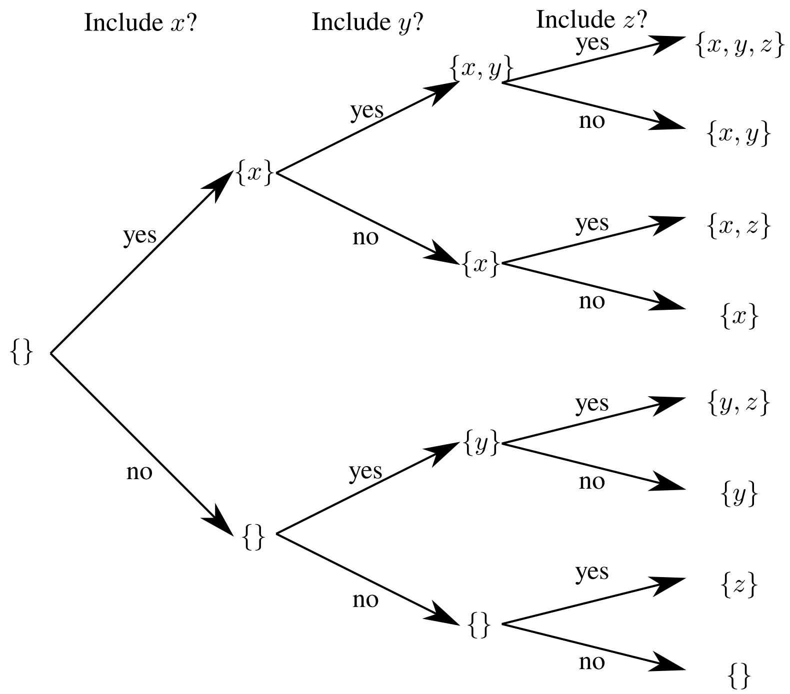

Figure 2: Visualisation of the power set $2^{S'}$

These figures make it apparent as to how the cardinality $|2^S| = 2^{|S|}$ precisely works. That is, for each new element to the set S, we must add a new branch of yes or no decisions for all previously existing power set elements. When the answer is no to the new element for all branches, we get 2^S. With all answers yes, we obtain $x \in 2^S : x \cup z$.
Formally, let us define $S' = S \cup \{\zeta\}$ where ζ is the new arbitrary element of a finite set S. Then, we can obtain the following formula for their power set:

$$2^{S^*} = \{x \in 2^S : x \cup \{\zeta\}\}$$
$$2^{S'} = 2^S \cup 2^{S^*}$$

2.2 Pseudocode

Now that the intuition and the mathematical formula has been well defined, we could write the pseudo code that we will implement into C. This will help us plan the structure of what is to come when writing our code.

Data: User input of elements into our set S
Result: A printed 2^S
Begin
Take input from user;
Initialise output variable array;
while *Our counter is not at the size of* $|S| + 1$ *elements* **do**
  Initialise counter = 1;
  if *first element* **then**
    Add first singleton set to output variable;
  else
    Add new element to each existing set in output variable and store them in output variable;
  end
  counter = counter+1;
end
Add empty set to output variable;
Print output variable;
End

Algorithm 1: Power set pseudocode

Now that the pseudocode has been defined, we could move onto writing the actual code and explaining each line's design.

2.3 Explanation of code design

This section will require that the file $powerset.c$ is opened as the lines of code that are expressed in this section correspond to the lines of code of that file.

- (Lines $7-15$)
- (Lines $29-33$)
- (Lines $37-40$)
- (Lines $62-83$)

2.4 Flowcharts

3 Problem 2