

I do not know if we get score or not on this work, but I am sure how to get full score if it takes place. That is to stick to the way of saying in the speech. Here is the rule from the speech, and I will keep on it. My work is under this blue box, so I wish for the one who read this not to think that there is no word to read in haste...

Library

Woman = more or less like a man, but not of the same sex

Person = a woman or a man – young or old

Machine = a thing that can do a task with no help, or not much help, from a person

<noun> :: = <noun that names one thing> "s"

| <noun that names one thing> "es" (two or more of a things, when we put end)

Names of Persons = []

Other = not the same

Other than = not the same as

<number> :: = 0 (nought)

| 1 + <number>

Many = more than two

Computer = a machine that can do at least what that machine can

Vocabulary = set of words

Language = vocabulary plus rules

Define = Tell what a word means

Program = a list of things for a computer to do

<noun> :: = <verb stem> "ing" (a noun that means what is done as meant by the verb)

Computer Programming Languages = ["The Java Programming Language", "APL"]

Definition = A part of program that does define a new word for use in other parts of the program

<verb> :: = <verb stem> "d"

| <verb stem> "ed" (a verb in the past tense)

Definition [Past Participle] = a form of the verb that says of a noun that what the verb means
has been done to it

<past participle> :: = <verb stem> "d"

| <verb stem> "ed"

Example = some one thing, out of a set of things, that is put in front so that we can see how
some part of that thing is in fact a part of each thing in the set

Syllable = a bit of sound that a mouth and tongue can say all at one time, more or less, in a
smooth way

Primitive = is a word for which we can take it for granted that we all know what it means

English = the language which this course (Programming Languages) is taught in

<word that can change what a noun means> :: = <word that can change what a noun means>
"er"

(more smooth to add "er" to the end of a word than to use the word "more" in front of it)

Better :: = <more good>

Because = the cause being that

Design = A plan for how to build a thing

<SKIP DEFINITION OF SOME LARGE NUMBERS IN THE SPEECH>

Ago = in the past, counting back from now

<noun> :: = <verb stem> "er" (a person or thing that does what the verb says to do)

Library = vocabulary designed to be added to a programming language to make its vocabulary larger

<SKIP Linux and Operating System>

Cathedral = a huge church (one designer in the plan)

Pattern = is a plan that has some number of parts and shows you how each pattern turns a face to the other parts, how each joins with the others or stands off, how each does what it does and how the other parts aid it or drag it down, and how all the parts may be grasped as a whole and made to serve as one thing, for some higher goal or as part of a larger pattern

Datum = set of bits that has a meaning

Data = the mass noun for a set of datums

Object = is a datum the meanings of whole parts is laid down by a set of language rules.

Method = a named piece of code that is part of an object

Generic Type = a pattern for building types

Operator = a glyph, such as a plus sign, that can be used in a language as if it were a word

And per se and = &...

Overloaded = to mean two or more things and the hearer/reader has to choose the meaning based on the rest of what is said

<SKIP polymorphic>

<SKIP complex number, rational number, interval, vector and matrix>

Meta = to step back from your own place

Language Design, Plan for Growth, Plan for Warts

Keep it short, Strive for Truth

I will use this library in the serve of this writing (Thus, I can use the word “library” and “writing”).

Though our task should not give terse sketch of the speech, I might have over there. I may have to start this thing now. I took what is set as rules in the speech as a library, since it will take too much time for me to start over and it is not droll as well.

Names of Persons = [“Guy Steele”]

The speech was on the Language Design creed, and the core theme of it kept changing as Programming Languages grew. The speaker, Guy Steele, noted that the goal of designing Programming Language is not to forge whole bunch of stuff that seems what would be needed and cram in as one Language, but to make the core and leave some tools for user to make their own tool. The point is to add what it need in deed, as with what the speaker called “Plan for Growth”. In fact, programmers may gripe for the lack of things the Programming Language can cope with, but they will give up learning all the forms and traits of the Language if it has too many things. The key is that the Language should grow with the users. It is cited by the speaker that “The Java Programming Language” would not have earn this much of users if they were at a size of now in just three years ago. I could not think in this kind of view up to now.

I like to make my words clear by matching with a like design, just like the speaker did to talk about the growing language with the case of starting the speech with one syllable English as primitives. He gave us the sense of raising the size of programming language by defining more words amid the speech and set some rules from the pattern of English to make the speech a bit smoother. Yet, new rules should not change the core part of the language, but it should act like it is built in from the beginning. This might sound like a base, but I could stride more from there to reach some broad wide end. We can put this view, or aim, in use to many more than just programming languages. One thing just popped up in my mind was Video Game.

Video Game = a type of game run on the computer

I used to trust there will be the great and prime game in my life, I still don’t know what that is, and I think I could play that game for all the time. But when a new game comes out, it is either too much load to start learning or too small to gorge. At last, the games that keep living for a long time is what has been changed and grown, just like what the speaker said. This might be the same for the game on our smart phone.

Definition [Value] = a thing that seems to have some more as a value

By the time one thing grows – it can be Programming Language, Game, or other things – the value in that thing grows as well, at least I think so. Well, it can be tools for Programming Languages, not like other thing that serve people for their joy or other value. Still, I think there are some pact by them.

Definition [Recursion] = when a meaning of a thing is in terms of it which loop back by recursion

By the way, I also want to put gist on the droll sense of speaker as a programming language designer. One of them is the form of definition. When he define new words, he didn’t use other rule to avoid the definition to rely on other rules. But above all, all the definition of words in this speech is defined by recursion. That is why I defined the two words in the following form, and we should use recursion to define recursion.

Definition [Variable] = a sign that means a place to store some things on a variable

All over, it is bold to say I felt a lot of awe in the way the speaker decided how to bring out his thought. From the view of Growth of Languages, he sic kept his way of speaking with the words set of first defined primitives, and soon defined more words that he needed in the mid of speech. His plan seemed that this way of speaking would bear his idea of growing vocabulary sets of a Language, but what I felt is that it is apt step of coding a program as well. Defining new word is like using new variable, new class, or new sources for next uses, when it seems to be used for other parts. All programmers might be groaning with this choice of use. It should now be clear for me as well that I thought really hard on this kind of choice many times in coming up with this writing.

Definition [Homework] = a piece of work asked by teacher to do out of a class as a homework.

As Guy Steele said, we can say a lot more than we think with English words set of one syllable. In fact, we can say many things, but he did not say that it would not be hard. Not in sure, this “homework” is straight in its meaning, ~~according to~~... as stated with the speaker. I grant that short words works well, and I hope I chose them well