Genome fragments

The genome of any living organism includes a succession of genes and non-coding fragments. These genes code for the proteins that perform a vast array of functions that control the cells that make up the organism. Genes may be oriented forward or backward, and we make the simplified assumption here that they never overlap. In between two successive genes there might be fragments that do not code for proteins. These non-coding fragments are sometimes also called *junk* DNA. Genome fragments can be represented in two different ways: graphical or symbolic. A genome fragment can be graphically represented in the following way:

- The number of symbols used for the graphical representation of a gene or a non-coding fragment is proportional to its length on the genome. You may assume here that all genes and non-coding fragments in the representation of a genome fragment (both graphical and symbolic) have a maximal length of 9.
- A forward gene is represented by zero or more equals signs (=) followed by a *greater then* character (>). As such, a forward gene of length four is represented as the string ===> and the string > itself represents a forward gene of length one.
- A backward gene is represented by a *less then* character (<) followed by zero or more equals signs (=).
- A non-coding fragment is represented by one or more dashes (-).
- If a backward gene is immediately followed by a forward gene — without an intermediate non-coding fragment — and if in addition at least one of both genes has a length that is at least two, these genes are separated by a vertical bar (|) in the graphical representation. This allows to determine unambiguously which equals signs belong to which genes. After all, the graphical representation <==> may be interpreted ambiguously as <|==>, <=|=> or <==|>. In contrast, the graphical representation <> cannot be interpreted ambiguously, and represents a backward gene of length one followed by a forward gene of length one.

According to these rules, the string ----==>--<===--<==|===>---- is the graphical representation of a genome fragment with a forward gene, followed by two successive backward genes and another forward gene. All genes are separated in this example by non-coding fragments, except for the last two genes. Because the last two genes are an example of the case described as the last item in the above list, a vertical bar must be included in between their graphical representation.

In the symbolic representation of a genome fragment, each gene or non-coding fragment is given as a code that is composed of a letter followed by a number. The letter indicates the type of the fragment (see table below) and the number gives the length of the fragment. As such, the code F4 describes a forward gene of length four, B1 a backward gene of length one, and N3 a non-coding fragment of length three. The codes for all successive fragments of the genome fragment are then concatenated into a single string. According to these rules, the genome fragment that was given above in its graphical representation, can be represented symbolically as N4F3N2B4N2B3F4N4.

fragment	letter	example	graphical
forward gene	F	F4	===>
backward gene	B	B1	<
non-coding fragment	N	N3	---

Assignment

1. Write a function code2graph, that translates a given code as used in the symbolic representation of a genome fragment into its corresponding graphical representation. A code must be passed as an argument to the function, and the function must return the corresponding graphical representation as a result. As such, if the code B3 is passed as an argument the function must return the string <==.
2. Use the function code2graph to write a function symb2graph. A symbolic representation of a genome fragment must be passed as an argument to this function. As such, if the string N4F3N2B4N2B3F4N4 is passed as an argument the function must return the string ----==>--<===--<==|===>----.
3. Write a function graph2symb. A graphical representation of a genome fragment must be passed as an argument to this function. The function must return the corresponding symbolic representation of the genome fragment as its result. As such, if the string ----==>--<====--<==|===>---- is passed as an argument the function must return the string N4F3N2B4N2B3F4N4.

Example

```
>>> code2graph('F4')
'===>'
>>> code2graph('B3')
'<=='

>>> symb2graph('N4F3N2B4N2B3F4N4')
'----==>--<===--<==|===>----'
>>> symb2graph('N2F4N2F4N2F4N2')
'--===>--===>--===>--'
>>> symb2graph('B1F1B1F1B1F1B1F1')
'<><><><>'
>>> symb2graph('B2F2B2F2B2F2B2F2')
'<=|=><=|=><=|=><=|=>'
>>> symb2graph('B2F1B1F2B2F1B1F2')
'<=|><|=><=|><|=>'

>>> graph2symb('----==>--<===--<==|===>----')
'N4F3N2B4N2B3F4N4'
>>> graph2symb('--===>--===>--===>--')
'N2F4N2F4N2F4N2'
>>> graph2symb('<><><><>')
'B1F1B1F1B1F1B1F1'
>>> graph2symb('<=|=><=|=><=|=><=|=>')
'B2F2B2F2B2F2B2F2'
>>> graph2symb('<=|><|=><=|><|=>')
'B2F1B1F2B2F1B1F2'
```

Het genoom van een levend organisme bestaat uit een opeenvolging van genen. Deze genen coderen voor de eiwitten die het functioneren van de cellen sturen waaruit het organisme is opgebouwd. Genen kunnen voorwaarts of achterwaarts georiënteerd zijn en we veronderstellen hier dat ze elkaar niet overlappen. Tussen twee genen kunnen fragmenten liggen die niet coderen voor eiwitten. Deze niet-coderende fragmenten worden soms ook *junk DNA* genoemd. Genoomfragmenten kunnen op twee manieren voorgesteld worden: *grafisch* of *symbolisch*. Een genoomfragment kan grafisch op de volgende manier voorgesteld worden:

- Het aantal symbolen dat gebruikt wordt voor de grafische voorstelling van een gen of een niet-coderend fragment staat in verhouding tot de lengte ervan op het genoom. Voor deze opgave mag je ervan uitgaan dat alle genen en niet-coderende fragmenten in de voorstelling van een genoomfragment (zowel grafisch als symbolisch) maximaal lengte 9 hebben.
- Een voorwaarts gen wordt voorgesteld door nul of meer opeenvolgende gelijktekens (=) gevolgd door een *groter dan* teken (>). Zo wordt een voorwaarts gen van lengte vier voorgesteld door de tekenreeks `===>` en stelt de tekenreeks `>` een voorwaarts gen van lengte één voor.
- Een achterwaarts gen wordt voorgesteld door een *kleiner dan* teken (<) gevolgd door nul of meer opeenvolgende gelijktekens (=).
- Een niet-coderend fragment wordt voorgesteld door één of meer opeenvolgende koppeltekens (-).
- Wanneer een achterwaarts gen onmiddellijk gevolgd wordt door een voorwaarts gen — zonder tussenliggend niet-coderend fragment — en wanneer bovendien minstens één van beide genen minstens lengte twee heeft, dan worden deze genen in de grafische voorstelling van elkaar gescheiden door een verticale streep (|). Op die manier kan ondubbelzinnig bepaald worden welke gelijktekens bij welk gen behoren. De grafische voorstelling `<===>` kan immers dubbelzinnig geïnterpreteerd worden als `<|==>`, `<=|=>` of `<==|>`. De grafische voorstelling `<>` kan daarentegen niet dubbelzinnig geïnterpreteerd worden, en stelt een achterwaarts gen van lengte één voor, gevolgd door een voorwaarts gen van lengte één.

Zo is de string `----==>--<====-<==|===>----` de grafische voorstelling van een genoomfragment met een voorwaarts gen, gevolgd door twee opeenvolgende achterwaartse genen, gevolgd door een voorwaarts gen. Alle genen worden hierbij gescheiden door niet-coderende fragmenten, behalve de laatste twee genen. Omdat deze laatste twee genen een voorbeeld vomen van het geval beschreven in het laatste puntje in bovenstaande lijst, moet er in de grafische voorstelling een verticale streep tussen beide genen geplaatst worden.

Bij de symbolische voorstelling van een genoomfragment wordt elk gen of niet-coderend fragment weergegeven als een code die bestaat uit een lettercode gevolgd door een getal. De lettercode geeft aan om welk type van fragment het gaat (zie onderstaande tabel), en het getal drukt de lengte van het fragment uit. Zo staat `V4` voor een voorwaarts gen van lengte vier, `A1` voor een achterwaarts gen van lengte één, en `N3` voor een niet-coderend fragment van lengte drie. De codes van de opeenvolgende fragmenten uit het genoomfragment worden dan na elkaar geplaatst in één enkele string. Zo kan het genoomfragment dat hierboven grafisch werd weergegeven, symbolisch worden voorgesteld als `N4V3N2A4N2A3V4N4`.

fragment	**lettercode**	**voorbeeld**	**grafisch**
voorwaarts gen	V	V4	`===>`
achterwaarts gen	A	A1	`<`
niet-coderend fragment	N	N3	`---`

Opgave

1. Schrijf een functie code2graf, die een gegeven code die gebruikt wordt in de symbolische voorstelling van een genoomfragment omzet naar de corresponderende grafische voorstelling. Er moet dus een code als argument aan de functie doorgegeven worden, en de functie moet de corresponderende grafische voorstelling als resultaat teruggeven. Zo moet de functie bijvoorbeeld voor de gegeven code A3 de string <== als resultaat teruggeven.
2. Gebruik de functie code2graf om een functie symb2graf te schrijven. Aan deze functie moet de symbolische voorstelling van een genoomfragment als argument doorgegeven worden. De functie moet dan de corresponderende grafische voorstelling van het genoomfragment als resultaat teruggeven. Zo moet de functie voor het argument N4V3N2A4N2A3V4N4 de waarde ----==>--<===--<==|===>---- teruggeven.
3. Schrijf een functie graf2symb. Aan deze functie moet de grafische voorstelling van een genoomfragment als argument doorgegeven worden. De functie moet dan de corresponderende symbolische voorstelling van het genoomfragment als resultaat teruggeven. Zo moet de functie voor de argument ----==>--<===--<==|===>---- de waarde N4V3N2A4N2A3V4N4 teruggeven.

Voorbeeld

```
>>> code2graf('V4')
'===>'
>>> code2graf('A3')
'<=='
>>> symb2graf('N4V3N2A4N2A3V4N4')
'----==>--<===--<==|===>----'
>>> graf2symb('----==>--<===--<==|===>----')
'N4V3N2A4N2A3V4N4'
>>> symb2graf('N4V3N2A4N2A3V4N4')
'----==>--<===--<==|===>----'
>>> graf2symb('----==>--<===--<==|===>----')
'N4V3N2A4N2A3V4N4'
>>> symb2graf('N2V4N2V4N2V4N2')
'--===>--===>--===>--'
>>> graf2symb('--===>--===>--===>--')
'N2V4N2V4N2V4N2'
>>> symb2graf('A1V1A1V1A1V1A1V1')
'<><><><>'
>>> graf2symb('<><><><>')
'A1V1A1V1A1V1A1V1'
>>> symb2graf('A2V2A2V2A2V2A2V2')
'<=|=><=|=><=|=><=|=>'
>>> graf2symb('<=|=><=|=><=|=><=|=>')
'A2V2A2V2A2V2A2V2'
>>> symb2graf('A2V1A1V2A2V1A1V2')
'<=|><|=><=|><|=>'
>>> graf2symb('<=|><|=><=|><|=>')
'A2V1A1V2A2V1A1V2'
```