EMMANUEL XAGORARAKIS

(E-mail: exagorarakis@gmail.com)

(Initial publication: November, 2017. Latest (present) update: April 2019)

The singular connection between the Binary Code and the Decimal System and the algorithm which offers the Singularity of Human Vs. Computer

And consequently

Nikola Tesla's number sequence of 3, 6, 9; why this works by the computational accuracy and endlessness and proves the binary code for the quantum computer.

For the Decade we have something that does not use factorization that is numbers beyond 1 and 2, in accordance with the non-factorization of the arithmetic entity of the dual code for which we know that 0 and 1 are not included under multiplication. Number 4 (I name it F) is the first-non-first (as non-prime) number. If F = 1^{st}, therefore order 1, the 1^{st} after the first in plain numerical order is number 2. 2 behaves the same so in addition (linear function) as in multiplication (factorization). So here the addition and the multiplication are so much in agreement as to the extent they are identified: 2+2 = 2X2. Thus for F we owe to apply both these cases for they are arithmetically identified: F(2)+2 or equally 2F+2. That is 2X4+2 = 10. The F is first/prime (prime systemic order – order 1) as non-first/non-prime (not the unit – 1), so this is translated in that since the Decade 10 = 2F+2 has only linear calculation, and since the F is first-non-first, therefore the decimal system is the system which is not first (it is not the unit, and the unit-by-unit calculation, like F is not the unit) arithmetically, but it is first systemically (like the F is first systemically, that is the first non-first number). Therefore the decimal system of arithmetic is singular, therefore indeconstructible as a system and not as mere and regardless-of-system (decimal, senary, and so on) calculation. That is the Decade (10 as numbers 0-9) is the singular/ incontestable factorization, therefore system, with all that this may possibly mean for the only indeconstructible numeric system. On these we can also say that 2 is the elementary (as first – order 1) prime number and 4 is the elementary (as first – order 1) non-prime number. And since the computer binary code (elements [0 – 1]) is also stated as elements [yes – no], and, factorial-wise, 2 is elementary [yes] and 4 is elementary [no], the decimal system is the binary [yes – no] in terms of factorization.

This binary [yes – no] as the relation of 2 with 4 is also secured as 2 being the [yes]-primality (as being prime) and 4 being homologous to 2 as prime (that is 2 and 4 are combined as homologous – both are prime) and opposite of 2 as 4 being non-prime (that is 2 and 4 are

the combination [yes – no]. Number 4 is prime-non-prime because it is non-factorial and factorial as n+n = nXn = 4; 2+2 = 2X2 = 4. This equality has been referred above.

I write F(2)+2 that is (FX2)+2 and not F+2(2) that is not (F+2)X2 = 6X2 = 12 for the F is, according to the above, the element of factorization –the prime/first factorization– so what is performs is apparently only factorizing (multiplying) and be factorized (multiplied). So there is FX2 and not F+2. But since, according to the above, there is mandatorily the +2 according to the addition 2+2, we have (FX2)+2 = F(2)+2 = 4(2)+2 = 8+2 = 10. If anything, as I mention above, there is equally 2+2 and 2X2. The complete/accurate statement of those two is (i) X2+2 and (ii) +2X2, that is we have number 2 so in its additional figure as in its factorial one. For (i) the X2 is not simple statement of the existence of 2 as "2" that is "+2", but factorial statement. Thus we must have the other factor so that we are talking about factorization that is factorial statement of 2 as X2. And, according to the above, for "X2" there is only 4X2. For "X2" there is the statement 1X2 = 2, but here, for factorization, we have, as we wrote, number 4 as the factor "1" that is "order 1", and since "X2" is factorization, we can only state [order 1]X2 that is 1^{st}X2 = 4 and not 1X2 = 2. Therefore, as it is obvious, the statement 2+2 (that is X2+2) in terms of this study, is valid only as 4X2+2.

The binary code and the decimal system are connected singularly. Artificial intelligence does not have a will of its own, but it always performs accurately. The computer does not function factorially, but we humans, who have our own will and judgment and furthermore we are the only wise species and thus have civilization and science, we calculate and, more generally, think factorially. That is factorial thinking comes with wisdom. And as by the hereby proof, which connects the accurate thinking that is computing, i.e. the binary code, with the decimal system, we see that for human life, science and civilization, the only accurate and wise (according to human wisdom) can only be performed by the decimal numeric system. Is it by luck that the decimal system is the globally instituted system, which serves the universal language that is Mathematics?

So, one issue finally rises by the, as proven, singular connection of the binary code (computing) with the Decimal System (humanity): what would be the functional unification of them so that computing is unified with humanity? That is the computer serving the human individual who is wise and autonomous, and serving in the singular way (according to the as above proven singular relationship) that is the unification of the computer with the human individual. I will try to prove this here based on the present proof I've already written.

It is obvious that we seek an algorithm which can serve the human in the singular that is absolute way. The equation proven is 4X2+2 = 10 and it has two parts; the factorial and the additional one. In the arithmetical respect, each functional element in the parameters of the algorithm must be dual that is composed by digits 0 and 1, according to the arithmetical accuracy of computing/ binary coding. That is we refer to linear function (addition) and this is given by case (i) above: in the X2+2 the +2 is the addition and +2 is analyzed in +1+1 or equally: 1, 1. That is, in the combinatorial sense, two digits, and these can only be 0 and 1. Those two elements are regarded in linear (computational) function, so they do not function factorially, i.e. not as in the decimal/ human system, so they must be interpreted in

functional elements that do not decide in the human way, but only in the computational (elementarily simple) one. So, [0-1] is [yes-no].

For case (ii) above we have 4X2, so this is the factorial (according to the decimal/ human) aspect of the supposed algorithm. Because this function is factorial, it is parameterized as 2^4. So the combinations are 16 and they are dual/binary because for our attempted algorithm we combine cases (i) and (ii). So, +2 and X2 absolutely agree in duality; number 2.

We saw the arithmetical aspect of the algorithm: 16 dual (0-1) parameters. In the linear (elementary) case (i) the elements are the computational [yes-no] (Computer Function – CF). In the factorial case (ii) the elements must be, as we already explained, the human ones (Human Function – HF). Now, what may be the only two and absolutely general aspects for the human intellect and existence? The only duality in this case can be [quantity-quality]. This duality is first stated by Aristotle, the Father of Logic, and it is also self-evident for every aspect of our intellect and life is studied and actually exists as both quantity and quality. This duality is universal. For case (i) quality cannot be a functional element for it calls for judging ability, i.e. decision of approval or rejection that is seeing right from wrong, and this ability belongs to the autonomous (human) and not to the mere machine (computer).

So we have two sets of [0-1]: The one (A) set is [quantity-quality] and the other (B) set is [yes-no]. So, the each parameter is a combination of (A) with (B). It is right that the (A) is always first for it is the one carrying the decision, and decision as a notion and in its general/ elementary sense, is the initiation of any process; this is an algorithm for human decision-making, and we cannot have any function unless we decide it.

So here is the pattern of the algorithm in a table:

Possible combinations: 2^4=16			
	HF	CF	Human Achievement
1	0 – 0	0 – 0	0
2	0 – 0	0 – 1	0
3	0 – 0	1 – 0	0
4	0 – 0	1 – 1	0
5	0 – 1	0 – 0	Result Verification
6	0 – 1	0 – 1	Result Verification
7	0 – 1	1 – 0	Result Verification
8	0 – 1	1 – 1	Result Verification
9	1 – 0	0 – 0	Result
10	1 – 0	0 – 1	Result
11	1 – 0	1 – 0	Result
12	1 – 0	1 – 1	Result
13	1 – 1	0 – 0	0
14	1 – 1	0 – 1	0
15	1 – 1	1 – 0	0
16	1 – 1	1 – 1	0

We remind that this is the algorithm that serves the human being's function in its accuracy and fullness, because it derives from the unification of computing with our own entity. Computing performs mathematically by the accuracy and fullness of mathematics. The human performs (and should always perform) by his own will. So, we must evaluate this algorithm's functionality.

For column CF the 0 and 1 (the Yes and No) must, as said, be seen functionally: If 0=No and 1=yes, then the combination 0 – 0 is No *to* No = Yes. So it is the function "Yes". Accordingly 0 – 1 = No, 1 – 1 = Yes, and 1 – 0 = No. But we cannot think the same for column HF: 0 = Quantity and 1 = Quality. As it is commonly known, *for* a quantity there is always a quality defining it, and *for* a quality there is always a quantity defining it. So 0 – 0 is quantity *for* quantity and 1 – 1 is quality *for* quality. As obvious, in column HF the 0 – 0 and 1 – 1 bear no function, so, in the table, series 1-4 and 13-16 are not functional. Nevertheless, in terms of the number-theoretical part of this study, column HF belongs, as it is produced by, the factorial part of the Decade –case (ii) above– that is 4X2. So, although series 1-4 and 13-16 are not functional because of column HF, theoretically the 0 and 1 of A only exist factorially, i.e. as multiplication. So 0X0 = quantity X quantity = quantity, that is 0X0 = 0, and quality X quality = quality, that is 1X1 = 1. And, as we mentioned, for quality there is always quantity, and vice versa. So there is [0 – 0] = [0] = [0 – 1] and [1 – 1] = [1] = [1 – 0]. So, the number theory discovered here is judged as valid.

Now, algorithmically thinking, we are interested in series 5-12 of the table. Column HF separates, as obvious, series 5-8 from series 9-12. In A, 0 = quantity and 1 = quality. And we remind that the 0 and 1 of A apply to human intellect and existence. As we wrote, the 1 = quality is about the choices and criteria that we perform. So, 0 = quantity is regardless of criteria and so it is about mere and pure effort that is *power of attention* regardless of what this serves (for 0 is not 1). So, for this algorithm to work on the human being, series 9-12 are the case. That is this algorithm causes *choice* (1) to our *attention* (0) as 1 – 0 (column HF) as to the 0 or 1 (No or Yes) of column CF. That is we have the effect of putting/ setting (choosing) attention (1 – 0).

We remind that as we above wrote, column CF offers the mere No (0) or mere Yes (1) in the algorithm. As we wrote, this algorithm applies to the human entity in its completeness. And the state of [No] is refraining from function, and the [Yes] is the state of approving function. In the set of series 9-12 there are two [No] (series 10, 11) and two [Yes] (series 9, 12). So there has to be an initial [Yes] and a final [Yes], as well as an initial [No] and a final [No]. So, how are these different and what is the priority between them (the [No]s and the [Yes]s)?

To answer this, we remind that column CF exists purely arithmetically/ linearly and not in the complex/ factorial way, that is as addition and not multiplication – unlike column HF. So, in the two [Yes], series 12 and 9, we have for series 12 [Yes *to* Yes] and for series 9 [No *to* No]. The [Yes] is the positive decision and the [No] is the negative decision of the human as to the algorithm. But in column CF the [No] is identified with number zero (0). Zero is not a positive number but it is not negative either! Nevertheless, series 9 is functional. Therefore we are only led to the conclusion that series 12 is the first to choose. So series 12 bears the existing positive function (as [Yes *to* Yes]). So since the **positivity** [Yes] = [Yes *to* Yes] = [1 – 1] is

preceded, and [Yes] = [1], then for the [No] of series 10 and 11, the positivity of [No] proceeds [Yes *to* No] (series 11) and follows series 10 [No *to* Yes]. We remind here that positivity (1) is indeed positivity while negativity (0) is actually not negativity, but a neutral element that is not actual negation, so therefore it comes after the dynamic/ actual positivity.

So in column CF, we have the notion/ element of dynamicity as preceding in time (algorithmic function) the notion/ element of neutrality. How can this be translated for the human intellect? Apparently, dynamicity is what fosters our function and makes choices, while neutrality is just and merely functioning. The mere function is the mere "servant" that does not push or choose. So it is clear that the pure/ mere logic is this neutrality for logic is the processor that provides the emotion with the sufficient data, so that the emotion or "heart" decides upon the logical data. But yet, the emotion has to firstly decide so that the "merely/ neutrally working" system that is logic, works.

So we have the initial and the final parameters/ functions as initial [Yes] or [No] that is parameters 12 and 11 applied/ offered simultaneously and final [Yes] or [No] that is parameters 10 and 9 applied/ offered simultaneously. Now about the each time preference between the simultaneously offered parameters, this is verified by the human and the proof on this lies in series 5-8. Series 5-8 are for column CF in one to one correspondence with series 9-12. The [0 – 1] of column HF in series 5-8 states that once the computer causes (choice to) attention [1 – 0] (series 9-12, column HF), and both the positive and negative choice are offered, the human has the fluent ability to verify the [Yes] or [No] or reject the [Yes] or [No] and turn to the opposite one. This is secured as proven by the existence of parameters in series 5-8 of the algorithm, because they are one to one correspondent with parameters 9-12; they are identified as to the [Yes] and [No] and they constitute the freedom on the fundamental choice on [Yes] or [No] because of column HF, series 9-12.

Nikola Tesla's number sequence of 3, 6, 9; why this works by the computational accuracy and endlessness and proves the binary code for the quantum computer.

Number 3 is ***second prime*** **[I]** after the first prime that is 2. Number 4 is the first-non-first (as we explained above) and this means that the first which is non-prime (i.e. first+1 non-prime=***second-non-prime*** **[II]**) after 4 is 6. Number 9 is the first prime after number 7, which means that it is two units after 7, and 7 is one unit after 6, and this means that 7+2 = prime = ***second prime*** (3) + ***second non-prime*** (6)= 3+6 = 9. So number 9 is second and it is prime-non-prime because the elements ***second*** and ***prime*** and ***non-prime*** all exist in the identity 3+6=9 and this identity is, as we'll reveal below, divisible-non-divisible and a prime-non-prime number that is 9, and so the identity is indivisible-non-indivisible, like number 4, and so we cannot disregard or subtract any element of it. So as we referred, this indivisible identity, which is an addition, is: 9 = 3+6 = ***second prime*** + ***second non-prime*** and the unified expression of it is ***second prime-non-prime*** **[III].** The first prime-non-prime, as we already discovered, is number 4. And the term ***second***, as we wrote initially in this manuscript, is order 2 that is either number 2 (second after number 1) or number 4 as the first-non-first (i.e. non-prime) that is the +1 after number 3 that is 1^{st} + 1 = 4 that is order 2 or, equally, number 2^{nd}. So the term ***second*** is secured for number 9. And this obviously means that 9 agrees, as we explained, both with the referred quality of number 2 and with the quality of number 4.

Case [I] is the YES primality (i.e. 3 it is a prime number). And this study is engaged in indivisible number-relations as based in primality, i.e. indivisibility. In this aspect the [I] is the YES, [II] is the NO of [I] and [III] is the YES-NO of [ii] (as it is apparent in the above paragraph) that is the YES-NO of [I]. As we wrote, [III] is indivisible/inseparable in its elements, so for it there is YES=NO. And as it is apparent right above, the three cases ([I], [II], [III]) are connected serially. And all these are inside the decimal system that is numerals 0-9, as synopsis (factorization) 10, which is as here proven, only perfect number that is the only binary (serial) factorization: F(2)+2 = 10.

So we have numbers 3, 6, 9 as proven to be, apart from the known binarity 0-1, the only and complete binary sequence YES-NO-YES-NO inside the, as here proven, only perfect number that is the only binary (serial) factorization: F(2)+2 = 10. So this is the proof why the additional sequence of 3, 6, 9 works accurately in the decimal system, which is the only perfect, as having no end in its accuracy. This means that numbers 3, 6, 9 as both serially and systemically (i.e. decimally) added and combined are always accurate.

The sequence 3, 6, 9, as we here analyzed and described, as being both serial and systemic/factorial, is always precise and complete, while it gives both the identity and the diversity in the computational YES-NO, thus giving the way for the function of the quantum computer.

The identity 0=1 and together with it the diversity 0-1 are given also by the algorithm of the present work for it is apparent in the second paragraph of page 5 above, for column HF of the algorithm that “there is [0 – 0] = [0] = [0 – 1] and [1 – 1] = [1] = [1 – 0]”. Therefore also the algorithm of the human being must be examined and function quantum-mechanically that is by the code of the quantum computer as it is here given based on the Tesla sequence 3, 6, 9.

The quantum binary observation/ calculation calls for two issues for mathematical/ computational statement and proof: the **superposition** and the **entanglement.** The code given here by the proof of the sequence 3, 6, 9 is, as we write above, the YES-NO-YES=NO. The superposition is given as by the identity YES=NO that is the third element of the as above sequence/ combination. The entanglement is an endless binary fuzziness. As by the above statement it is given that the first element [YES] is combined with the third element [YES=NO]. Thus the YES is YES, which is NO, thus the YES is and is not YES, and the YES is and is not NO. Similarly as by the combination of the second element of the sequence/ combination, the [NO], with the third one, the [YES=NO], it comes about that the NO is NO, which is YES, thus the NO is and is not NO, and the NO is and is not YES. Consequently, by the mathematical accuracy and simplicity, it has also been here proven the entanglement of the quantum binary observation.