Discarding Algorithm for Rational Roots of Integer Polynomials (DARRIP)

Jesús Álvarez Lobo

J3A3L3@gmail.com

Spain

Dedicated to Nityangi

ABSTRACT

The algorithm presented here is to be applied to polynomials whose independent term has many divisors. This type of polynomials can be hostile to the search for their integer roots, either because they do not have them, or because the first tests performed have not been fortunate.

This algorithm was first published in ***Revista Escolar de la Olimpíada Iberoamericana de Matemática***, ***Number 19 (July - August 2005). ISSN – 1698-277X***, in Spanish, with the title ***ALGORITMO DE DESCARTE*** **DE RAÍCES ENTERAS DE POLINOMIOS.**

When making this English translation 12 years later, some erratum has been corrected and when observing from the perspective of time that some passages were somewhat obscure, they have been rewritten trying to make them more intelligible.

The algorithm is based on three properties of divisibility of integer polynomials, which, astutely implemented, define a very compact systematic that can simplify significantly the exhaustive search of integer roots.

Although there are many other methods for discarding roots, for example, those based on bounding rules, which sometimes drastically reduce the search interval, for the sake of simplicity, they will not be considered here.

Despite it is a basic issue that supposedly has already been studied exhaustively, the truth is that the foundations of science are the most important, since everything else must be supported on them.

Each year, hundreds of thousands of mathematical articles are published, most of them at a postgraduate level, so that it is already physically impossible to take into account all the information related to a certain subject of study.

The study presented here could be useful to almost all the young people of the planet, since at some stage of their academic training they will have to solve polynomial equations with integer coefficients, looking for rational solutions, integer or fractional.

All of us who have dedicated ourselves to the teaching of Mathematics can remember good students who, for example, have abandoned an exercise of simplification of algebraic fractions because they were unable to factorize some of the polynomials involved, more for lack of time than for knowledge. This has been precisely the motivation to undertake the study of DARRIP

The author's dream, fully convinced of the advantages of the method presented here, is nothing more than to see that the DARRIP becomes incorporated into the curricula of all the elementary study centers in the world.

▣ 1. Properties underlying the algorithm for integer roots of integer polynomials.

Let

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 = \sum_{j=0}^{n} a_j x^j .$$

be an integer polynomial, that is, a polynomial with integer coefficients.

- **1.1.** *If* $x_0 \neq 0$ *is an integer root of* $p(x)$, *then* x_0 *is a divisor of* a_0.

  Proof

$$p(x_0) = \sum_{j=0}^{n} a_j x_0^j = a_0 + \sum_{j=1}^{n} a_j x_0^j = 0 \Rightarrow a_0 = -\sum_{j=1}^{n} a_j x_0^j = -x_0 \sum_{j=1}^{n} a_j x_0^{j-1}$$

$$\Rightarrow \boxed{\frac{a_0}{x_0} = -\sum_{j=1}^{n} a_j x_0^{j-1} \in \mathbb{Z}}$$

- **1.2.** *If* $x_0 \notin \{-1,\ 0,\ 1\}$ *is an integer root of* $p(x)$, *then* $x_0 - 1$ *is a divisor of* $p(+1)$.

  Proof

$$\frac{p(+1)}{x_0 - 1} = \frac{a_0 + \sum_{j=1}^{n} a_j}{x_0 - 1} = \frac{-\sum_{j=1}^{n} a_j x_0^j + \sum_{j=1}^{n} a_j}{x_0 - 1} = -\frac{\sum_{j=1}^{n} a_j (x_0^j - 1)}{x_0 - 1}$$

By the *remainder theorem*, it is clear that the binomials $p_k(x_0) = x_0^k - 1$, as polynomials in x_0, are divisible by $x_0 - 1$, since $p_k(1) = 0$. Let

$$q_k(x_0) = \frac{x_0^k - 1}{x_0 - 1},\ k = 1,\ 2,\ \cdots,\ n.$$

Then,

$$\frac{p(+1)}{x_0 - 1} = -a_n \frac{x_0^n - 1}{x_0 - 1} - a_{n-1} \frac{x_0^{n-1} - 1}{x_0 - 1} - \cdots - a_1 \frac{x_0 - 1}{x_0 - 1} = -\sum_{k=1}^{n} a_k q_k(x_0),$$

and considering again the indeterminate x_0 as an integer constant, it follows that

$$\frac{p(+1)}{x_0 - 1} = -\sum_{k=1}^{n} a_k q_k(x_0) \in \mathbb{Z} \sim \{-1,\ 0,\ 1\},$$

that is, $x_0 - 1$ divides $p(+1)$.

Note: Although the property is trivially verified for $x_0 = 0$, is not considered this value because it has no practical interest in implementing the algorithm.

- **1.3.** *If* $x_0 \in \mathbb{Z} \sim \{-1, 0, 1\}$ *is a root of* $p(x)$, *then* $x_0 + 1$ *is a divisor of* $p(-1)$.

Proof

The proof of this property is analogous to the previous one.

The properties 1.2 and 1.3 can be stated in this more compact form:

- **1.2-3.** *If* $x_0 \in \mathbb{Z} \sim \{-1, 0, 1\}$ *is a root of* $p(x)$, *then* $x_0 \pm 1$ *is a divisor of* $p(\mp 1)$.

2. DARRIP fundamentals

By 1.1, integer roots must divide the independent term. So, we will take as initial candidates to integer roots the set of divisors of a_0.

The application of the DARRIP will only be profitable if a_0 has many divisors, in which case it can reduce the number of divisions necessary to discover the integer roots.

This algorithm, based on the three properties demonstrated in the previous section, is a much finer “sieve” than the one based only on property 1.1, since it involves two new criteria, determining a much more selective “filtering”; often, the integers that cross it are only the roots of the polynomial.

The algorithm has been designed in a way that minimizes writing, and that its execution is intuitive and mnemonic.

2.1. Complexity

If a_0 has d positive divisors greater than 1, also it has d negative divisors less than -1.

To apply the properties 1.2-3, these $2d$ divisors must be incremented by ±1, which produces $4d$ other numbers to consider, and the number of divisors would be $6d$.

Thus, the direct application of 1.2-3 would have the following price: it would be necessary to make $4d$ "extra" tests of divisibility, and to write a threefold number of divisors of absolute value greater than one, in exchange for avoiding a certain number of divisions...

In order to improve the implementation of the algorithm, let's note the following additional properties:

If x_0 is a positive integer root of $p(x)$, then:

- **2.2.** $x_0 - 1$ *divides* $p(+1)$, *and this implies that* $-(x_0 - 1) = -x_0 + 1$ *divides* $p(+1)$.

- **2.3.** $x_0 + 1$ *divides* $p(-1)$, *and this implies that* $-(x_0 + 1) = -x_0 - 1$ *divides* $p(-1)$.

These last properties, although they may seem trivial, allow to reduce drastically the amount of writing, simply **extending the divisibility tests of the positive divisors**:

- *If* $x_0 - 1$ doesn´t divide $p(+1)$, x_0 *is discarded as root*.
- *If* $x_0 - 1$ doesn´t divide $p(-1)$, $-x_0$ *is discarded as root*.
- *If* $x_0 + 1$ doesn´t divide $p(+1)$, $-x_0$ *is discarded as root*.
- *If* $x_0 + 1$ doesn´t divide $p(-1)$, x_0 *is discarded as root*.

In a more compact form:

$$\boxed{\frac{p(\pm 1)}{x_0 \mp 1} \notin \mathbb{Z} \Rightarrow p(x_0) \neq 0}$$
$$\boxed{\frac{p(\pm 1)}{x_0 \pm 1} \notin \mathbb{Z} \Rightarrow p(-x_0) \neq 0}$$

Obviously, for the purposes of divisibility, $p(\pm 1)$ can be replaced by $|p(\pm 1)|$.

- **2.4. Algorithm configuration** ____

Let $D_0 = \{ d_1, d_2, \cdots, d_{s-1}, d_s \}$ be the **set of all positive divisors** of a_0, such that $1 < d_1 < d_2 < \cdots < d_{s-1} < d_s$.

Let´s set up the following schema:

$\lvert p(+1) \rvert$						$\lvert p(+1) \rvert$
	d_1	d_2	$\cdots$	$d_{s\text{-}1}$	d_s	
$\lvert p(-1) \rvert$						$\lvert p(-1) \rvert$

For $k = 1, 2, \ldots, s$, we must perform the following divisibility tests:

❶ Is $|p(+1)|$ divisible by $d_k - 1$?

❷ Is $|p(-1)|$ divisible by $d_k - 1$?

❸ Is $|p(+1)|$ divisible by $d_k + 1$?

❹ Is $|p(-1)|$ divisible by $d_k + 1$?

- **2.3. Rule of diagonals** ____

When we test the divisibility by $d_k - 1$ we work with $|p(+1)|$ and $|p(-1)|$ on the left; when we test the divisibility by $d_k + 1$, we work with $|p(+1)|$ and $|p(-1)|$ on the right.

Whenever one of these divisibility tests is negative, we plot in the cell containing the divisor tested, d_k, the diagonal whose direction is determined by the relative position of $|p(+1)|$ or $|p(-1)|$ with respect to d_k.

There are four cases:

- If $d_k - 1$ doesn´t divide $|p(+1)|$, we must draw the *main diagonal* (ascending from right to the left), since $|p(+1)|$ is considered to the left and above d_k when is tested $d_k - 1$. This means that d_k has been discarded.

- If $d_k - 1$ doesn´t divide $|p(-1)|$, we must draw the *secondary diagonal* (down from right to the left), since $|p(-1)|$ is considered to the left and below d_k when is tested $d_k - 1$. This means that $-d_k$ has been discarded.

- If $d_k + 1$ doesn´t divide $|p(+1)|$, we must draw the *secondary diagonal*, since $|p(+1)|$ is considered to the right and above d_k when is tested $d_k + 1$. This means that $-d_k$ has been discarded.

- If $d_k + 1$ doesn´t divide $|p(-1)|$, we must draw the *main diagonal*, since $|p(-1)|$ is considered to the right and below d_k when is tested $d_k + 1$. This means that d_k has been discarded.

Discard criteria

- *A **main diagonal** discards a **positive divisor**.*
- *A **secondary diagonal** discards a **negative divisor**.*

Obviously, the discarded divisors no longer need to be tested.

▣ 3. DARRIP Implementation

Steps for applying the Algorithm:

① Find the polynomial $q(x)$ resulting from eliminating all the roots 0, 1, -1 (with any degree of multiplicity) from $p(x)$ and to reduce their coefficients to *relatively prime* (dividing, if necessary, by the GCD of these). The Discarding algorithm is applied to this *reduced polynomial*, $q(x)$.

② Calculate the values $q(+1)$ and $q(-1)$.

Let $p_1(x)$ be the polynomial resulting from eliminating in $p(x)$ the roots 0 and 1 and reduce their coefficients to relatively prime.

If -1 isn´t a root of $p(x)$, i.e., if $p(-1) \neq 0$, then $q(x) = p_1(x)$ and $q(+1) = p_1(+1)$ is the first nonzero remainder obtained during the test of the root +1.

But if -1 is a root of $p(x)$, then $q(x) \neq p_1(x)$. In this case, we can get $q(+1)$ without having to divide $q(x)$ by $x-1$, nor apply the *remainder theorem*.

Indeed, if -1 is a root of $p_1(x)$ with order of multiplicity μ, we have:

$$p_1(x) = (x+1)^\mu q(x) \Rightarrow q(x) = \frac{p_1(x)}{(x+1)^\mu} \Rightarrow \boxed{q(+1) = \frac{p_1(+1)}{2^\mu}}$$

Thus, by this *correction formula*, $q(+1)$ can be calculated from $p_1(+1)$, whose value is simply the first nonzero remainder obtained during the test of the root +1.

$q(-1)$ is the first nonzero remainder obtained during the test of the root -1.

③ Write in the algorithmic schema the divisors of the *independent term* of $q(x)$ greater than unity, as well as $q(+1)$ and $q(-1)$, duplicated in the form previously indicated.

④ Proof the divisibility of $q(+1)$ and $q(-1)$, by each divisor decremented by 1, applying the *rule of diagonals* when the test is negative.

⑤ Proof the divisibility of $q(+1)$ and $q(-1)$, by each divisor incremented by 1, applying the *rule of diagonals* when the test is negative.

Shortcuts. Before applying the discarding algorithm, it is advisable to observe if any of these conditions is met:

❶ **All the coefficients are of the same sign.** In this case, **positive real roots** cannot exist; therefore, all the main diagonals would be drawn.

❷ **The coefficients of the terms with different parity in their degrees have a different sign.** In this case, **negative real roots** cannot exist; therefore, all the secondary diagonals would be drawn.

❸ **The independent term is odd and the sum of the remaining coefficients is even.** In this case **integer roots** cannot exist, so it is no longer necessary to apply the discarding algorithm.

Proof of ❶

It is obvious that if all the coefficients have the same sign (positive / negative), the polynomial cannot be null for any positive value.

Let $p(x)=\sum_{j=0}^{n} a_j x^j,\ a_j<0,\ j=0,1,2,\cdots,n;\ x_0\in\mathbb{R}^+$. Then, $a_j x^j<0,\ j=0,1,2,\cdots,n$;

so, $p(x_0)=\sum_{j=0}^{n} a_j x^j<0.$

Let $p(x)=\sum_{j=0}^{n} a_j x^j,\ a_j>0,\ j=0,1,2,\cdots,n;\ x_0\in\mathbb{R}^+$. Then, $a_j x^j>0,\ j=0,1,2,\cdots,n$;

so, $p(x_0)=\sum_{j=0}^{n} a_j x^j>0.$

Proof of ❷

If $x_0\in\mathbb{R}^-$ then the terms of even degree do not change sign but the odd degree terms do change; therefore, all terms have the same sign. So, the value of the polynomial it isn´t null.

Proof of ❸

Since $p(+1)=\sum_{j=0}^{n} a_j$, then $p(+1)-a_0=\sum_{j=1}^{n} a_j$, and if a_0 is odd and $\sum_{j=1}^{n} a_j$ is even, $p(+1)-a_0$ is even, so, $p(+1)$ is odd. Let x_0 be an integer root of $p(x)$; then, x_0 should be a divisor of a_0, therefore, x_0 must be odd and then x_0-1 should be even. Thus, x_0-1 it isn´t a divisor of $p(+1)$ that is odd, in contradiction with property 1.2. So, x_0 it isn´t a root of $p(x)$, $\forall x_0\in\mathbb{Z}\sim\{0\}$.

4. Example 1

Let's look at the process through an example.

Find all the integer roots of the polynomial,

$$\boxed{p(x)=2x^8+50x^7+48x^6-2x^5-52x^4-96x^3+2x^2+48x}$$

Extracting the greatest common factor,

$$p(x)=2x(x^7+25x^6+24x^5-x^4-26x^3-48x^2+x+24)$$

Applying now *Ruffini´s algorithm* to $x^7+25x^6+24x^5-x^4-26x^3-48x^2+x+24$ for the roots ± 1:

	1	**25**	**24**	**-1**	**-26**	**-48**	**1**	**24**	
+1		1	26	50	49	23	-25	-24	
	1	**26**	**50**	**49**	**23**	**-25**	**-24**	0	
+1		1	27	77	126	149	124		
	1	**27**	**77**	**126**	**149**	**124**	100		$p_1(+1)=100$
	1	**26**	**50**	**49**	**23**	**-25**	**-24**		
-1		-1	-25	-25	-24	1	24		
	1	**25**	**25**	**24**	**-1**	**-24**	**0**		
-1		-1	-24	-1	-23	24			
	1	**24**	**1**	**23**	**-24**	**0**			$q(x)=x^4+24x^3+x^2+23x-24$
-1		-1	-23	22	-45				
	1	**23**	**-22**	**45**	**-69**				$q(-1)=-69$

The calculations made so far are the usual ones for the search of the roots ± 1.

The first nonzero remainder got by iterating in the *Ruffini´s algorithm* for the root -1 is $q(-1)$. If $p(-1)\neq 0$, i.e., -1 isn´t a root of $p(x)$, then $q(+1)=p_1(+1)$; but in this case -1 is a root of $p(x)$, with order of multiplicity μ = 2; therefore, we will apply the above formula of correction:

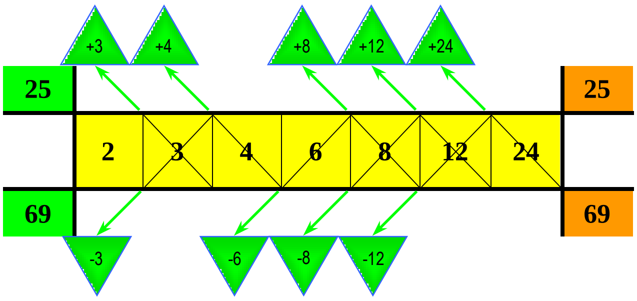
$$\boxed{q(+1)=\frac{p_1(+1)}{2^\mu}}=\frac{100}{2^2}=25 \Rightarrow \boxed{q(+1)=25}$$

And the set of divisors of 24 greater than 1 is {2, 3, 4, 6, 8, 12, 24}.

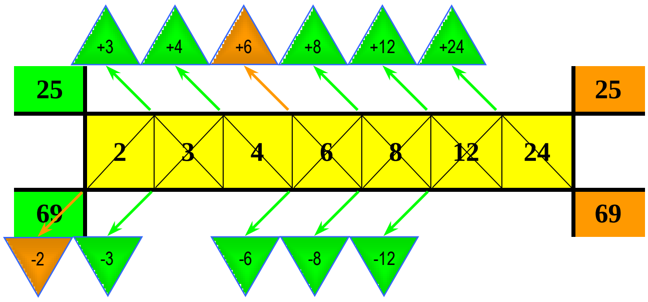
So, the schema for the discarding algorithm is the following:

25								**25**
	2	**3**	**4**	**6**	**8**	**12**	**24**	
69								**69**

We test the divisibility of 25 and 69 by each of the divisors decreased by 1. Applying the *rule of diagonals*, the values inside the green triangles are discarded as roots:



We test the divisibility of 25 and 69 by each of the divisors increased by 1. Applying the *rule of diagonals*, the values inside the orange triangles are discarded as roots:



Up to now we have discarded 11 of the 14 candidates to integer root other than $0, \pm 1$ (7 positive and 7 negative), with very little effort.

In this case, by the application of the discarding algorithm have been avoided 11 divisions by the *Ruffini algorithm*, in exchange for the simple scheme,

25								**25**
	2	**3**	**4**	**6**	**8**	**12**	**24**	
69								**69**

whose construction only involves operations that are carried out quickly and without difficulty: subtract or add the unit and apply criteria of divisibility.

It only remains to prove by division or *theorem of the rest* whether $+2,\ -4,\ -24$ are roots of $q(x)$:

	1	**24**	**1**	**23**	**-24**
+2		2	52	106	258
	1	**26**	**53**	**129**	**234**

	1	**24**	**1**	**23**	**-24**
-4		-4	-80	316	258
	1	**20**	**-79**	**339**	**-1380**

	1	**24**	**1**	**23**	**-24**
-24		-24	0	-24	24
	1	**0**	**1**	**-1**	**0**

Hence, the only integer root of $q(x)$ is $\boxed{x=-24}$.

Factoring $q(x)$,

$$q(x)=x^4+24x^3+x^2+23x-24=(x+24)(x^3+x-1)$$

The other three roots of $q(x)$ are the real one,

$$x=\sqrt[3]{\frac{\sqrt{93}}{18}+\frac{1}{2}}-\sqrt[3]{\frac{\sqrt{93}}{18}-\frac{1}{2}}$$

and the two complex conjugates,

$$x=\frac{1}{2}\left(\sqrt[3]{\frac{\sqrt{93}}{18}-\frac{1}{2}}-\sqrt[3]{\frac{\sqrt{93}}{18}+\frac{1}{2}}\right)\ \pm\ i\left(\frac{1}{2}\left(\sqrt[3]{\frac{\sqrt{31}+3\sqrt{3}}{2}}+\sqrt[3]{\frac{\sqrt{31}-3\sqrt{3}}{2}}\right)\right)$$

and they have been calculated using symbolic computation software.

⊡ 5. FRACTIONARY RATIONAL ROOTS

Fortunately, the polynomials with which we use to work are usually not as hostile as the $q(x)$ of the previous example.

However, the applications of the discard algorithm are not limited to the search for integer roots. In fact, since any polynomial can be transformed by a change of variable into another polynomial whose rational roots are all integers, the discarding algorithm is apt to be applied for the discarding of *fractional rational roots*.

- **5.1. Fractional rational roots: possibility of existence criterion.**

Let $p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$ be an integer polynomial of degree *n*. If the irreducible fraction $\dfrac{a}{b}$ is a root of $p(x)$, we have:

$$p\left(\frac{a}{b}\right) = a_n \left(\frac{a}{b}\right)^n + a_{n-1} \left(\frac{a}{b}\right)^{n-1} + \cdots + a_1 \left(\frac{a}{b}\right) + a_0 = 0\,,$$

and from this equation, multiplying all its terms by b^{n-1},

$$a_n \frac{a^n}{b} = -\left(a_{n-1}a^{n-1} + a_{n-2}a^{n-2}b + \cdots + a_1 a b^{n-2} + a_0 b^{n-1}\right) \in \mathbb{Z},$$

and being *a* and *b* co-prime, also a^n and *b* must be co-prime; therefore, we conclude that *b* must divide the coefficient a_n.

Otherwise,

$$\frac{b^n}{a}\, p\left(\frac{a}{b}\right) = a_n a^{n-1} + a_{n-1} a^{n-2} b + \cdots + a_1 b^{n-1} + a_0 \frac{b^n}{a} = 0,$$

and

$$a_0 \frac{b^n}{a} = -\left(a_n a^{n-1} + a_{n-1} a^{n-2} b + \cdots + a_1 b^{n-1}\right) \in \mathbb{Z}.$$

A reasoning analogous to the previous one, proves that *a*, that doesn´t divide b^n, has to divide a_0. Therefore, is met this ***necessary condition for fractional rational roots:***

"An irreducible fraction a/b can be a root of an integer polynomial $p(x)$ only if *a* divide the independent term of $p(x)$ and *b* divide the coefficient of the term of greatest degree of $p(x)$." See the *Rational Root Theorem* [1].

Therefore, an integer polynomial can´t have fractional rational roots if the coefficient of the highest degree (leading coefficient) is 1, i.e., if the polynomial is *monic*.

The result we have reached is the so-called *Rational Root Theorem (RRT)* [2].

- **5.2. Transformed to integer polynomial whose rational roots are all integers.**

Hence, we are interested in the transformation of a polynomial with coefficients in $\mathbb{Z}$ into a *monic polynomial* with coefficients in $\mathbb{Z}$. Such transformation is carried out by the change of variable,

$$\boxed{x \equiv \frac{y}{a_n}}$$

and multiplying all the coefficients of $p(x)$ by a_n^{n-1}:

$$\boxed{p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0}$$
$$\downarrow$$
$$\boxed{x \equiv \frac{y}{a_n},\ q(y) = a_n^{n-1} p\left(\frac{y}{a_n}\right)}$$
$$\downarrow$$
$$\boxed{q(y) = a_n^{n-1} p\left(\frac{y}{a_n}\right) = y^n + a_{n-1} y^{n-1} + \cdots + a_n^{n-2} a_1 y + a_n^{n-1} a_0}$$

The polynomial $q(y)$ can be written directly, simply multiplying each coefficient a_k of $p(x)$ by a_n^{n-k-1}, for $k = n, n-1, \cdots, 1, 0$, as shown below:

$$\boxed{p(x)} \Rightarrow \left\{\begin{array}{lcl} a_n & \xrightarrow{\cdot\, a_n^{-1}} & 1 \\ a_{n-1} & \xrightarrow{\cdot\, a_n^{0}} & a_{n-1} \\ a_{n-2} & \xrightarrow{\cdot\, a_n^{1}} & a_n a_{n-2} \\ a_{n-3} & \xrightarrow{\cdot\, a_n^{2}} & a_n^2 a_{n-3} \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \\ a_1 & \xrightarrow{\cdot\, a_n^{n-2}} & a_n^{n-2} a_1 \\ a_0 & \xrightarrow{\cdot\, a_n^{n-1}} & a_n^{n-1} a_0 \end{array}\right\} \Rightarrow \boxed{q(y) = y^n + a_{n-1} y^{n-1} + \cdots + a_n^{n-2} a_1 x + a_n^{n-1} a_0}$$

The *fractional roots* of the primitive polynomial $p(x)$ are obtained by dividing the integer roots of the transformed polynomial $q(y)$ by the leading coefficient a_n:

$$\boxed{x_i \equiv \frac{y_i}{a_n}}$$

6. Example 2

Find all the rational roots of the polynomial,

$$\boxed{p(x) = 3x^5 + x^4 - 2x^3 - 12x + 8}$$

Monic transformed of $p(x)$:

$$y = 3x \Rightarrow q(y) = 3^4 p\left(\frac{y}{3}\right) \Rightarrow \boxed{q(y) = y^5 + y^4 - 6y^3 - 324y + 648}$$

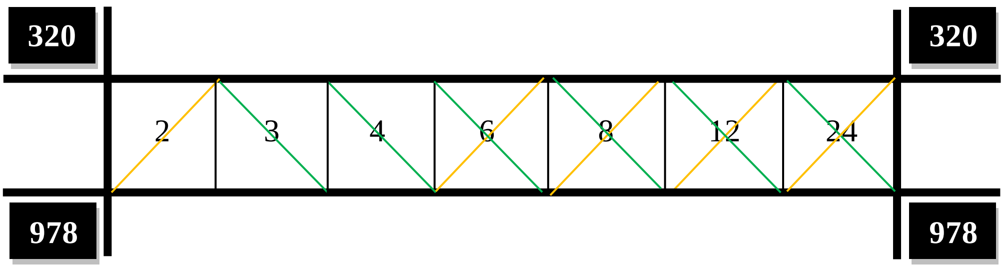
Applying the traditional method of testing the divisors of 648 in increasing order with Horner´s algorithm, we would get:

	1	**1**	**-6**	**0**	**-324**	**648**
+1		1	2	-4	-4	-328
	1	**2**	**-4**	**-4**	**-328**	**320**

	1	**1**	**-6**	**0**	**-324**	**648**
-1		-1	0	6	-6	330
	1	**0**	**-6**	**6**	**-330**	**978**

According to the *RRT* (5.1), if $p(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$ is an integer polynomial, $a, b \in \mathbb{Z} \mid \mathrm{GCD}(a,b) = 1$ and $p\left(a/b\right) = 0$, then $a \mid a_0$, $b \mid a_n$.

Taking into account the change of variable $y = 3x$, if a/b is an irreducible rational root of $p(x)$, a must be a divisor of 8, therefore, by the RRT, as candidates for rational roots of $p(x)$ it is not necessary to take the huge* set of the divisors of 648, since it is sufficient to consider the subset $\Delta = \{\delta \mid \delta \in \mathbb{Z}^+,\ \delta \mid 3 \cdot 8\} \sim \{1\} = \{2,\ 3,\ 4,\ 6,\ 8,\ 12,\ 24\}$.



(*) $648 = 2^3 \cdot 2^4 = 2(3+1)(4+1) = 40$ divisors!.

And now it is enough to try using Horner's algorithm if any of the integers 2, -3 and -4 are roots of the monic polynomial $q(y)$:

	1	**1**	**-6**	**0**	**-324**	**648**
+2		2	6	0	0	-648
	1	**3**	**0**	**0**	**-324**	**0**

Hence, $y = 2$ is an integer root of $q(y)$; so, $x = \frac{2}{3}$ is a rational root of $p(x)$.

After removing the root $y = 2$ in $q(y)$, we have the polynomial $r(y) = y^4 + 3y^3 - 324$.

	1	**3**	**0**	**0**	**-324**
-3		-3	0	0	0
	1	**0**	**0**	**0**	**-324**

	1	**3**	**0**	**0**	**-324**
-4		-4		0	64
	1	**-1**	**4**	**-16**	**-260**

Hence, in this case the discarding algorithm only filtered two false candidates to rational roots, which supposes an economy of 27 divisions.

So, $p(x)$ has only one rational root, $\boxed{x = \frac{2}{3}}$.

7. Alternative method for fractional rational roots

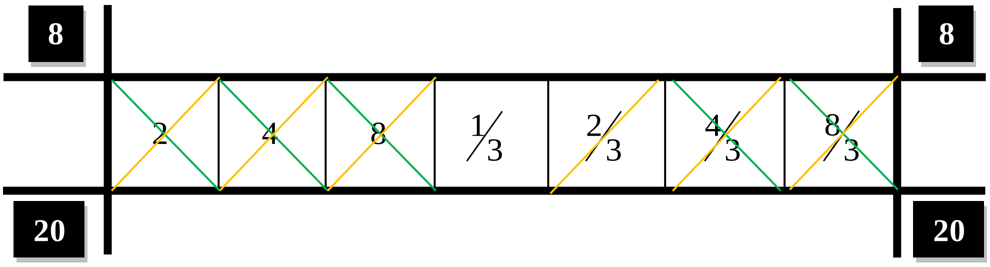
By suggestion of *Evan O'Dorney* (a brilliant American Mathematics´student), I expose other way of seeking for fractional rational roots, without reducing the polynomial to a monic form. The procedure is based on the following divisibility properties:

- **7.1.** If $p(x)$ is an integer polynomial, $a,b\in\mathbb{Z}\,|\,\mathrm{GCD}(a,b)=1$ and $p\left(a/b\right)=0$, then $a\pm b\mid p(\mp 1)$.

- **7.2.** Applying these properties to the polynomial of the previous example, we have:

	3	**1**	**-2**	**0**	**-12**	**8**
+1		3	4	2	2	-10
	3	**4**	**2**	**2**	**-10**	$\boxed{\mathbf{-8}}$

	3	**1**	**-2**	**0**	**-12**	**8**
-1		-3	2	0	0	12
	3	**-2**	**0**	**0**	**-12**	$\boxed{\mathbf{20}}$



	3	**1**	**-2**	**0**	**-12**	**8**
$1/3$		1	$2/3$		$-4/27$	$-328/81$
	3	**2**	$-4/3$	$-4/9$	$-328/27$	$\boxed{320/81}$

	3	**1**	**-2**	**0**	**-12**	**8**
$-1/3$		-1	0		$-2/9$	$110/27$
	3	**0**	**-2**	$2/3$	$-110/9$	$\boxed{326/27}$

	3	**1**	**-2**	**0**	**-12**	**8**
$2/3$		2	2			-8
	3	**3**	**0**	**0**	**-12**	$\boxed{\mathbf{0}}$

The properties 7.1. allows to apply the DARRIP directly to non-monic polynomials for the discarding of fractional roots; however, given that these properties are only *necessary* but not *sufficient*, the fractional candidates who have not been discarded must be subjected to the division test and this entails a cost in effort that goes against the spirit of the method, which is ***maximum simplicity***.

8. Example 3

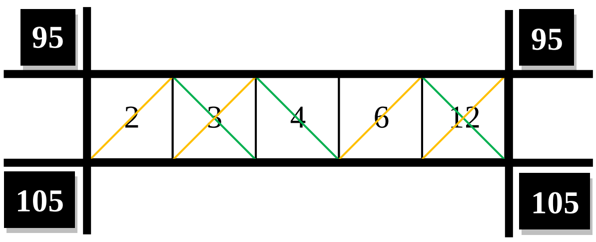
$$\boxed{p(x) = 3x^4 + 4x^3 - 3x^2 - x + 4} \longrightarrow \boxed{q(y) = y^4 + 4y^3 - 9y^2 - 9y + 108}$$

- **Set of values to be tested in the DARRIP:**

$$\Delta = \left\{ \delta \mid \delta \in \mathbb{Z}^+,\ \delta \mid 3 \cdot 4 \right\} \sim \{1\} = \{2,\ 3,\ 4,\ 6,\ 12\}$$

	1	**4**	**-9**	**-9**	**108**
+1		1	5	-4	-13
	1	**5**	**-4**	**-13**	**95**

	1	**4**	**-9**	**-9**	**108**
-1		-1	-3	12	-3
	1	**3**	**-12**	**3**	**105**



	1	**4**	**-9**	**-9**	**108**
2		2	12	6	-6
	1	**6**	**3**	**-3**	**102**

	1	**4**	**-9**	**-9**	**108**
-4		-4		36	-108
	1	**0**	**-9**	**27**	**0**

	1	**4**	**-9**	**-9**	**108**
6		6	60	306	1782
	1	**10**	**51**	**297**	**1890**

So, $p(x)$ has only one rational root, $\boxed{x = -\dfrac{4}{3}}$.

▣ 9. Example 4

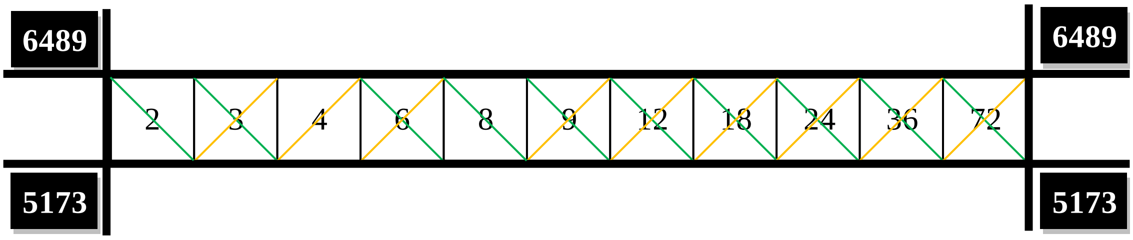
$$\boxed{p(x)=9x^5-x^4-8x^3+9x+8} \longrightarrow \boxed{q(y)=y^5-y^4-72y^3+729y+5832}$$

$$\Delta=\left\{\delta \mid \delta\in\mathbb{Z}^+,\ \delta \mid 9\cdot 8\right\}\sim\{1\}=\left\{2,\ 3,\ 4,\ 6,\ 8,\ 9,\ 12,\ 18,\ 24,\ 36,\ 72\right\}$$

	1	**-1**	**-72**	**0**	**729**	**5832**
+1		1			-72	657
	1	**0**	**-72**	**-72**	**657**	**6489**

	1	**-1**	**-72**	**0**	**729**	**5832**
-1		-1	2		-70	659
	1	**-2**	**-70**	**70**	**659**	**5173**

After applying the DARRIP,



all the candidates for *rational root numerator* of $p(x)$ are discarded except two, -2 & -8, which must be subjected to the synthetic division test:

	1	**-1**	**-72**	**0**	**729**	**5832**
-2		-2	6		-264	-930
	1	**-3**	**-66**	**132**	**465**	**4902**

	1	**-1**	**-72**	**0**	**729**	**5832**
-8		-8	72			-5832
	1	**-9**	**0**	**0**	**729**	**0**

So, $p(x)$ has only one rational root, $\boxed{x=-\dfrac{8}{9}}$.

As can be seen, although the monic polynomial has an independent term with 56 divisors, initial candidates, under the RRT many of these are excluded. Comparing the effort we have made to find the rational roots of $p(x)$ through the DARRIP with the exhaustive inspection by synthetic division (*Horner-Ruffini´s algorithm*), which would require 24 divisions, 16 of which involving fractions, the advantage of the DARRIP when we face integer polynomials with many rational root candidates is clear.

References

[1] https://en.wikipedia.org/wiki/Rational_root_theorem

[2] https://www.cut-the-knot.org/Generalization/RationalRootTheorem.shtml

ꙮ