Civ1 Land Value calculation explained

Discussion in 'Civ1 - General Discussions' started by darkpanda, Jun 18, 2013.

darkpanda
Dark Prince

Joined:	Oct 28, 2007
Messages:	600

It took me a while, but I finally nailed it: by diving into CIV.EXE's internal workings (version EN 474.01), I eventually figured out the **exact land value computation algorithm** employed by CIV to assign values to map squares, in order for AI Civs to choose suitable locations for building cities.

Before going into the details, I want to, once again, express my grateful thanks to **Gowron**, without whom I would probably never have made it so far.

So, here it goes, in order to compute the land value of map squares, CIV processes as follows:

- Compute a **score for each terrain type**, based on the terrain type characteristics; this score has a different value for *normal* terrain types and terrain types with *special resources*
- Compute the **land value for each map square** depending on the square's surrounding 'city squares'

1. Terrain Type Score Calculation

For each terrain type, the **score** is calculated based on 2 sets of data:

- Terrain **production** attributes
- Terrain **modification** attributes

The terrain **production** attributes are described by **Gowron** in this post, under title **4. Terrain Stats**.

Basically, they control how each terrain type will generate food, shields and trade, as well as impede movement and benefit defense. CIV keeps different production attributes for *normal* terrain and for terrain with *special resources*.
Hereunder are the production attributes for all 24 terrain types in CIV (12 normal, 12 with special resource):

Column 1	**Column 2**	**Column 3**	**Column 4**	**Column 5**	**Column 6**	**Column 7**	**Column 8**
Name	Movement cost	Defense ratio	Food	Shields	Trade	unknown	ID
Desert	1	2	0	1	0	1	14
Plains	1	2	1	1	0	1	6
Grassland	1	2	2	1	0	1	10
Forest	2	3	1	2	0	2	2
Hills	2	4	1	0	0	2	12

Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8
Mountains	3	6	0	1	0	3	13
Tundra	1	2	1	0	0	0	7
Arctic	2	2	0	0	0	0	15
Swamp	2	3	1	0	0	0	3
Jungle	2	3	1	0	0	0	11
Ocean	1	2	1	0	2	0	1
River	1	3	2	1	1	2	9
Oasis	1	2	3	1	0	1	14
Horses	1	2	1	3	0	1	6
Grassland special	1	2	2	1	0	1	10
Game	2	3	3	2	0	2	2
Coal	2	4	1	2	0	2	12
Gold	3	6	0	1	6	3	13
Game	1	2	3	0	0	0	7
Seals	2	2	2	0	0	0	15
Oil	2	3	1	4	0	0	3
Gems	2	3	1	0	4	0	11
Fish	1	2	3	0	2	0	1
River special	1	3	2	1	1	2	9

The terrain **modification** attributes are described by **Gowron** in this post, under title **5. Terrain Improvement**.

They control the *modification behaviour* of a terrain type: how much food/shields does it provide if mined/irrigated, to which terrain type does it change if mined/irrigated, and additional AI-only data.

CIV only keeps 12 records for modification attibutes, which are the same whether a terrain type has a special resource or not:

Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7
Name	Irrig. food bonus	Irrig. cost	Mining shield bonus	Mining cost	Can AI improve under Despotism/Anarchy?	Can AI improve under Monarchy or above?
Desert	-2	5	-2	5	no	no
Plains	-2	5	2	15	yes	yes

Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7
Grassland	-2	5	2	10	no	yes
Forest	6	5	-1	5	no	no
Hills	-2	10	-4	10	yes	yes
Mountains	-1	0	-2	10	no	no
Tundra	-1	0	-1	0	no	no
Arctic	-1	0	-1	0	no	no
Swamp	10	15	2	15	no	no
Jungle	10	15	2	15	no	no
Ocean	-1	0	-1	0	no	no
River	-2	5	-1	0	no	yes

The **score calculation** is done as follows:

- Initial score is **trade + 3*food**
- If the terrain **is neither River nor Grassland**, then **add 2*shield** to the score
- If the terrain **has Mining shield bonus (<0)**, then **add -(Mining shield bonus + 1)** to the score
- **ELSE** If the terrain **has Irrigation food bonus (<0)**, then **add -2*(Irrigation food bonus + 1)** to the score

When applying this calculation formula to CIV's default terrain attributes, the scores obtained are as below:

Column 1	Column 2	Column 3	Column 4	Column 5
Name	trade + 3*food	+2*shield (0 if not applicable)	+mining bonus OR +2*irrig bonus	**SCORE**
Desert	0	2	1	**3**
Plains	3	2	2	**7**
Grassland	6	0	2	**8**
Forest	3	4	0	**7**
Hills	3	0	3	**6**
Mountains	0	2	1	**3**
Tundra	3	0	0	**3**
Arctic	0	0	0	**0**
Swamp	3	0	0	**3**
Jungle	3	0	0	**3**
Ocean	5	0	0	**5**
River	7	0	0	**7**
Oasis	9	2	1	**12**
Horses	3	6	2	**11**

Column 1	Column 2	Column 3	Column 4	Column 5
Grassland special	6	0	2	**8**
Game	9	4	0	**13**
Coal	3	4	3	**10**
Gold	6	2	1	**9**
Game	9	0	0	**9**
Seals	6	0	0	**6**
Oil	3	8	0	**11**
Gems	7	0	0	**7**
Fish	11	0	0	**11**
River special	7	0	0	**7**

The scores above are used when calculating the map squares' land value, as described in the next section.

2. Map Square Land Value Calculation

First of all, map squares for which the land value is calculated are **in the range [2,2] - [77,47]**. That is to say that the 4 rows of arctic and antarctic land (rows 0, 1, 48 and 49), as well as the 4 columns around the "TimeZone Meridian" (columns 0, 1, 78 and 79) have a land value of 0, whatever their type.

For each square within this range, the **land value is calculated as follows (initial value is 0)**:

- If the square's terrain type is **not Plains, Grassland or River**, then its land value is **0**
- Else, **for each 'city square' neighbouring the map square** (i.e. each square following the city area pattern, including the map square itself, so totally 21 'neighbours'), compute the following **neighbour value** (initially 0):
    - If the neighbour square type is **Grassland special** or **River special**, **add 2**, then **add** the non-special **Grassland or River terrain type score** to the neighbour value
    - Else **add neighbour's terrain type score** to the neighbour value
    - If the neighbour square is in the map square **inner circle**, i.e. one of the 8 neighbours immediatly surrounding the map square, then **multiply the neighbour value by 2**
    - If the neighbour square is the **North square** (relative offset 0,-1), then **multiply the neighbour value by 2** ; *note: I actually think that this is a bug, and that the intention was rather to multiply by 2 if the 'neighbour' was the central map square itself... the actual CIV code for this is to check if the 'neighbour index' is '0'; the neighbour index is used to retrieve the neighbour's relative offset coordinates (x,y) from the central square, and the central square itself is actually the* last *in the list (index 20), the* first *one (index 0) being the North neighbour; another '7x7 neighbour pattern' table found in CIV code does indeed set the central square at index 0, and this why I believe ths is a programmer's mistake...*
    - **Add the neighbour's value to the map square total value** and loop to next neighbour
- After all neighbours are processed, if the *central* map square's terrain type is **non-special Grassland or River**, **subtract 16** from total land value
- **Substract 120 (0x78)** from the total land value, and **remember its sign**
- Set the land value to the **absolute land value** (i.e. negate it if it is negative)
  {*] **Divide the land value by 8**
- If the **land value was negative 3 steps before, then negate the land value and add 1, i.e. value = 1-value**
- **Adjust the land value to the range [1..15]**

- **Divide the land value by 2**
- And finally, **add 8** to the land value

That's it!

This algorithm was reversed-engineered from CIV.EXE: after some debugging and fine-tuning, I have counter-tested it against a half-dozen random new CIV games, yielding a 100% success rate compared to Civ-calculated values.

If you have any problem implementing or applying it by yourself, or or if you are not obtaining the expected values, please let me know, I may have misdescribed or missed some elements from the algorithm.

Obviously, my next step will be to implement it within **JCivED** 🙂

JCivED - a toolbox for Sid Meier's Civilization (MS-DOS)

darkpanda, Jun 18, 2013 #1

---

kirkham7
Chieftain

Joined: Dec 22, 2012
Messages: 288
Location: Hayward, CA

Wow, this is amazing.

kirkham7, Jun 18, 2013 #2

Renergy
Chieftain

Joined: Nov 2, 2010
Messages: 197
Location: Czech Republic

Wow.

In my opinion, there ought to be a special sub-forum for top quality posts/threads such as this one, which I would name "**CIV internals**".

List of threads that could go there (I'm not saying that list is complete)

http://forums.civfanatics.com/showthread.php?t=498532
http://forums.civfanatics.com/showthread.php?t=331078
http://forums.civfanatics.com/showthread.php?t=493581
http://forums.civfanatics.com/showthread.php?t=486470
http://forums.civfanatics.com/showthread.php?t=492843
http://forums.civfanatics.com/showthread.php?t=491335
http://forums.civfanatics.com/showthread.php?t=126921
http://forums.civfanatics.com/showthread.php?t=333598
http://forums.civfanatics.com/showthread.php?t=490003
http://forums.civfanatics.com/showthread.php?t=478234
http://forums.civfanatics.com/showthread.php?t=494994
http://forums.civfanatics.com/showthread.php?t=495521
http://forums.civfanatics.com/showthread.php?t=494886
http://forums.civfanatics.com/showthread.php?t=339049
http://forums.civfanatics.com/showthread.php?t=492500
http://forums.civfanatics.com/showthread.php?t=235082
http://forums.civfanatics.com/showthread.php?t=223182

perhaps even my own thread could be included as well
http://forums.civfanatics.com/showthread.php?t=488141

Edit: Perhaps a better title for the subforum would be "**CIV internals & modding**"

Renergy, Jun 18, 2013 #3

simonnomis
Chieftain

Joined: Nov 29, 2005
Messages: 195

Excellent job Darkpanda and Gowron!

I have memories of long arduous trawls through these numbers with Dack. I'm glad someone finally figured it out.

simonnomis, Jun 18, 2013 #4

---

darkpanda
Dark Prince

Joined: Oct 28, 2007
Messages: 600

> simonnomis said: ↑
>
> *I have memories of long arduous trawls through these numbers with Dack. I'm glad someone finally figured it out.*

I **did** spend a large amount of time creating sample maps, and even routines to make statistical analysis of Civ-generated land values, but now that I see the complexity of it, I don't think it could have been guessed otherwise... Especially the final steps "substract 120, divide by 8, ..."

JCivED - a toolbox for Sid Meier's Civilization (MS-DOS)

darkpanda, Jun 18, 2013 #5

---

simonnomis
Chieftain

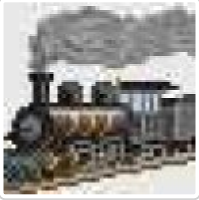
Joined: Nov 29, 2005
Messages: 195

I can imagine that it took that amount of effort!

You're right about guessing it, I think most of us gave up, realising that too many factors were involved in the calculation for a simple analysis. Which is not really surprising, given that the positioning of cities is probably the most critical factor in the whole game.

It's a shame that the game has to be decrypted in this way to reveal its secrets.

simonnomis, Jun 18, 2013 #6



Dack
Terra Form
Moderator

Joined: Nov 26, 2003
Messages: 530
Location: civ dos 474.05

The following table taken from the first post in this thread indicating the base **scores** for each of the 24 land types.

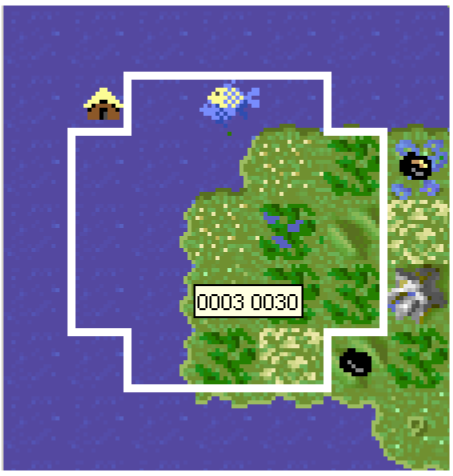
Column 1	Column 2	Column 3
Land type	regular	special
00	0	00
01 ocean	5	11 Fish
02 forest	7	13 Game
03 swamp	3	11 Oil
04	0	00
05	0	00
06 plain	7	11 Horses
07 tundra	3	09 Game
08	0	00
09 river	7	07
10 grassland	8	08

Column 1	Column 2	Column 3
11 jungle	3	07 Gems
12 hill	6	10 Coal
13 mountain	3	09 Gold
14 desert	3	12 Oasis
15 arctic	0	06 Seals

To calculate the city build value of each of the 21 squares I stepped through each neighborhood square in the following order (00 to 20).

Code:

```
  |11|12|13|
--+--+--+--+--
10|04|00|05|14
--+--+--+--+--
09|03|20|01|15
--+--+--+--+--
08|07|02|06|16
--+--+--+--+--
  |19|18|17|
```



Square 0 thru 7 makeup the **inner circle** and get the ZS (zero thru seven) treatment base score times 2.
N is the **North square** an additional multiply by 2.
gr is for non-SpecialResourcesSquare (river or grassland) who get a -16 added to its total.
rT is the running total for all 21 city squares.

Code:

```
                             special  Score
offset                       resource
 X  Y square XY   Land       True/false      events
+0 -1 i00  x03y29 L01 ocean     srsF S05
+0 -1 i00  x03y29 L01 ocean     srsF S05  ZS+010
+0 -1 i00  x03y29 L01 ocean     srsF S05  ZS+010 N+020
+0 -1 i00  x03y29 L01 ocean     srsF S05  ZS+010 N+020   rT+0020
+1 +0 i01  x04y30 L11 jungle    srsF S03
+1 +0 i01  x04y30 L11 jungle    srsF S03  ZS+006
+1 +0 i01  x04y30 L11 jungle    srsF S03  ZS+006         rT+0026
+0 +1 i02  x03y31 L10 grassland srsF S08
+0 +1 i02  x03y31 L10 grassland srsF S08  ZS+016
+0 +1 i02  x03y31 L10 grassland srsF S08  ZS+016 gr+000
+0 +1 i02  x03y31 L10 grassland srsF S08  ZS+016 gr+000  rT+0026
-1 +0 i03  x02y30 L01 ocean     srsF S05
-1 +0 i03  x02y30 L01 ocean     srsF S05  ZS+010
-1 +0 i03  x02y30 L01 ocean     srsF S05  ZS+010         rT+0036
-1 -1 i04  x02y29 L01 ocean     srsF S05
-1 -1 i04  x02y29 L01 ocean     srsF S05  ZS+010

-1 -1 i04  x02y29 L01 ocean     srsF S05  ZS+010         rT+0046
+1 -1 i05  x04y29 L06 plain     srsF S07
```

```
+1 -1 i05  x04y29 L06 plain     srsF S07
+1 -1 i05  x04y29 L06 plain     srsF S07  ZS+014
+1 -1 i05  x04y29 L06 plain     srsF S07  ZS+014         rT+0060
+1 +1 i06  x04y31 L02 forest    srsF S07
+1 +1 i06  x04y31 L02 forest    srsF S07  ZS+014
+1 +1 i06  x04y31 L02 forest    srsF S07  ZS+014         rT+0074
-1 +1 i07  x02y31 L01 ocean     srsF S05
-1 +1 i07  x02y31 L01 ocean     srsF S05  ZS+010
-1 +1 i07  x02y31 L01 ocean     srsF S05  ZS+010         rT+0084
-2 +1 i08  x01y31 L01 ocean     srsF S05
-2 +1 i08  x01y31 L01 ocean     srsF S05                 rT+0089
```

In this example the total is 162.

Then subtract 120

162 - 120 = 42

Divide by 8

42 \ 8 = 5 integer divide

Adjust the land value to the range [1..15]
Should this be mod 16 or the equivalent bit clear 0xFFF0 ?

5 MOD 16 = 5

Another divide by 2

5 \ 2 = 2 integer divide

And finally, add 8 to the land value

I assume this could be a bit set 0x8 but add should also work as the value should be below 7 at this point.

so the value I derived is 10d or 0xA or 1010 binary

The CIV calculation is 12d or 0xC

Where did I go wrong?

Attached Files:

CIVIL3.SVE File size: 37 KB Views: 32	CIVIL3.MAP File size: 12.4 KB Views: 49
ms3.png File size: 8 KB Views: 733	

Dack, Jul 6, 2013 #7

darkpanda
Dark Prince

Joined: Oct 28, 2007
Messages: 600

> Dack said: ↑
>
> *gr is for non-SpecialResourcesSquare (river or grassland) who get a -16 added to its total.*

The **subtraction of value 16** if the square is not special river/grassland **only occurs once, i.e. when the *central* square is not special r/g**.

I rechecked my description of the algo, and this step is correctly indented, *after* all neighbours are processed, but I realize the wording "map square" may not be clear enough... I'll update it later on.

Thanks for your careful review!

JCivED - a toolbox for Sid Meier's Civilization (MS-DOS)

darkpanda, Jul 7, 2013 #8

Dack
Terra Form
Moderator

Joined: Nov 26, 2003
Messages: 530
Location: civ dos 474.05

> darkpanda said: ↑
>
> *The* ***subtraction of value 16*** *if the square is not special river/grassland* ***only occurs once, i.e. when the central square is not special r/g****.*
>
> *I rechecked my description of the algo, and this step is correctly indented, after all neighbours are processed, but I realize the wording "map square" may not be clear enough... I'll update it later on.*

I tried it both ways. I thought it was out of the loop but was unsure. Thanks for the quick response.

Unfortunately I must be doing something else incorrect.

I have check and rechecked my **score** table. To the extent of having the program print it out in debug mode. It appears to match the table in your first post.

My assumption is that **Grassland special** and **River special** refer to squares that would be special resource squares but happen to be terrain type Grassland or River. Is that correct?

Code:

```
        regular   special resource
        score     score
00            00  00
01 ocean      05  11   Fish
02 forest     07  13   Game
03 swamp      03  11   Oil
04            00  00
05            00  00
06 plain      07  11   Horses
07 tundra     03  09   Game
08            00  00
09 river      07  07
10 grassland  08  08
11 jungle     03  07   Gems
12 hill       06  10   Coal
13 mountain   03  09   Gold
14 desert     03  12   Oasis
15 arctic     00  06   Seals
```

> darkpanda said: ↑
>
> - *If the neighbour square type is* ***Grassland special*** *or* ***River special****,* ***add 2****, then* ***add*** *the non-special* ***Grassland or River terrain type score*** *to the neighbour value*
> - *Else* ***add neighbour's terrain type score*** *to the neighbour value*
> - *If the neighbour square is in the map square* ***inner circle****, i.e. one of the 8 neighbours immediatly surrounding the map square, then* ***multiply the neighbour value by 2***
> - *If the neighbour square is the* ***North square*** *(relative offset 0,-1), then* ***multiply the neighbour value by 2*** *;* ***...***
> - ***Add the neighbour's value to the map square total value*** *and loop to next neighbour*

Following the above
ZS is the multiply by 2 for the inner circle
Note: inner circle does not include the center square
N is the north square it gets a multiply by 2

Code:

```
                             special  Score
offset                       resource
 X  Y square XY    Land      True/false      events
+0 -1 i00  x03y29 L01 ocean     srsF S05
+0 -1 i00  x03y29 L01 ocean     srsF S05  ZS+010

+0 -1 i00  x03y29 L01 ocean     srsF S05  ZS+010 N+020
+0 -1 i00  x03y29 L01 ocean     srsF S05  ZS+010 N+020 pT+0020
```

```
+0 -1 i00  x03y29 L01 ocean     srsF S05  ZS+010 N+020 rT+0020
+1 +0 i01  x04y30 L11 jungle    srsF S03
+1 +0 i01  x04y30 L11 jungle    srsF S03  ZS+006
+1 +0 i01  x04y30 L11 jungle    srsF S03  ZS+006 rT+0026
+0 +1 i02  x03y31 L10 grassland srsF S08
+0 +1 i02  x03y31 L10 grassland srsF S08  ZS+016
+0 +1 i02  x03y31 L10 grassland srsF S08  ZS+016 rT+0042
-1 +0 i03  x02y30 L01 ocean     srsF S05
-1 +0 i03  x02y30 L01 ocean     srsF S05  ZS+010
-1 +0 i03  x02y30 L01 ocean     srsF S05  ZS+010 rT+0052
-1 -1 i04  x02y29 L01 ocean     srsF S05
-1 -1 i04  x02y29 L01 ocean     srsF S05  ZS+010
-1 -1 i04  x02y29 L01 ocean     srsF S05  ZS+010 rT+0062
+1 -1 i05  x04y29 L06 plain     srsF S07
+1 -1 i05  x04y29 L06 plain     srsF S07  ZS+014
+1 -1 i05  x04y29 L06 plain     srsF S07  ZS+014 rT+0076
+1 +1 i06  x04y31 L02 forest    srsF S07
+1 +1 i06  x04y31 L02 forest    srsF S07  ZS+014
+1 +1 i06  x04y31 L02 forest    srsF S07  ZS+014 rT+0090
-1 +1 i07  x02y31 L01 ocean     srsF S05
-1 +1 i07  x02y31 L01 ocean     srsF S05  ZS+010
-1 +1 i07  x02y31 L01 ocean     srsF S05  ZS+010 rT+0100
-2 +1 i08  x01y31 L01 ocean     srsF S05
-2 +1 i08  x01y31 L01 ocean     srsF S05  rT+0105
-2 +0 i09  x01y30 L01 ocean     srsF S05
```

For this central square the total is 178

> darkpanda said: ↑
>
> - *After all neighbours are processed, if the central map square's terrain type is **non-special Grassland or River**, **subtract 16** from total land value*
> - ***Substract 120 (0x78)** from the total land value, and **remember its sign***
> - *Set the land value to the **absolute land value** (i.e. negate it if it is negative)*
>
> Click to expand...

0178 - 120 = 0058
absolute value is 0058
divide by 8 is 0007
bit clear &HFFF0 is 0007 <======= **Adjust the land value to the range [1..15]**
I assume the the above is mod 16

divide by 2 = is 0003
bit set &H0008 is 0011
Calculated value = +0011

CIV City Build value = +0012

The square in question is X=3 Y=30 in the MAP & SVE in my previous post.

Questions:
1) Is the total score amount correct? (178 ; Is my error in the iteration thru the squares or in the final calculation?)
2)

> darkpanda said: ↑
>
> - ***Adjust the land value to the range [1..15]***

Do you mean 0 to 15?

Thanks for your assistance.

Dack, Jul 7, 2013 #9

darkpanda
Dark Prince

Joined: Oct 28, 2007
Messages: 600

> Dack said: ↑
>
> *My assumption is that **Grassland special** and **River special** refer to squares that would be special resource squares but happen to be terrain type Grassland or River. Is that correct?*

You have the wrong assumption: **grassland special** and **river special** are grassland and river squares which follow the special rule "(x+y)%4==0 || (x+y)%4==3".

> Dack said: ↑
>
> Code:
>
> ```
> +0 -2 i12  x03y28 L01 ocean     srsT S11
> +0 -2 i12  x03y28 L01 ocean     srsT S11  rT+0131
> ```

Ok, this is where the problem is: actually, CIV considers the **square at (0,-2) as an inner circle neighbour**... Again, I believe this is a programmer's mistake, just like CIV multiplies the value of North neighbour (0,-1) by 2, thinking that it is actually the central square... This is most likely a confusion regarding the array of "neighbour coordinates" that the programmer believe to start with ID=0=Central Square, whereas in fact ID=0=North neighbour (Central square is actually ID=20, the **last** map square from the neighbour array...)
Checking for "**inner circle neighbour**" is done by verifiying that **neighbour ID is strictly below 9**, which includes ID=8, which is (0,-2)...

Taking this into account, your value "11" above for the ocean with fish is multiplied by 2, and this gives the total 189 eventually, which matches your total 178 + the missing 11. The other values are correct.

Thanks for highlighting this, I didn't notice this issue before!

So the final computation is:

Code:

```
   TOTAL VALUE so far: 189 [0xBD]
   abs(value -120) -> value = 69 [0x45:0b1000101]
   abs(value/8) -> value = 8 [0x8:0b1000]
   range bound to [1..15] -> value = 8
   final compute -> value = value/2+8 -> [B]value = 12[/B]
```

> Dack said: ↑
>
> *bit clear &HFFF0 is 0007 <======= **Adjust the land value to the range [1..15]***
> ***I assume the the above is mod 16***

The actual operation is **min(max(1,landval),15)**, like pushing the value into the [1-15] range...
Once again, I realize my wording is confusing, I'll update it as well.

> Dack said: ↑
>
> *Do you mean 0 to 15?*

It's definitely **1** to 15.

Thanks again for your review,
Cheers!

JCivED - a toolbox for Sid Meier's Civilization (MS-DOS)

darkpanda, Jul 7, 2013 #10

Dack
Terra Form
Moderator

Joined: Nov 26, 2003
Messages: 530
Location: civ dos 474.05

darkpanda,
Again many thanks for the follow up on this topic. This certainly couldn&#8217;t be deduced from the out side.
Dack

> darkpanda said: ↑
>
> ***grassland special** and **river special** are grassland and river squares which follow the special rule "(x+y)%4==0 || (x+y)%4==3"*

A thread on grasslands
In my code I refer to it as **grasslands II**, another name perhaps **grasslands with shield**. The river in the similar position has no markings. I was going to raise the issue of why **grasslands II** didn&#8217;t come into play but my basic understanding was so far off the mark that it didn't seemed like a concern at the moment. But why its importance is for the center square only seems odd.

> darkpanda said: ↑
>
> *Checking for "**inner circle neighbour**" is done by verifiying that **neighbour ID is strictly below 9**, which includes ID=8, which is (0,-2)...*

A) The following is my current index numbering
B) My current understanding if the **inner circle**.
C) Perhaps you could fill in the grid with the index order use in the CIV code. Also indicate which squares are in the **inner circle**

Code:

```
         A                    B                    C

     |11|12|13|           |  | X|  |           |  |  |  |
   --+--+--+--+--       --+--+--+--+--       --+--+--+--+--
   10|04|00|05|14         | x| x| x|           |  |  |  |
   --+--+--+--+--       --+--+--+--+--       --+--+--+--+--
   09|03|20|01|15         | x|  | x|           |  |  |  |
   --+--+--+--+--       --+--+--+--+--       --+--+--+--+--
   08|07|02|06|16         | x| x| x|           |  |  |  |
   --+--+--+--+--       --+--+--+--+--       --+--+--+--+--
     |19|18|17|           |  |  |  |           |  |  |  |
```

> darkpanda said: ↑
>
> *The actual operation is **min(max(1,landval),15)**, like pushing the value into the [1-15] range...*

~~If you could describe this in a more programmatically way step by step. I am unsure of what is needed to produce the value.~~

Edit:
min(max(1,landval),15)
Unfamiliarity with the function, I missed its simplicity

```
If LV < 1 Then LV = 1
If LV > 15 Then LV = 15
```

=========================
The corrections that I made to my code have now reduced the number of mismatched values to ~~three~~ *one*. They are at the following locations in the above MAP & SVE file.

1] X027 Y016 <== working

2] X050 Y008

~~I think that each stem from my lack of understanding of how to **Adjust the land value to the range [1..15]**~~

The third one is odd in that the CIV value from MAP file is +0008 0x08.

Does the above MAP & SVE file pass your code for deriving the **Land Value** ?

Spoiler : Show

Dack, Jul 8, 2013 #11

darkpanda
Dark Prince

Joined:	Oct 28, 2007
Messages:	600

> Dack said: ↑
>
> *darkpanda,*
> *Again many thanks for the follow up on this topic. This certainly couldn’t be deduced from the out side.*
> *Dack*

No problem 🙂 I do strive for quality and exhaustiveness, unfortunately I rarely take the time for them...

I have filled in the grid below as you asked:

> Dack said: ↑
>
> *A) The following is my current index numbering*
> *B) My current understanding if the **inner circle**.*
> *C) Perhaps you could fill in the grid with the index order use in the CIV code. Also indicate which squares are in the **inner circle***
>
> *Click to expand...*

For the record, the relative coordinates of neighbour squares are stored in CIV.EXE as **2 arrays of 21 words** (2-byte values), i.e. totally 84 bytes, for all 20 "city" neighbours + 1 for the central square.

Those values can be found at offset 0x26CE5 in CIV EN 47401, and offset 0x26BB1 in CIV EN 47405:

Code:

```
00026ce5h: [COLOR="Blue"]00 00 01 00 00 00 FF FF 01 00 01 00 FF FF FF FF[/COLOR]
00026cf5h: [COLOR="Blue"]00 00 02 00 00 00 FE FF FF FF 01 00 02 00 02 00[/COLOR]
00026d05h: [COLOR="Blue"]01 00 FF FF FE FF FE FF 00 00[/COLOR] [COLOR="Red"]FF FF 00
00026d15h: [COLOR="Red"]00 00 FF FF 01 00 01 00 FF FF FE FF 00 00 02 00[/COLOR]
00026d25h: [COLOR="Red"]00 00 FE FF FE FF FF FF 01 00 02 00 02 00 01 00[/COLOR]
00026d35h: [COLOR="Red"]FF FF 00 00[/COLOR]

Decimal values:
id: 0  1  2  3   4  5  6  7     8  9 10 11   12  13  14  15  16  17  18  19  20
X:  0  1  0 -1   1  1 -1 -1     0  2  0 -2   -1   1   2   2   1  -1  -2  -2   0
Y: -1  0  1  0  -1  1  1 -1    -2  0  2  0   -2  -2  -1   1   2   2   1  -1   0
->  N  E  S  W  NE SE SW NW    NN EE SS WW  NNW NNE ENE ESE SSE SSW WSW WNW CITY
```

> Dack said: ↑
>
> ...
> *2] X050 Y008*
>
> *Code:*

> Click to expand...

Two things to mention:

- First, I made a mistake, although it doesn't affect our current discussion: the "add 1" should be removed (a misreading on my part) and algorithm operation should actually be:

  > ◦ *If the **land value was negative 3 steps before, then negate the land value again, i.e. value = -value ~~and add 1, i.e. value = 1-value~~***

- Second, as I highlighted within your quote, our computations differ at this precise point: you **didn't negate** the land value although it *was* negative 3 steps above... If you negated it, the total value would be -2 at this point, which would be changed to 1 when realigning to [1..15] (indeed, -2 < 1 == true), and then 1/2+8 = 0+8 = **8**

I must underline that the whole "absolute value" and "negate" steps are an almost literal translation of CIV's code, that I found hard to understand.

Spoiler : Show

But if we look at the actual values, **when "value - 120" is negative**, then the [1..15] step will change it to 1, and **the land value is always 8**, end of story.

I should take time to update the main algorithm and rationalize it a little, but I really can't spare time for this at the moment... Holidays ☺

JCivED - a toolbox for Sid Meier's Civilization (MS-DOS)

darkpanda, Jul 9, 2013 #12

Dack
Terra Form
Moderator

Joined: Nov 26, 2003
Messages: 530
Location: civ dos 474.05

darkpanda Thanks for the effort. The last tweak was the charm.

There seems to be one type land pattern that didn&#8217;t fit the pattern of the algorithm. I had observed it when the square being evaluated has some number of artic squares in its neighborhood. Because of the negative swing I would get plus or minus one or two when the mismatch occurred.

> simonnomis said: ↑
>
> *I can imagine that it took that amount of effort!*

Can you believe the hours and hours that we spent trying to come up with an algorithm? In the end I think we had one that was about 60% accurate. I applied it to TerraForm but only to squares that were actually changed by the user. But given the programming errors in CIV (the outlier in the inner circle and the north square instead of the center square) probably any approximation that picked out the relatively good squares would work.

Dack, Jul 9, 2013 #13

---

kirkham7
Chieftain

So with everything that has been done, what exactly do you plan on doing with all of this? You guys have spent so much time, what are you working towards?

kirkham7, Jul 9, 2013 #14

Joined: Dec 22, 2012
Messages: 288
Location: Hayward, CA

Renergy
Chieftain

Joined: Nov 2, 2010
Messages: 197
Location: Czech Republic

> kirkham7 said: ↑
>
> *So with everything that has been done, what exactly do you plan on doing with all of this? You guys have spent so much time, what are you working towards?*

I'm taking the liberty of answering - as far as I can tell, this land value is crucial for the AI to work "as usual" (especially found cities, but not only that) on custom designed maps, so I guess Dack is incorporating the described calculation into Terraform and darkpanda has already added it to JCived.

Without it, it was already possible to design custom maps, but they could only be used properly e.g. after exporting them to MAP.PIC file and starting with the "Earth" option, in which case civ would calculate the values itself...

Renergy, Jul 10, 2013 #15

---

simonnomis
Chieftain

Joined: Nov 29, 2005
Messages: 195

> Dack said: ↑
>
> *Can you believe the hours and hours that we spent trying to come up with an algorithm? In the end I think we had one that was about 60% accurate. I applied it to TerraForm but only to squares that were actually changed by the user. But given the programming errors in CIV (the outlier in the inner circle and the north square instead of the center square) probably any approximation that picked out the relatively good squares would work.*

Lol, yes, many many hours playing with numbers 🙂 I think the algorithm you implemented was much better than 60% accurate however, at least in gameplay terms. And having played around with Darkpanda's editor and assigned all kinds of values to different terrain, it really doesn't seem to matter too much, as long as it's beyond the closest city squares + orbiting '8'-value surrounding squares.

simonnomis, Jul 19, 2013 #16

Nesretnik
Chieftain

Joined: Jul 17, 2013
Messages: 29
Location: Konjscina, Croatia

> Renergy said: ↑
>
> *In my opinion, there ought to be a special sub-forum for top quality posts/threads such as this one, which I would name "**CIV internals**".*
>
> *List of threads that could go there (I'm not saying that list is complete)...*
>
> *Edit: Perhaps a better title for the subforum would be "**CIV internals & modding**"*

Renergy, you're a champion! As a new member, still feeling my way around, your post is arguably the most valuable I've come across. Your list of threads (which I've removed from the quote above to save space) is priceless! The work of people like Dack, Gowron and darkpanda is extraordinary, and has given me new zest for this great game. Yes, absolutely, I would wholeheartedly support the creation of a resource (such as a dedicated thread) for this kind of stuff, and let new members be signposted clearly towards it!

Nesretnik, Aug 6, 2013 #17

Renergy
Chieftain

Joined: Nov 2, 2010
Messages: 197
Location:

> Nesretnik said: ↑
>
> *Renergy, you're a champion! As a new member, still feeling my way around, your post is arguably the most valuable I've come across. Your list of threads (which I've removed from the quote above to save space) is priceless! The work of people like Dack, Gowron and darkpanda is extraordinary, and has given me new zest for this great game. Yes, absolutely, I would wholeheartedly support the creation of a resource (such as a dedicated thread) for this kind of stuff, and let new members be signposted clearly towards it!*

Glad it helped & welcome here, Nesretnik!

As I've written, subforum would IMO be the best. Currently, there is only one subforum, concerned with game of the month. Another option is to have a sticky post where anyone could

Czech Republic

post links to his/her favorite articles?

Well chosen title would attract in itself IMO - "CIV internals & modding" seems fine to me (either as a subforum or sticky post title).

Dack, what do you think? I do realize having to keep the subforum tidy and to the point would require additional work, but it would be worth it. Perhaps there is also a technical issue with moving pages between subforums (and breaking links), in that case I'm all for a sticky post with links.

PS: Off topic comment about Croatia and Czech Republic

Spoiler : Show

Renergy, Aug 7, 2013 #18

darkpanda
Dark Prince

Joined:	Oct 28, 2007
Messages:	600

Going further on my initial impression that there is a programmer's bug in the original CIV logic for computing land values, I decided to try out a fixed-up version of this algorithm, which basically goes as follows (differences with the original algorithm are marked in **blue**):

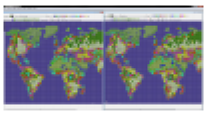
***Corrected* Map Square Land Value Calculation**

First of all, map squares for which the land value is calculated are **in the range [2,2] - [77,47]**. That is to say that the 4 rows of arctic and antarctic land (rows 0, 1, 48 and 49), as well as the 4 columns around the "TimeZone Meridian" (columns 0, 1, 78 and 79) have a land value of 0, whatever their type.

For each square within this range, the **land value is calculated as follows (initial value is 0)**:

- If the square's terrain type is **not Plains, Grassland or River**, then its land value is **0**
- Else, **for each 'city square' neighbouring the map square** (i.e. each square following the city area pattern, including the map square itself, so totally 21 'neighbours'), compute the following **neighbour value** (initially 0):
    - If the neighbour square type is **Grassland special** or **River special**, **add 2**, then **add** the non-special **Grassland or River terrain type score** to the neighbour value
    - Else **add neighbour's terrain type score** to the neighbour value
    - If the neighbour square is in the map square **inner circle**, i.e. one of the 8 neighbours immediately surrounding the map square, then **multiply the neighbour value by 2**
    - If the neighbour square is the **central square**, then **multiply the neighbour value by 4** <- *in the original algorithm, the neighbour value was multiplied by 2 if the index is 0 (north square); but from the previous instruction, it was also multiplied by 2 if the index was below 9 (inner circle, including north square too); so if the index was 0, it was multiplied by 2* ***twice****, as 0 == 0 and 0 < 9 are both true; based on the supposed confusion that index 0 was the central square in the programmer's mind, I interpret this as an intent to multiply the central square value by 2*2 =* ***4*** *eventually; in the fixed-up algorithm, the central square index is 20, but since 20 < 8 is false, the previous instruction does not multiply by 2, so we must directly multiply by 4 instead of 2*
    - **Add the neighbour's value to the map square total value** and loop to next neighbour
- After all neighbours are processed, if the *central* map square's terrain type is **non-special Grassland or River**, **subtract 16** from total land value
- **Subtract 120 (0x78)** from the total land value
- If the **land value is negative**, then the **land value is set to 8**, and the process terminates here
- Else, **divide the land value by 8**
- **Adjust the land value to the range [1..15]**, that is: if land value<1, then land value=1; if land value>15, then land value=15;
- **Divide the land value by 2**
- And finally, **add 8** to the land value

In the current DEV version of JCivED, I added the possibility to run either version of the algorithm, and then did a test run to compare the impact on the EARTH map. Results are shown in the picture below (using MAP.PIC, i.e. without any special resource):



Globally, the impact is not very visible, but getting down in the details, it seems like the fixed-up algorithm tends to create higher values, all the while some squares still get lower values than before.

On the EARTH map, the total world value (sum of all square values) is 11151 with the original CIV algorithm, while it is 11333 for the fixed-up algorithm, which represents less than a 2% increase in value.

This can also be **patched directly into CIV.EXE**, as follows:

Code:

```
offsets:
  EN 47401: [b]0x35456[/b]
  EN 47403: [b]0x34C7C[/b]
  EN 47404: [b]0x34C7C[/b]
  EN 47405: [b]0x3549C[/b]
  EN 47501: [b]0x35456[/b]
  FR 47405: [b]0x3687C[/b]

original bytes: [color=red]83 7E C0 09  7D 06 8B 46  FC 01 46 FC  83 7E C0 00
                75 06 8B 46  FC 01 46 FC  8B 46 FC [/color]
 patched bytes: [color=blue][b]8B 46 FC 83  7E C0 08 7D  03 D1 E0 90  83 7E C0 14
                75 06 B9 04  00 F7 E9 90  90 90 90 [/b][/color]
```

JCivED - a toolbox for Sid Meier's Civilization (MS-DOS)

darkpanda, Oct 23, 2013 #19

(You must log in or sign up to reply here.)

Forum software by XenForo™ ©2010-2016 XenForo Ltd. | Ads by Curse | ©2000-2019 CivFanatics Terms and Rules