旷野大计算

出题人：清华大学 吕凯风
验题人：清华大学 王逸松 于纪平 张瑞喆

July 26, 2016

闲扯

大家好！我 vfk 又回来了！

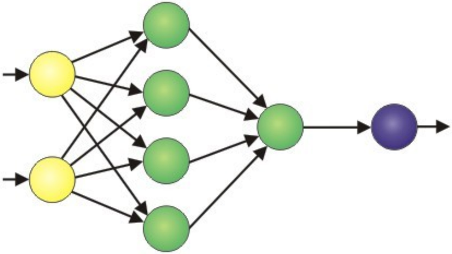
这次给大家出了一道旷野大计算，不知道大家有没有旷野大得分呀？

（听说跳蚤国已占领 NOI 题面了，但我们确实是分开写的题面 233）

题目大意

[1] 图片来源：电影《赤壁》

题目大意



[1]图片来源: http://logicalgenetics.com/assorted/recurrent-nns/feedforward-small.jpg

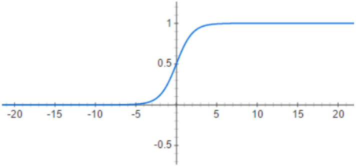
题目大意

名称	操作符	计算结果	特点
输入节点	I	输入	基本的输入输出
输出节点	O	输出	
加法节点	+	$x_t = x_i + x_j$	加减法
偏移节点	C	$x_t = x_i + c$	
取反节点	-	$x_t = -x_i$	
左移节点	<	$x_t = x_i \cdot 2^k$	二的幂次的乘除
右移节点	>	$x_t = x_i / 2^k$	
S 型节点	S	$x_t = s(x_i)$	一个非线性操作
比较节点	P	$x_t = \mathrm{cmp}(x_i, x_j)$	功能强大但要倒扣分
Max 节点	M	$x_t = \max(x_i, x_j)$	
乘法节点	*	$x_t = x_i \cdot x_j$	

题目大意

Sigmoid 函数：

$$s(x) = \frac{1}{1+e^{-x}}$$



[1] 图片来源：Google 搜索的截图

题目大意

给定的 10 个计算任务。

对于每个任务，用一些基本的操作，构造一个计算网络。

使用的节点数越少越好。

旷野大得分

测试点编号	最高分	平均分
1	10	9.7
2	10	9.6
3	10	8.4
4	9	5.6
5	10	9.1
6	9	2.3
7	10	1.3
8	10	4.5
9	10	1.2
10	10	0.4

旷野大得分

最高分：

旷野大得分

最高分：97 分（袁宇韬）

旷野大得分

最高分：97 分（袁宇韬）

平均分：

旷野大得分

最高分：97 分（袁宇韬）

平均分：52.1 分

旷野大得分

最高分：97 分（袁宇韬）

平均分：52.1 分

中位数：

旷野大得分

最高分：97 分（袁宇韬）

平均分：52.1 分

中位数：52 分

旷野大得分

最高分：97 分（袁宇韬）

平均分：52.1 分

中位数：52 分

众数：

旷野大得分

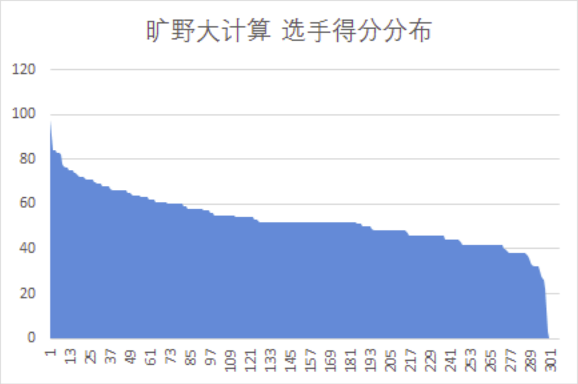
最高分：97 分（袁宇韬）

平均分：52.1 分

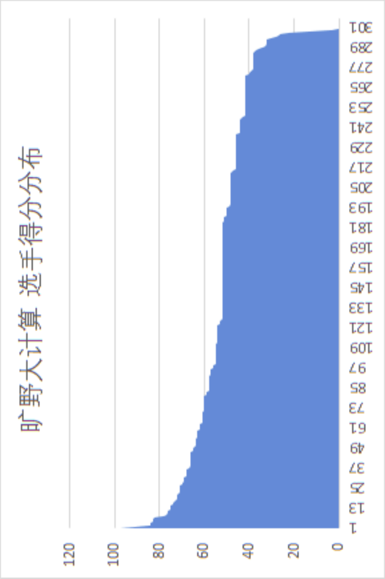
中位数：52 分

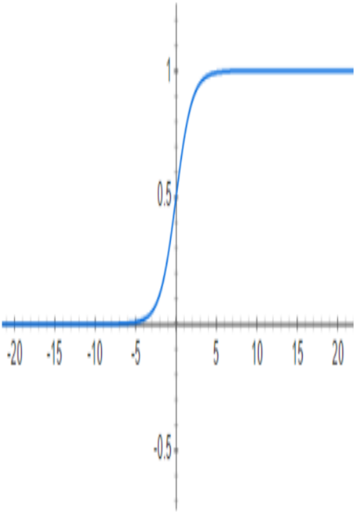
众数：52 分

旷野大得分的大秘密


旷野大计算 选手得分分布
120
100
80
60
40
20
0
1
13
25
37
49
61
73
85
97
109
121
133
145
157
169
181
193
205
217
229
241
253
265
277
289
301

旷野大得分的大秘密


旷野大计算 选手得分分布
0
20
40
60
80
100
120
1
13
25
37
49
61
73
85
97
109
121
133
145
157
169
181
193
205
217
229
241
253
265
277
289
301


1
0.5
-0.5
-20
-15
-10
-5
5
10
15
20

旷野大吐槽

编号	输入	输入限制	输出
1	a, b	$\lvert a\rvert, \lvert b\rvert \le 10^9$ 小数部分不超过 9 位	$-2a-2b$
2	a	$\lvert a\rvert \le 10^9$ 小数部分不超过 9 位	$\frac{1}{1+e^{17a}}$
3	a	$\lvert a\rvert \le 10^9$ 小数部分不超过 9 位	$\begin{cases} -1 & a<0 \\ 0 & a=0 \\ 1 & a>0 \end{cases}$
4	a	$\lvert a\rvert \le 10^9$ 小数部分不超过 9 位	$\lvert a\rvert$，即 a 的绝对值
5	$a_1, \ldots, a_{32}$	$a_1, \ldots, a_{32} \in \{0, 1\}$	把 $a_1, \ldots, a_{32}$ 从左到右看成一个二进制整数，高位在左低位在右，输出该整数的值
6	a	$0 \le a < 2^{32}$ a 为整数	输出 32 个整数，从高位到低位输出 a 的二进制表示（不足 32 位的在高位补 0）
7	a, b	$0 \le a, b < 2^{32}$ a, b 均为整数	a, b 按位异或的结果
8	a	$\lvert a\rvert \le 10^9$ 小数部分不超过 9 位	$\frac{a}{10}$
9	$a_1, \ldots, a_{16}$	$\lvert a_1\rvert, \ldots, \lvert a_{16}\rvert \le 10^9$ 小数部分不超过 9 位	输出 16 个实数，表示 $a_1, \ldots, a_{16}$ 从小到大排序后的结果
10	a, b, m	$0 \le a, b < 2^{32}$ $1 \le m < 2^{32}$ a, b, m 均为整数	$a \cdot b$ 除以 m 的余数

测试点 1 ——熟悉基础操作

输入 a, b，计算 $-2a-2b$。

5 分解法

$$-(a<<1)+(-(b<<1))$$

共 8 次操作。

10 分解法

$$-((a+b)<<1)$$

共 6 次操作。

测试点 2 ——熟悉基础操作

输入 a，计算 $\frac{1}{1+e^{17a}}$。

4 分解法

$$s(-(a+a+a+\cdots+a))$$

共 20 次操作。

10 分解法

$$s(-(a+(a<<4)))$$

共 6 次操作。

测试点 3 ——判断正负

输入 a，计算

$$\operatorname{sgn}(a) = \begin{cases} -1 & a < 0, \\ 0 & a = 0, \\ 1 & a > 0. \end{cases}$$

6 分解法

使用扣分操作。先 $a >> 1000$ 构造出 0，再和 a 一起扔进比较节点。

10 分解法

$$s(x << 500) \approx p(x) = \begin{cases} 0 & x < 0, \\ 1/2 & x = 0, \\ 1 & x > 0. \end{cases}$$

把 $p(a)$ 平移放缩即可得到 sgn。

测试点 4 ——取绝对值

输入 a，计算 $|a|$。

2 分解法

使用扣分操作，利用比较节点，算出 $\mathrm{sgn}(a)$，然后使用乘法节点计算 $|a| = a \cdot \mathrm{sgn}(a)$。

6 分解法

使用扣分操作，利用测试点 3 算出 $\mathrm{sgn}(a)$，然后使用乘法节点计算 $|a| = a \cdot \mathrm{sgn}(a)$。
或者直接使用 Max 节点也能计算出绝对值。

测试点 4 ——取绝对值

10 分解法

思考：再套一层 s 呢？考虑一个 x，

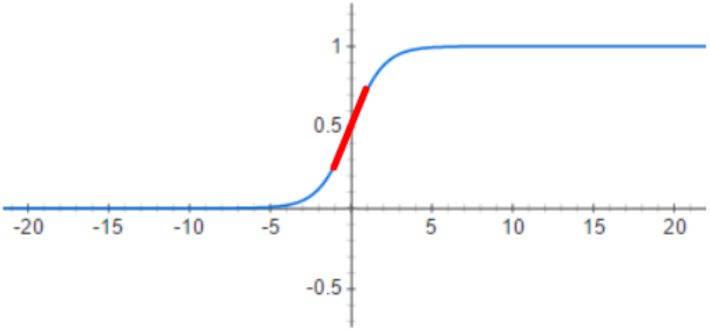
$$s(x + (p(x) << 500)) \approx \begin{cases} s(x) & x < 0, \\ 1 & x \geq 0. \end{cases}$$

如果不是 $s(x)$ 而是 x 就很妙了！

测试点 4 ——取绝对值

10 分解法

无穷小的时候会发生什么？



测试点 4 ——取绝对值

10 分解法

由导数的定义知，当 x 很接近于 0 的时候，

$$\frac{s(x)-s(0)}{x-0} \approx s'(0) = \frac{1}{4}.$$

所以，如果 x 很小，我们就能通过 $s(x)$ 解出 x。

测试点 4 ——取绝对值

10 分解法

$$s((x>>150)+(p(x)<<500)) \approx \begin{cases} 1/2+(x>>154) & x<0, \\ 1 & x \geq 0. \end{cases}$$

然后就能构造出 $\min(x,0)$，于是也就能得到 $|x|$ 了。

测试点 4 ——取绝对值

10 分解法

进行一些必要的卡常数可以得到 10 分。

代码

```
x = in();
p = s((x + 1e-30) << 500) << 152;
y = s((x >> 150) + p);
r = x + ((-y + 0.5) << 153) + p
out(r);
```

共 14 次操作。

测试点 5 ——二进制转换

输出 $a_1, \ldots, a_{32}$ 表示的二进制整数。

10 分解法

直接使用加法和乘法就行，共 95 次操作。

测试点 6 ——二进制转换

输出整数 $a \in [0, 2^{32})$ 的二进制表示。

基本思路

```
a = in();
for (i = 31; i >= 0; i--)
{
    if (x >= (1 << i))
        b[i] = 1;
    else
        b[i] = 0;
    x = x + (-(b[i] << i));
}
out(b);
```

测试点 6 ——二进制转换

6 分做法

使用比较节点水过。

9 分做法

利用如前所述的 $p(x)$ 函数实现比较节点，进行一些必要的卡常数。例如使用 $x << 500$ 代替 x 进行计算，因为反正在计算 p 函数时也得左移。

10 分做法

注意到 $i = 0$ 的时候直接令 $b[0] = x$ 就行了，共 190 次操作。

前 6 个测试点小结

前 6 个测试点都是些非常基本的操作。

把这些基本操作玩清楚，后面 4 个较为高级的算法的测试点也就不难了。

（而且后面的评分参数都卡得很松松松）

测试点 7 ——二进制运算

输入 a, b，计算 $a \operatorname{xor} b$。

首先测试点 5, 6 要玩出比较好的结果。进行二进制转换后，只用考虑两个 01 变量 x, y 的异或。

10 分解法

```
s = x + y;
return s + (-p(s + (-1.5)) << 1);
```

共 603 次操作。

验题人 saffah 卡出了 539。

测试点 8 ——一般常数的除法

输入 a，计算 $a/10$。

6 分解法

使用扣分操作。先 $x >> 1000$ 构造出 0，再加上 0.1 得到 0.1，使用乘法节点计算 $0.1x$。

得分不明的解法

可以把 a 乘以 2^{32} 之后转为整数，然后敲个二进制除法。

7 到 8 分解法

把 0.1 化为二进制，变为若干个 x 右移再相加的式子。

9 分解法

在前一个算法的基础上用减法卡卡，能去掉二进制表示中连续的一段 1。或者使用“循环之美”大法，算出一个循环节然后倍增。

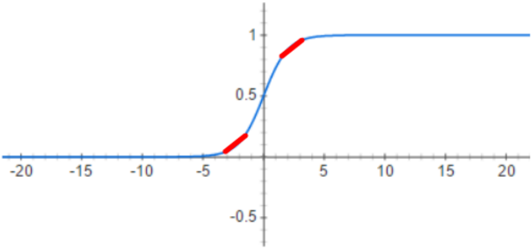
测试点 8 ——一般常数的除法

10 分解法

由导数的定义知，当 x 很接近于 x_0 的时候，

$$\frac{s(x)-s(x_0)}{x-x_0}\approx s'(x_0).$$

解出一个 x_0 使得 $s'(x_0)=0.1$ 就行了！只需 7 次操作即可解决问题。由于只用 10^{-9} 的精度，可以用 `double` 算出 x_0，然后扔进 `checker` 算出高精度的 $s(x_0)$。



测试点 9 ——排序

输入 $a_1, \ldots, a_{16}$，将其进行排序。

这个计算任务很能说明该计算网络的局限性，只有冒泡排序、选择排序这样的才适合运行。当然复杂度更低的有 Bitonic Sort 之类，然而本题中使用冒泡排序就已经足够了。

关键在于比较器的实现，下面是一个例子：

10 分解法

```
s = x[i] + x[j];
x[i] = x[i] + min(-x[i] + x[j], 0);
x[j] = s + -x[i];
```

共 2072 次操作，具体 $\min(x, 0)$ 的实现可参照测试点 4。

测试点 10 ——模意义下的乘法

输入 a, b, m，计算 $a \cdot b \bmod m$。

10 分解法

实现一个“快速加”，就是把快速幂里的乘法改成加法后的产物，需要稍微卡一卡常数。
只需实现一个 $[0, 2m)$ 之间的数对 m 取模。

这样已经可以获得满分了，但可以操作数更少吗？

测试点 10 ——模意义下的乘法

更厉害的 10 分解法

由泰勒展开：

$$s(x+\ln 3)=\frac{1}{4}+\frac{3}{16}x+\frac{3}{64}x^2-\frac{1}{256}x^3-\frac{5}{1024}x^4+O(x^5)$$

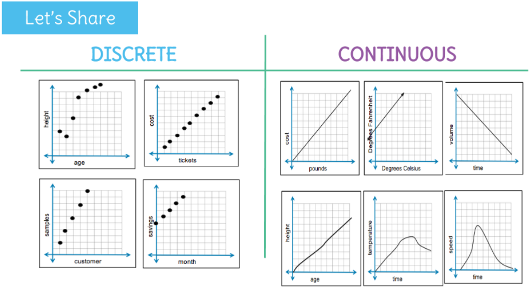
由此可构造 x^2，进而得到 xy 的构造，再仿照把整数化二进制的方法实现个取模就行了。

可以做到只用 819 个操作。

启示：模块化

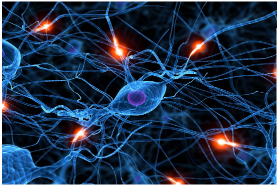
[1] 图片来源：http://a3.att.hudong.com/24/15/19300001310069131044151722588.jpg

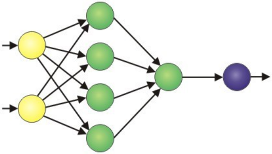
启示：离散与连续各有千秋

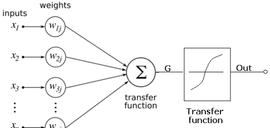


1 图片来源：https://www.filepicker.io/api/file/G4Cob0O0RjiKchLVPHWw?policy=eyJleHBpcnkiOjQ1NTg5ODY5NjAsImNhbGwiOiJyZWFkIn0%3D&signature=99571cbec16a027370fcedc747f069e0a82d6eea8c103a413c3040dff6e5b55a

启示：神经网络







[1]图片来源：`http://www.extremetech.com/wp-content/uploads/2013/09/340.jpg`

[2]图片来源：`http://www.global-warming-and-the-climate.com/images/Neuron-input.gif`

讲完啦！

完成一项人生成就：在 NOI 中出题！

感谢 CCF 提供学习和交流的平台。

感谢验题并加强评分参数的王逸松大爷，于纪平大爷。

感谢斟酌题面的张地主。

谢谢大家！