

1 武器

编号 type	名称 name	价值 value	射击速度 shoot_speed	子弹速度 pill_speed	穿透陨石数 hurt_count	伤害方式 use
0	手枪	5	18	2×10^6	1	<code>x-=static_cast<intmp_t>(3*power_rate_pill* power_rate_meteorite);</code>
1	机关枪	10	4	4×10^6	1	<code>x-=static_cast<intmp_t>(1*power_rate_pill* power_rate_meteorite);</code>
2	大炮	15	60	1.7×10^6	1	<code>x-=static_cast<intmp_t>(50*power_rate_pill* power_rate_meteorite);</code>
3	小型激光枪	30	5	∞	1	<code>x-=static_cast<intmp_t>(1*power_rate_pill* power_rate_meteorite);</code>
4	中型激光枪	50	5	∞	3	<code>x-=static_cast<intmp_t>(2*power_rate_pill* power_rate_meteorite);</code>
5	大型激光枪	80	50	∞	20	<code>x-=static_cast<intmp_t>(100*power_rate_pill* power_rate_meteorite);</code>
6	连续型激光枪	80	0	∞	20	<code>x-=static_cast<intmp_t>(3*power_rate_pill* power_rate_meteorite);</code>
7	衰变之枪	75	20	2×10^6	1	<code>x=static_cast<intmp_t>(exp(log(floatmp_t(x))-log(floatmp_t(1.2)))*power_rate_pill* power_rate_meteorite));</code>
8	强化的衰变之枪	95	20	2×10^6	1	<code>x=static_cast<intmp_t>(exp(log(floatmp_t(x))-log(floatmp_t(1.7)))*power_rate_pill* power_rate_meteorite));</code>
9	二分之枪	120	20	2×10^6	1	<code>x=static_cast<intmp_t>(exp(log(floatmp_t(x))-log(floatmp_t(2))*power_rate_pill*power_rate_meteorite));</code>
10	开方之枪	200	40	3×10^6	2	<code>x=static_cast<intmp_t>(pow(floatmp_t(x),1/(2* power_rate_pill*power_rate_meteorite))));</code>

(续表)

编号 type	名称 name	价值 value	射击速度 shoot_speed	子弹速度 pill_speed	可击中的陨石数 hurt_count	伤害方式 use
11	对数之枪	400	60	5×10^5	10	<code>x=static_cast<intmp_t>(log(floatmp_t(x))/log(floatmp_t(2)))</code>
12	反向调分之枪	100	10	2×10^6	1	<code>x=x*x/y;</code>
13	随机打乱之枪	20	5	1×10^7	1	<code>auto tmp=x.str();std::shuffle(tmp.begin(),tmp.end(),rand64);x=intmp_t(tmp.substr(rand64()%10?0:1));</code>

2 效果

编号 type	名称 name	价值 value	持续时间 time	接受者 reciever	编号 detail	效果	图片资源
0	快速射击 I	10	1500	当前武器	0	射击速度 $\times 0.7$	0
1	快速射击 II	15	1500	当前武器	1	射击速度 $\times 0.5$	0
2	快速射击 III	20	1500	当前武器	2	射击速度 $\times 0.2$	0
3	快速射击 IV	20	1500	所有武器	1	射击速度 $\times 0.5$	0
4	快速射击 V	40	1500	所有武器	2	射击速度 $\times 0.2$	0
5	保护盾 I	25	500	行星	0	行星被击中时, 完整度不下降	1
6	保护盾 II	60	1500	行星	0	行星被击中时, 完整度不下降	1
7	保护盾 III	100	3000	行星	0	行星被击中时, 完整度不下降	1
8	修补 I	15	1	行星	1	<code>health_add=100</code>	4
9	修补 II	20	1	行星	2	<code>health_add=500</code>	4
10	修补 III	30	1	行星	3	<code>health_add=1000</code>	4
11	修补 IV	50	1	行星	4	<code>health_mul=2</code>	4
12	修补 V	80	1	行星	5	<code>health_mul=5</code>	4
13	冻结 I	17	250	陨石	0	陨石速度 $\times 0.5$	5
14	冻结 II	31	250	陨石	1	陨石速度 $\times 0.1$	5

(续表)

编号 type	名称 name	价值 value	持续时间 time	接受者 reciever	编号 detail	效果描述	图片资源
15	冻结 III	50	250	陨石	2	陨石速度 $\times 0$	5
16	冻结 IV	60	500	陨石	2	陨石速度 $\times 0$	5
17	速捷 I	10	1500	玩家	0	移动速度 $\times 1.5$	9
18	速捷 II	15	1500	玩家	1	移动速度 $\times 2$	9
19	饱和	20	1500	玩家	2	饥饿值停止下降	6
20	坚硬 I	19	1500	补给箱	0	power_rate=0.5	8
21	坚硬 II	26	1500	补给箱	1	power_rate=0.1	8
22	坚硬 III	38	1500	补给箱	2	hurt_by_weapon=false	8
23	生命恢复 I	18	250	行星	6	health_add=1	2
24	生命恢复 II	24	250	行星	7	health_add=3	2
25	生命恢复 III	36	250	行星	8	health_add=6	2
26	生命恢复 IV	68	250	行星	9	health_mul=1.004	2
27	生命恢复 V	89	250	行星	10	health_mul=1.009	2
28	清理 I	40	500	陨石	3	power_rate=5,speed_rate=0.8	7
29	清理 II	47	500	陨石	4	kill_after_shooted=true, speed_rate=0.8	7
30	清理 III	64	50	陨石	5	health_mul=0.9,health_add=-1	7
31	无限 I	20	1500	当前武器	3	射击后子弹不减少	10
32	无限 II	60	1500	所有武器	3	射击后子弹不减少	10
33	穿透 I	13	1500	当前武器	4	infinite_hurt_count=true	3
34	穿透 II	32	1500	所有武器	4	infinite_hurt_count=true	3
35	清理 IV	72	瞬时	陨石	-	box_and_mete_left= meteorites_list.size(); meteorites_list.clear();	7

3 食物

编号 type	名称 name	价值 value	恢复的饥饿值 add_hunger	编号 type	名称 name	价值 value	恢复的饥饿值 add_hunger	编号 type	名称 name	价值 value	恢复的饥饿值 add_hunger
0	糖果	3	4000	1	面包	4	8000	2	牛排	6	15000
3	数学分析教程	30	2147483647								

4 陨石

编号 type	下落时间 fly_time	强度 strength	大小 size	图片资源	伤害 hurt
0	400-500	5	5×10^6	0	health-=static_cast<intmp_t>(100*hurt_rate_planet* hurt_rate_meteorite)*(is_neg?-1:1);
1	400-500	10	5×10^6	0	health-=static_cast<intmp_t>(150*hurt_rate_planet* hurt_rate_meteorite)*(is_neg?-1:1);
2	200-250	8	4×10^6	0	health-=static_cast<intmp_t>(80*hurt_rate_planet* hurt_rate_meteorite)*(is_neg?-1:1);
3	200-250	3	5×10^6	3	health-=static_cast<intmp_t>(150*hurt_rate_planet* hurt_rate_meteorite)*(is_neg?-1:1);
4	1380-1500	100	6.8×10^6	2	health-=static_cast<intmp_t>(1000*hurt_rate_planet* hurt_rate_meteorite)*(is_neg?-1:1);
5	150-170	10	3×10^6	4	health-=static_cast<intmp_t>(30*hurt_rate_planet* hurt_rate_meteorite)*(is_neg?-1:1);
6	1800-1900	1000	9×10^6	5	health=intmp_t(exp(log(floatmp_t(health))-log(floatmp_t>(3)))*complete_rate*hurt_rate_planet* hurt_rate_meteorite*(is_neg?-1:1)));

(续表)

编号 type	下落时间 fly_time	强度 strength	大小 size	图片资源	伤害 hurt
7	1800-1900	2 ⁸⁰	8.5 × 10 ⁶	6	<pre>double tmp=complete_rate*hurt_rate_planet* hurt_rate_meteorite;if(!is_neg){if(tmp>0.7)health= intmp_t(log(floatmp_t(health))/log(2));else if(tmp >0.3)health=intmp_t(sqrt(floatmp_t(health)));else health/=2;}</pre>
8	500-600	30	4 × 10 ⁶	7	<pre>health-=static_cast<intmp_t>(500*complete_rate* hurt_rate_planet*hurt_rate_meteorite)*(is_neg?-1:1) ;</pre>
9	900-1000	70	5 × 10 ⁶	8	<pre>health=intmp_t(exp(log(floatmp_t(health))-log(floatmp_t(10))*complete_rate*hurt_rate_planet* hurt_rate_meteorite*(is_neg?-1:1)));</pre>
10	1300-1400	5000	6 × 10 ⁶	9	<pre>if(is_neg){auto s=health.str();std::shuffle(s.begin (),s.end(),rand64);health=intmp_t(s);}else{auto s= health.str();std::shuffle(s.begin(),s.end(),rand64) ;health=intmp_t(s.substr(rand64()%2?1:0));}</pre>
11	1700-1800	100	7 × 10 ⁶	10	<pre>health-=static_cast<intmp_t>(10/complete_rate* hurt_rate_planet*hurt_rate_meteorite)*(is_neg?-1:1) ;</pre>

5 补给箱

编号 type	下落时间 fly_time	强度 strength	大小 size	图片资源	价值 total_value	内容物 hurt
0	200-300	1	3.7 × 10 ⁶	1	0-0	子弹 ×30
1	500-550	15	2.3 × 10 ⁶	0	17-19	子弹、食物 0（糖果） 效果 0（快速射击 I）

(续表)

编号 type	下落时间 fly_time	强度 strength	大小 size	图片资源	价值 total_value	内容物 hurt
2	50-100	5	5.0×10^6	0	0	效果 15 (冻结 III)
3	100-150	20	4.0×10^6	0	15-20	效果 8 (修补 I)、效果 9 (修补 II)
4	100-150	20	5.0×10^6	0	40-50	糖果 ×5、面包、子弹
5	200-200	10	5.0×10^6	0	15-36	效果 9、效果 24、效果 23
6	100-200	10	5.0×10^6	0	0	武器 12
7	150-250	10	6.0×10^6	0	40-50	效果 19、效果 2、效果 28