基本資料

使用前成績使用後成績		使用後成績	Silver 965	
	上 夕 力 井 回		Sliver 903	
	[石 次 截 回			
Spider	Attack 🏋 Silver League 💽 Rank 965 / 2,186		(o) <u>1</u>	
	CodinGame	Rey = 2	с 🗆	
5		1 // 408231171 7 2 #include <iost 3 #include <stri< td=""><td></td></stri<></iost 		
← BACK	SPRINI	4 #include < 4 #include <<<a _y7uua5w<="" by="" hammel:="" href="</td><td>or)</td></tr><tr><td></td><td></td><td>6 #include <cmat 7 #include ini 8 #include <map> x:17142</td><td><u>s</u></td></tr><tr><td>FORUM</td><td>CHALLENG</td><td>9 #include 10 11 using namespac</td><td></td></tr><tr><td></td><td>ZOPP .</td><td>12
13 class Entity
14 {</td><td></td></tr><tr><td>LAST BATTLES</td><td></td><td></td><td></td></tr><tr><td></td><td>transfer of the manufaction of the transfer of</td><td>17 int x_;
18 int y_;
19 int id_;</td><td></td></tr><tr><td>₩
HISTORY</td><td></td><td>20 int type_; 21 int shield 22 int is_con</td><td></td></tr><tr><td></td><td>V</td><td>23 int health
24 int vx_;</td><td></td></tr><tr><td>₩</td><td>Starter Kit</td><td>25 int vy_;
26 int nearBa</td><td></td></tr><tr><td></td><td>arter Als are available in the Starter Kit. They can help yo
ur own bot.</td><td>u get started with 27 int threat 28 Entity(con 29</td><td>For_;
st int &id, const int &type, const int &x, const int &y, const</td></tr><tr><td></td><td></td><td>30 public:
31 Entity();</td><td></td></tr><tr><td>SETTINGS Intro</td><td></td><td>33 const int</td><td>st int &x, const int &y);
&get_X() const { return x_; };</td></tr><tr><td>870.00</td><td>ductory video by Mathis Hammel: https://youtu.be/MyHjWftmMft
d to Silver" https:="" mathis="" td="" video="" youtu.be=""><td>ANSC 35 const int</td><td><pre>&get_Y() const { return y_; }; &get_ShieldLife() const { return shield_life_; };</pre></td>	ANSC 35 const int	<pre>&get_Y() const { return y_; }; &get_ShieldLife() const { return shield_life_; };</pre>
			<pre>aget_10() const { return id_; };</pre> <pre>r overloading for computing distance between two entities</pre>	

量化指標

		1	
大項	細項指標	同意打	總評
指標		$\sqrt{}$	(1-5)
易讀性	類別拆解方式(有哪些類別)符合直覺。	V	4
	類別、變數與函式名稱清楚,符合直覺。	V	
	單個函式內的程式碼量適中、功能單一明確。	V	
	避免使用難推測其意義的字面常數。		
	避免使用特別炫技且難以理解的程式碼。	√	
	程式碼使用一致的風格與模式。	√	
	程式碼結構儘量扁平。	√	
	在有需要的地方提供適量清楚的註解。		
使用性	提供足量、適切且容易選取的(成員)函式以達任務需求。	√	5
	函式參數個數適中、易於分辨其意義。	√	
	容易除錯:避免使用全域變數。	√	
	容易除錯:善用 const/constexpr。	√	
擴 修 充 改	適時使用迴圈與陣列,擴充資料量時不需修改太多程式碼。	√	4
	善用具名常數,易於調整數值。		<u> </u>

111-2 進階程式設計 Class Design: A Practice 評量表

(成員)函式:實作與介面切割清楚,修改時不會牽動過多其它	\checkmark	
程式碼。		
(成員)函式:功能單一明確,修改時不會牽動過多其它程式	√	
碼。		

質性意見

這個函式庫的文件很清楚,都在 github 上,變數與函式的命名方式也符合直覺,易於理解。 函式內的程式碼量不會太多,閱讀起來也算容易,功能單一明確,程式碼風格一致

但在一些地方,如下圖,程式碼中直接使用數值,例如 1280、2200 等。這些數值應該用具 名常數替代,才方便做調整。

```
bool canWind(const Hero &a, const Entity &b)
{
    return ((a - b) < 1280 && my_mana > 10 && b.get_ShieldLife() == 0);
}
bool canControl(const Hero &a, const Entity &b)
{
    return ((a - b) < 2200 && my_mana > 10 && b.get_ShieldLife() == 0);
}
bool canShield(const Hero &a, const Entity &b)
{
    return ((a - b) < 2200 && my_mana > 10 && b.get_ShieldLife() == 0);
}
```

還有註解可以再多一些