

APPRENDRE LES FONDAMENTAUX DE LA PROGRAMMATION A TRAVERS UN JEU

- SUPPORT DE COURS POUR LES PROFESSEURS •

L'OBJECTIF DE CE TUTORIEL EST DE :

1. VOUS PERMETTRE D'ENSEIGNER LES FONDAMENTAUX DE LA PROGRAMMATION **EN UNE HEURE**, VIA UN **JEU**
2. DE MONTRER AUX JEUNES GÉNÉRATIONS QU'IL EST POSSIBLE DE CRÉER DES CHOSES INCROYABLES AVEC SEULEMENT **QUELQUES LIGNES DE CODE** !

The screenshot shows the CodinGame interface with the following sections:

- Onboarding:** A game environment showing a central player unit surrounded by five enemy units labeled Whacker, Zap, Charger, Buzz, and DangerDart. The player has a score of 6000, 5 enemies in zone, 6 enemies killed, and a threat level of 02. The background is dark with glowing orange particles.
- Editeur de code (Python3):** A code editor window containing the following Python code:

```
1 import sys, math
2
3 # Auto-generated code below aims at helping you parse
4 # the standard input according to the problem statement.
5
6
7 # game loop
8
9 import sys, math
10
11 # game loop
12 while 1:
13     enemy1 = input()
14     dist1 = int(input())
15     enemy2 = input()
16     dist2 = int(input())
17
18     if dist1 < dist2:
19         print(enemy1)
20     else:
21         print(enemy2)
```
- Sortie console:** A terminal window showing game information:

```
Game information:
You have been targeted
Threats within range:
Zap 2m
Charger 3m
Standard Output Stream:
> Zap
```

Game information:
- Jeu de tests:** A test results section with two entries:

Test ID	Description	Status
01	Cinq ennemis	Pass (Yellow)
02	Danger imminent	Pass (Green)

Buttons include "LANCER TOUS LES TESTS" and "ENVOYER". A comment "how is that possible" is visible at the bottom right.

COMMENT FONCTIONNE LE JEU ?

L'OBJECTIF EST D'ÉCRIRE UN PROGRAMME QUI PERMETTE DE CONTRÔLER UN CANON SPATIAL. LE CANON DOIT DÉTRUIRE LES VAISSEAUX ENNEMIS QUI VOUS ATTAQUENT PAR VAGUES.

QUELLES SONT LES RÈGLES ?

LE CANON DOIT :

- ★ LIRE LE NOM DU 1^{er} ET DU 2^e VAISSEAU ENNEMI, AINSI QUE LA DISTANCE À LAQUELLE ILS SE TROUVENT.
- ★ SI LE PREMIER VAISSEAU ENNEMI EST PLUS PRÈS, LE CANON DOIT LUI TIRER DESSUS. SI NON, LE CANON DOIT TIRER SUR LE 2^e ENNEMI.
- ★ LE CANON DOIT CONTINUER À FAIRE CELA JUSQU'À CE QUE TOUS LES ENNEMIS AIENT ÉTÉ DÉTRUITS (CONCEPT DE BOUCLE DE JEU).

1. COMMENCEZ PAR INVITER VOS ÉTUDIANTS À SE CONNECTER À L'IDE DU ONBOARDING, EN MODE DEBUG, ET EXPLIQUEZ-LEUR LE CONCEPT DU JEU :

The screenshot shows the CodinGame IDE interface. The top left features a dark-themed background with the word "ONBOARDING" in large, glowing yellow letters. A small "WELCOME TO THE..." text box is visible in the top left corner. The top right contains a code editor window titled "Code editor (Python3)" showing the following code:

```
1 import sys, math
2
3 # Auto-generated code below aims at helping you parse
4 # the standard input according to the problem statement.
5
6
7 # game loop
8 while 1:
9     enemy1 = input()
10    dist1 = int(input())
11    enemy2 = input()
12    dist2 = int(input())
13
14    # Write an action using print
15    # To debug: print("Debug messages...", file=sys.stderr)
16
17    print("HotDroid")
```

The bottom left panel is labeled "Console output" and is currently empty. The bottom right panel is labeled "Test cases" and lists four test cases:

Case	Description	Status
01	Five enemies	Play
02	Imminent danger	Play
03	Surrounded	Play
04	Ron's nest	Play

A "PLAY ALL TESTS" button is located above the test cases. The bottom left corner of the slide features the CodinGame logo.

2. UNE FOIS QU'IL SONT FAMILIARISÉS AVEC LE JEU, VOUS POUVEZ COMMENCER À EXPLIQUER A VOS ÉTUDIANTS LES DIFFÉRENTES RÈGLES, UNE PAR UNE, ET PRÉCISER COMMENT ELLES SE TRADUISENT EN TERME DE CODE (EXEMPLE EN PYTHON 3)

1

```
enemy1 = input()  
dist1 = int(input())  
enemy2 = input()  
dist2 = int(input())
```

LIRE LE NOM ET LA DISTANCE DES DEUX PLUS PROCHES VAISSEAUX

```
enemy1 = input()  
dist1 = int(input())  
enemy2 = input()  
dist2 = int(input())
```

```
if dist1 < dist2:  
    print(enemy1)  
else:  
    print(enemy2)
```

AJOUTEZ LA
COMPARAISON DE
LA DISTANCE ET
L'ACTION

CONSEIL : GRÂCE AU MODE DEBUG, VOUS POUVEZ UTILISER LE NOM DES VAISSEAUX POUR EXPLIQUER LE CONCEPT DE « **IF THEN ELSE** ».

Par exemple : **IF** :« HOTDROID_DISTANCE » > « BUZZ_DISTANCE »

THEN: TARGET« BUZZ »

ELSE : TARGET« HOTDROID »

3

while 1:

```
enemy1 = input()
dist1 = int(input())
enemy2 = input()
dist2 = int(input())
if dist1 < dist2:
    print(enemy1)
else:
    print(enemy2)
```

**TERMINEZ AVEC LA
« BOUCLE DE JEU »**

**3. MAINTENANT, VOUS POUVEZ JOUER AVEC VOS ÉTUDIANTS EN LEUR FAISANT
PASSER LES DIFFÉRENTS JEUX DE TESTS EN MODE NORMAL :**

- 
1. UN ENNEMI
 2. DANGER IMMINENT
 3. ENCERCLÉ
 4. ATTAQUE D'ABEILLES

QU'EST-CE QUE VOS ÉTUDIANTS ONT APPRIS ?



1. QU'ILS PEUVENT FAIRE DES CHOSES **INCROYABLES** AVEC QUELQUES BOUTS DE CODE

2. UNE SÉRIE DE CONCEPTS FONDAMENTAUX :

- LE CONCEPT IN/OUT
- LE CONCEPT « IF THEN ELSE »
- COMMENT ECRIRE UNE VARIABLE
- COMMENT FAIRE UNE **BOUCLE** DE JEU

MERCI DE FAIRE PASSER LE MOT QUE CODER EST UN JEU!

L'EQUIPE CODINGAME
contact@codingame.com