

# Co Inter Math-Info

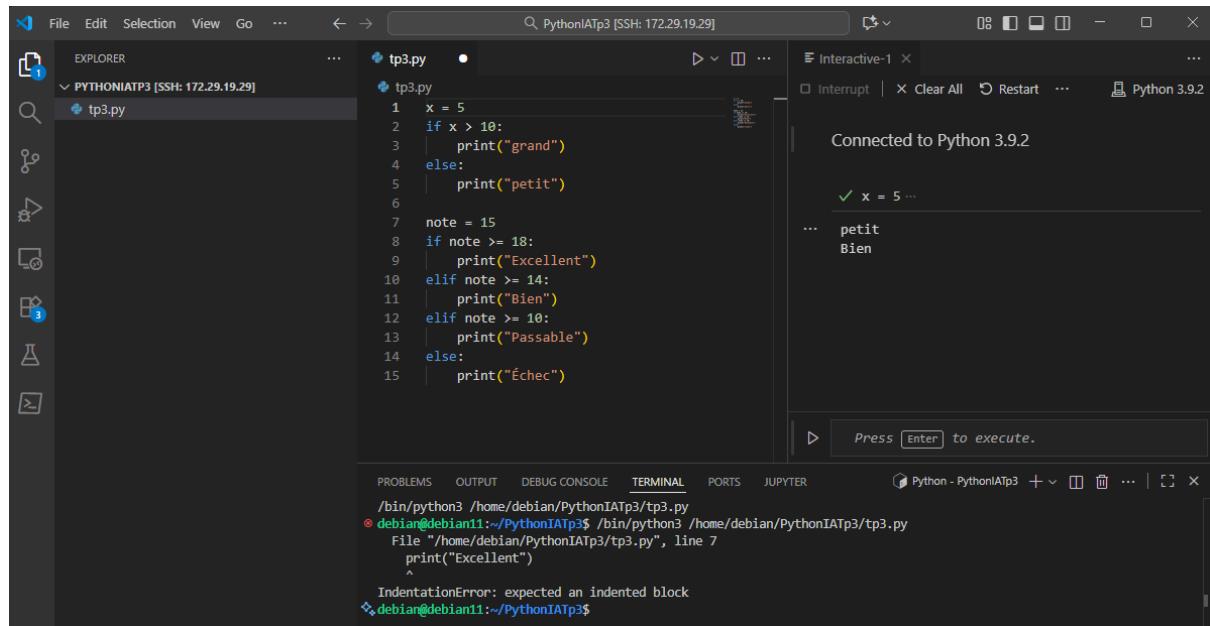
## TP3 IA



Groupe: Romain

Exemple :

### Python : Les condition si sinon



The screenshot shows a Python IDE interface. On the left, the Explorer panel shows a file named 'tp3.py'. The code in 'tp3.py' is:

```

1 x = 5
2 if x > 10:
3     print("grand")
4 else:
5     print("petit")
6
7 note = 15
8 if note >= 18:
9     print("Excellent")
10 elif note >= 14:
11     print("Bien")
12 elif note >= 10:
13     print("Passable")
14 else:
15     print("Échec")

```

In the center, the Interactive-1 pane shows the output of the code execution:

```

x = 5 ...
petit
Bien

```

At the bottom, the Terminal pane shows the command run and its output:

```

/bin/python3 /home/debian/PythonIATp3/tp3.py
 debian@debian11:~/PythonIATp3$ ./bin/python3 /home/debian/PythonIATp3/tp3.py
 File "/home/debian/PythonIATp3/tp3.py", line 7
     print("Excellent")
     ^
IndentationError: expected an indented block
 debian@debian11:~/PythonIATp3$
```

### Js: Les condition si sinon



The screenshot shows a code editor with the following JavaScript code:

```

if (x > 10) {
    console.log('grand');
} else {
    console.log('petit');
}

let note = 15;
if (note >= 18) {
    console.log("Excellent");
} else if (note >= 14) {
    console.log("Bien");
} else if (note >= 10) {
    console.log("Passable");
} else {
    console.log("Échec");
}
petit
Bien

```

### Python : Les boucles

```
for i in range(5):
    print(i)

x = 0
while x < 10:
    x += 1
    print(x)
```

Output:

```
0
1
2
3
4
5
6
7
8
9
```

Press Enter to execute.

### Js : Les boucles

```
> for (let i = 0; i < 5; i++) {
    console.log(i);
}

while (x < 10) {
    x++;
    console.log(x);
}
0
1
2
3
4
```

### Python : Les collection de tableau

```
nums = [1, 2, 3]
nums.append(4)
print(nums)

info = {"nom": "Bob", "age": 30}
print(info["nom"])
```

Connected to Python 3.9.2

Output:

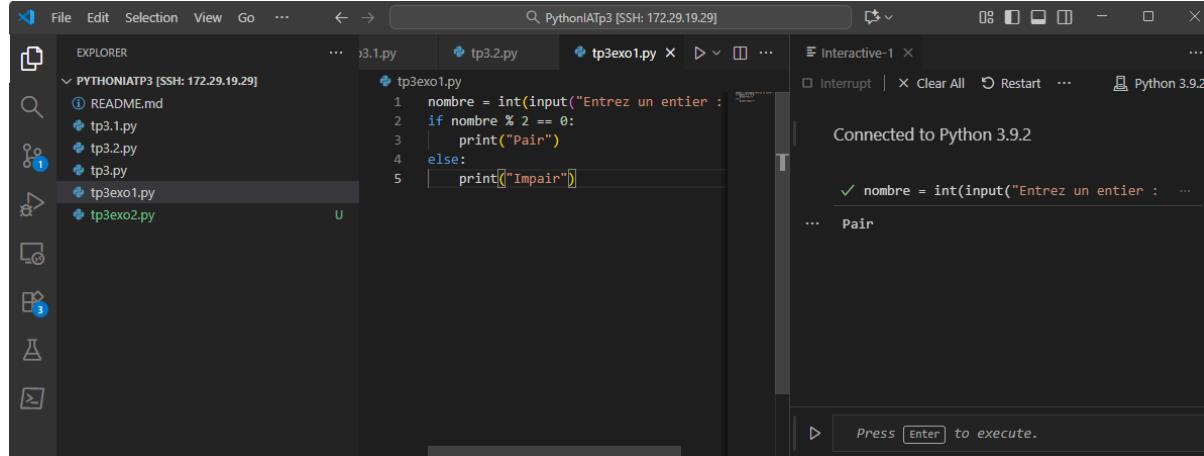
```
nums = [1, 2, 3] ...
[1, 2, 3, 4]
Bob
```

### Js : Les collection de tableau

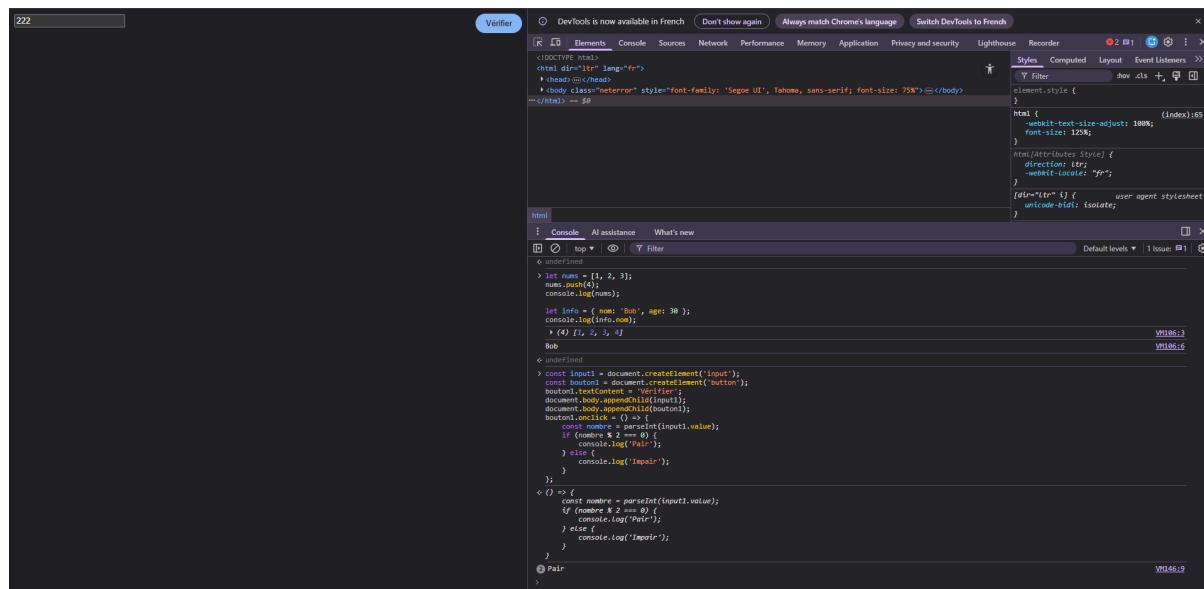
```
> let nums = [1, 2, 3];
  nums.push(4);
  console.log(nums);

let info = { nom: 'Bob', age: 30 };
console.log(info.nom);
> (4) [1, 2, 3, 4]
Bob
```

## Exercice 1 (Jaune)



```
File Edit Selection View Go ... < > PythonIATp3 [SSH: 172.29.19.29] ... Interactive-1 ... File Explorer ... tp3.1.py tp3.2.py tp3exo1.py tp3exo1.py ... Python 3.9.2 ... Connected to Python 3.9.2 ... nombr... ... Pair
```



```
Vérifier DevTools is now available in French Don't show again Always match Chrome's language Switch DevTools to French
```

```
Elements Console Sources Network Performance Memory Application Privacy and security Lighthouse Recorder Styles Computed Layout Event Listeners
```

```
html { -webkit-text-size-adjust: 100%; font-size: 125%; }
```

```
html[Attributes Style] { direction: ltr; -webkit-locale: "fr"; }
```

```
[dir="rtl"] { user agent stylesheet } unicode-bidi: isolate;
```

```
html
```

```
Console AI assistance What's new
```

```
top | Filter
```

```
undefined
```

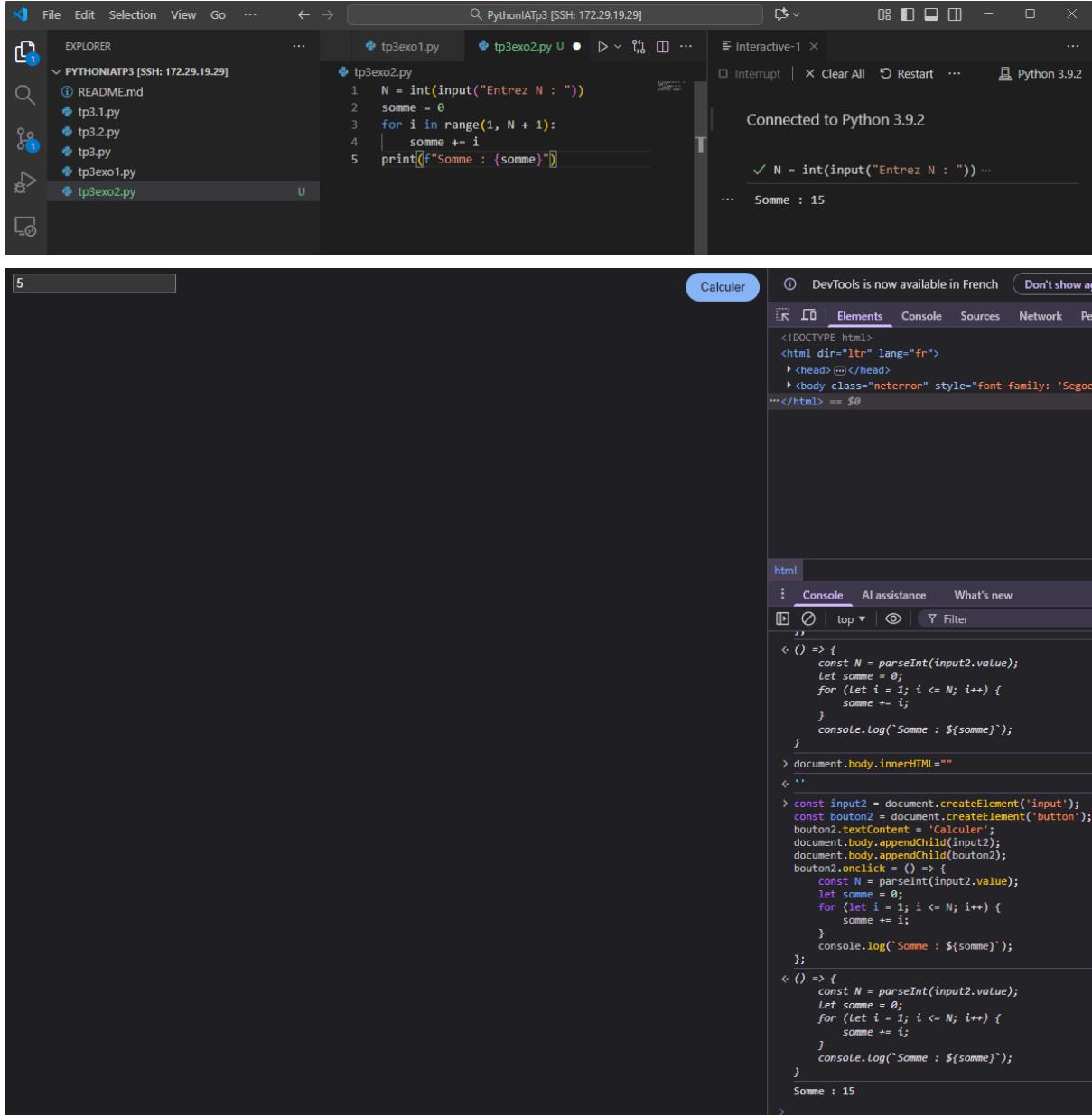
```
> let num = [1, 2, 3];
num.forEach(console.log);
let info = { nom: 'Bob', age: 30 };
console.log(info);
> (4) [1, 2, 3, 4]
Bob
```

```
undefined
```

```
> const input1 = document.createElement('input');
const bouton1 = document.createElement('button');
bouton1.textContent = 'Vérifier';
document.body.appendChild(input1);
document.body.appendChild(bouton1);
bouton1.onclick = () => {
    const nombre = parseInt(input1.value);
    if (nombre % 2 === 0) {
        console.log('Pair');
    } else {
        console.log('Impair');
    }
}
< (0) => {
    const nombre = parseInt((input1.value));
    if (nombre % 2 === 0) {
        console.log('Pair');
    } else {
        console.log('Impair');
    }
}
```

```
const input1 = document.createElement('input');
const bouton1 = document.createElement('button');
bouton1.textContent = 'Vérifier';
document.body.appendChild(input1);
document.body.appendChild(bouton1);
bouton1.onclick = () => {
    const nombre = parseInt(input1.value);
    if (nombre % 2 === 0) {
        console.log('Pair');
    } else {
        console.log('Impair');
    }
};
```

## Exercice 2 (Bleu)



The screenshot shows a terminal window titled "Python|ATp3 [SSH: 172.29.19.29]" and a browser developer tools interface.

**Terminal (Python):**

```

File Edit Selection View Go ... ← → ⌘ Python|ATp3 [SSH: 172.29.19.29]
EXPLORER ... tp3exo1.py tp3exo2.py U ...
PYTHONIATP3 [SSH: 172.29.19.29]
 README.md
 tp3.1.py
 tp3.2.py
 tp3.py
 tp3exo1.py
 tp3exo2.py U ...
tp3exo2.py
1 N = int(input("Entrez N : "))
2 somme = 0
3 for i in range(1, N + 1):
4     somme += i
5 print(f"Somme : {somme}")
Connected to Python 3.9.2
N = int(input("Entrez N : "))
Somme : 15

```

**Developer Tools (Browser):**

The browser's DevTools Elements tab shows the page structure:

```

<!DOCTYPE html>
<html dir="ltr" lang="fr">
  <head> ...</head>
  <body class="neterror" style="font-family: 'Segoe UI', sans-serif;">...</body>

```

The Console tab shows the executed JavaScript code and its output:

```

Calculator DevTools is now available in French Don't show again
Elements Console Sources Network Performance
<() => {
  const N = parseInt(input2.value);
  let somme = 0;
  for (let i = 1; i <= N; i++) {
    somme += i;
  }
  console.log(`Somme : ${somme}`);
}
> document.body.innerHTML=""
< ''
> const input2 = document.createElement('input');
const bouton2 = document.createElement('button');
bouton2.textContent = 'Calculer';
document.body.appendChild(input2);
document.body.appendChild(bouton2);
bouton2.onclick = () => {
  const N = parseInt(input2.value);
  let somme = 0;
  for (let i = 1; i <= N; i++) {
    somme += i;
  }
  console.log(`Somme : ${somme}`);
};
< () => {
  const N = parseInt(input2.value);
  let somme = 0;
  for (let i = 1; i <= N; i++) {
    somme += i;
  }
  console.log(`Somme : ${somme}`);
}
> Somme : 15

```

```

const input2 = document.createElement('input');
const bouton2 = document.createElement('button');
bouton2.textContent = 'Calculer';
document.body.appendChild(input2);
document.body.appendChild(bouton2);
bouton2.onclick = () => {
  const N = parseInt(input2.value);
  let somme = 0;
  for (let i = 1; i <= N; i++) {
    somme += i;
  }
  console.log(`Somme : ${somme}`);
}

```

};

## Exercice 3 (Bleu) si c'est Optimisé (Vert)

The screenshot shows a VS Code interface with the following details:

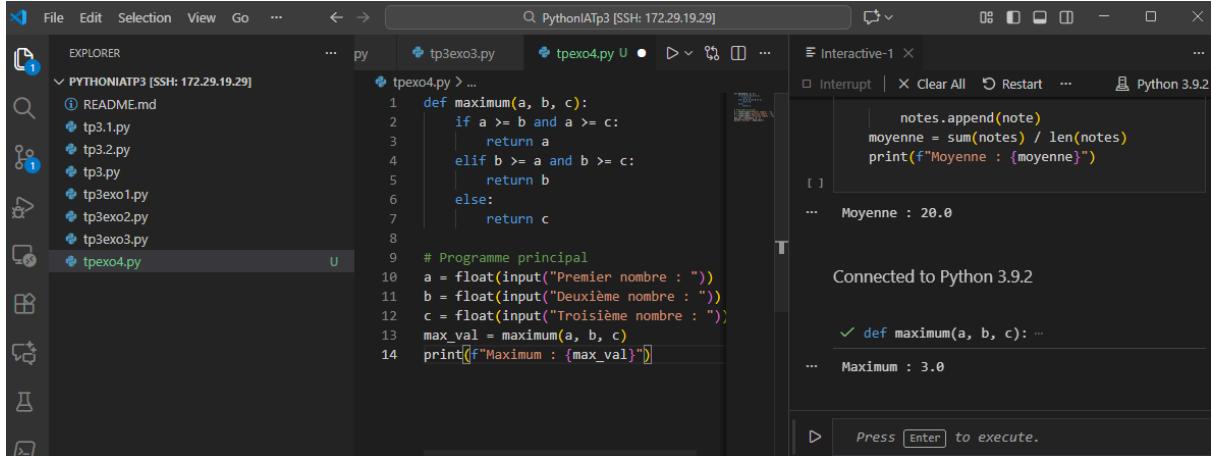
- File Explorer:** Shows a folder named "PYTHONIATP3 [SSH: 172.29.19.29]" containing files: README.md, tp3.1.py, tp3.2.py, tp3.py, tp3exo1.py, tp3exo2.py, tp3exo3.py, and tp3exo3.py (the active file).
- Code Editor:** Displays the content of tp3exo3.py:

```
notes = []
n = int(input("Nombre de notes : "))
for i in range(n):
    note = float(input(f"Note {i+1} : "))
    notes.append(note)
moyenne = sum(notes) / len(notes)
print(f"Moyenne : {moyenne}")
```
- Terminal:** Shows the output of the script: "Connected to Python 3.9.2", "notes = []", and "Moyenne : 20.0".
- Status Bar:** Shows the status bar with "PythonIATP3 [SSH: 172.29.19.29]".

```
> let notes = [];
  const n = parseInt(prompt("Nombre de notes : "));
  for (let i = 0; i < n; i++) {
    const note = parseFloat(prompt(`Note ${i+1} : `));
    notes.push(note);
  }
  const moyenne = notes.reduce((a, b) => a + b, 0) / notes.length;
  console.log(`Moyenne : ${moyenne}`);
Moyenne : 20
```

```
let notes = [];
const n = parseInt(prompt("Nombre de notes : "));
for (let i = 0; i < n; i++) {
    const note = parseFloat(prompt(`Note ${i+1} : `));
    notes.push(note);
}
const moyenne = notes.reduce((a, b) => a + b, 0) / notes.length;
console.log(`Moyenne : ${moyenne}`);
```

## Exercice 4 (Vert)



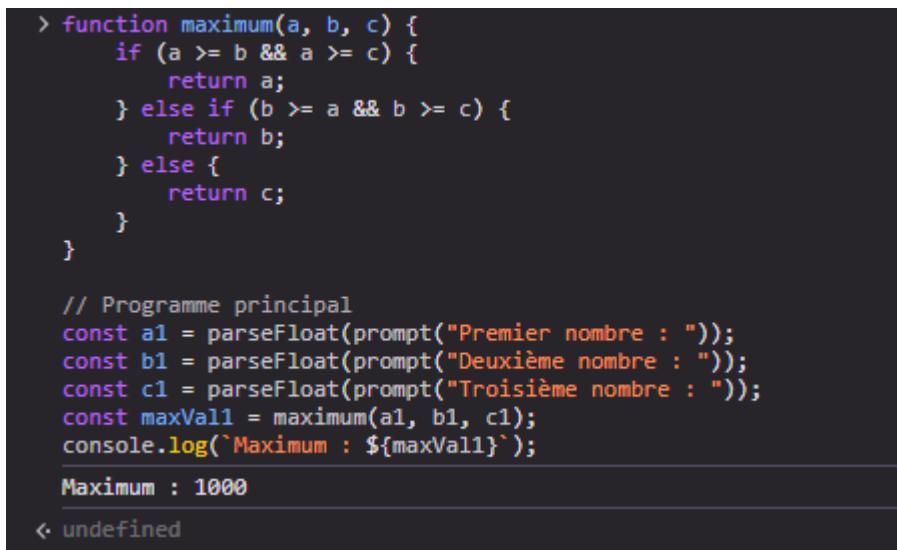
The screenshot shows a Python IDE interface. On the left, the file tree displays files like README.md, tp3.1.py, tp3.2.py, tp3.py, tp3exo1.py, tp3exo2.py, tp3exo3.py, and tpexo4.py. The main editor window contains Python code for finding the maximum of three numbers and calculating the average of a list of notes. The right panel shows the interactive shell connected to Python 3.9.2, displaying the results of the executed code.

```

def maximum(a, b, c):
    if a >= b and a >= c:
        return a
    elif b >= a and b >= c:
        return b
    else:
        return c

# Programme principal
notes.append(note)
moyenne = sum(notes) / len(notes)
print(f"\"Moyenne : {moyenne}\")"

```



The screenshot shows a terminal or code editor window. It displays a JavaScript function named maximum that takes three arguments (a, b, c) and returns the maximum value. Below it, a block of JavaScript code demonstrates the function's usage by prompting for three numbers and logging the result to the console. The output shows the maximum value as 1000.

```

function maximum(a, b, c) {
    if (a >= b && a >= c) {
        return a;
    } else if (b >= a && b >= c) {
        return b;
    } else {
        return c;
    }
}

// Programme principal
const a1 = parseFloat(prompt("Premier nombre : "));
const b1 = parseFloat(prompt("Deuxième nombre : "));
const c1 = parseFloat(prompt("Troisième nombre : "));
const maxVal1 = maximum(a1, b1, c1);
console.log(`Maximum : ${maxVal1}`);

```

```

function maximum(a, b, c) {
    if (a >= b && a >= c) {
        return a;
    } else if (b >= a && b >= c) {
        return b;
    } else {
        return c;
    }
}

// Programme principal
const a1 = parseFloat(prompt("Premier nombre : "));
const b1 = parseFloat(prompt("Deuxième nombre : "));
const c1 = parseFloat(prompt("Troisième nombre : "));
const maxVal1 = maximum(a1, b1, c1);
console.log(`Maximum : ${maxVal1}`);

```