Contrôler le déroulement du programme

Question 1 :

Quelle est la valeur de `num` après l'exécution du code suivant si l'utilisateur entre `10` au clavier ?

```
int num;
cin >> num;

if (num > 10)
   num -= 10;
else
   num += 10;
```

20

Question 2 :

Qu'affiche l'extrait de code suivant si l'utilisateur entre `70` au clavier ?

```
int temperature;
cout << "Enter a temperature: ";
cin >> temperature;
if (temperature < 50);
   cout << "It's cold!" << endl;
if (temperature > 50)
   cout << "It's hot!" << endl;
else
   cout << "Maybe it's raining?";
```

sortie :

```
It's cold!
It's hot!
```

```
Excellent! C'était une question délicate. Le point-virgule à la fin de la première
instruction if signifie ne rien faire si la condition est vraie.
```

Question 3 :

Qu'affiche l'extrait de code suivant si l'utilisateur entre `20` au clavier ?

```
int favorite;
cout << "Enter your favorite number: ";
cin >> favorite;
if (favorite == 13)
   cout << "That my favorite number too!" << endl;
   cout << "That's amazing!" << endl;
   cout << "Great minds think alike!" << endl;
```

sortie :

That's amazing! Great minds think alike!

```
Super! Le code est mal indenté, il semble donc que toutes ces instructions font
partie de l'instruction if.
```

Question 4 :

Qu'affichera l'extrait de code suivant ?

```
int num = 10;
while (num >= 1)
   cout << num << " ";
   num--;
```

sortie :

10 10 10 10 10 ... infiniment

Question 5 :

La boucle while est un exemple de a(n) _____.

boucle de pré-test

```
Bien! Nous testons d'abord la condition. Ensuite on boucle si la condition est
vraie.
```

Question 6 : Une boucle do-while est garantie d'exécuter _____.

*au moins une fois**

```
Bien! La boucle do s'exécute toujours la première fois.
```

Question 7 :

La boucle for contient 3 expressions dans l'ordre suivant :

initialiser, tester, incrémenter

Question 8 :

Une boucle située à l'intérieur d'une autre boucle est appelée a(n) _____.

boucle imbriquée

Question 9 :

Afin de terminer l'exécution d'une boucle, nous pouvons utiliser l'instruction _____.

break

Question 10 :

Si vous savez à l'avance combien de fois vous devez effectuer une boucle, quelle boucle utiliseriez-vous ?

boucle for

```
Utilisez une boucle for lorsque vous savez combien de fois vous devez effectuer
une boucle. Il est plus explicite à lire et plus facile à comprendre, tester et
déboguer.
```