

TD V :  
.NET

- a) Télécharger et ajouter "TD5.cs" au projet TD\_console
- b) Ajouter les lignes suivantes dans Main() de "Program.cs"

```
TD5 tD5 = new TD5();  
tD5.Test();
```

- c) Compléter la méthode "Titaah\_encode" qui retourne la séquence en paramètre encodé en Morse cadencé. Essayer avec "Hello world !" et afficher "Titititi ti titahtiti titahtiti tahtahtaah, titahtaah tahtahtaah titahti titahtiti tahhtiti, !".
  - L'espacement entre les lettres d'un mot est représenté par 1 espace.
  - L'espacement entre les mots est représenté par une virgule ( , ) suivi d'un espace : " , ".
- d) Compléter la méthode "Titaah\_decode" qui retourne la séquence en paramètre décodé du Morse cadencé. Essayer avec "Titititi ti titahtiti titahtiti tahtahtaah, titahtaah tahtahtaah titahti titahtiti tahhtiti, !" et afficher "Hello world !".

e) Compléter la méthode "Passerelle\_display" qui retourne l'affichage du tableau en paramètre.

- Chaque ligne du tableau sera représentée par 2 colonnes, la première représentera le nombre d'éléments, la seconde représentera l'emplacement des passerelles.
- Afficher le point d'entrée et le point de sortie de chaque ligne. Les points d'entrées sont des lettres majuscules qui commencent à 'A', les points de sortie sont des chiffres qui commencent à 1. Si il y a plus de 26 lignes, les points d'entrée des lignes suivantes seront : "AA" ; "AB" ; "AC" ; etc...
- Chaque élément d'une ligne est représenté par un pipe ( | )
- Chaque Passerelle est représentée par un égal ( = )

```
A B C D F
|=| |=| |
| |=|=| |=
| | |=| |
|=| | |=|
1 2 3 4 5
```

- f) Compléter la méthode "Passerelle" qui retourne pour chaque ligne du tableau en paramètre son point d'entrée suivi de son point de sortie.
- Si la case est un **1**, c'est une passerelle, on descend d'une ligne. Si la case de la ligne du dessus est un **1**, c'est une passerelle, on monte d'une ligne.
  - On ne passe qu'une fois sur une passerelle.
  - Si on a le choix entre deux passerelle, retourner "**Error**" pour la ligne en cours. Si on sort du tableau, retourner "**Error**" pour la ligne en cours.
  - Exemple :

```
A B C D F
|=| | | |
| |=|=| |=
| | |=| |
|=| | |=|
1 2 3 4 5
```

```
// -> "A3 ; B2 ; Error ; D1 ; Error."
```

g) Compléter la méthode "Cadran" qui affiche le contenu du tableau en paramètre ligne par ligne.

- Chaque chiffre et symbole est représenté par une suite de caractères d'une hauteur de 5 et d'une largeur de 3.
- Dans l'ordre : 0 1 2 3 4 5 6 7 8 9 + - \* / :

```
111 001 111 111 100 111 111 111 111 111 000 000 000 000
101 001 001 001 100 100 100 001 101 101 000 000 000 000
101 001 111 111 111 111 111 001 111 111 010 000 101 001
101 001 100 001 001 001 101 001 101 001 111 111 010 010
111 001 111 111 001 111 111 001 111 111 010 000 101 100
```

h) Compléter la méthode "Cadran\_to\_long" qui retourne le tableau transcrit en nombre.

- Retourner "Error" si il n y a pas de correspondance de chiffre ou de symbole.

```
001 111 111 111 111
001 001 101 001 101
001 111 101 001 111
001 001 101 001 101
001 111 111 001 111
```

```
// -> 13078
```

- i) Compléter la méthode "Cadran\_addition" qui retourne le résultat de l'addition entre le premier paramètre et la second paramètre.
- j) Compléter la méthode "Cadran\_soustraction" qui retourne le résultat de la soustraction entre le premier paramètre et la second paramètre.
- k) Compléter la méthode "Cadran\_multiplication" qui retourne le résultat de la multiplication entre le premier paramètre et la second paramètre.
- l) Compléter la méthode "Cadran\_division" qui retourne le résultat de la division entre le premier paramètre et la second paramètre.
- m) Compléter la méthode "Cadran\_operation" qui retourne le résultat de l'opération du paramètre.
- n) Compléter la méthode "Open\_close\_is\_valid" qui retourne si la chaîne de caractères passé en paramètre ferme correctement toutes les occurrences de parenthèses ( **()** ), accolades ( **{ }** ) et crochet ( **[ ]** ). Essayer avec "**((ahy[op\_i]]op{ }())**" et afficher **true**.
  - Si une occurrence n'a pas d'ouverture, afficher **false**.
  - Exemple : "**] [**" affiche **false**.

o) Compléter la méthode "Laby" qui affiche le contenu du tableau en paramètre ligne par ligne.

- [1, 0, 1, 1] représente dans l'ordre si il y a un mur à gauche, en haut, à droite et en bas. 1 est un mur, 0 un espace.
- Les murs sont représentés par des underscores ( \_ ) et des pipes ( | ).



p) Compléter la méthode "Laby\_is\_valid" qui affiche si le labyrinthe est correcte.

- Retourner false si il y a un trou dans le labyrinthe.

q) Compléter la méthode "Laby\_resolve" qui affiche le minimum de nombre de case entre le premier paramètre et le second paramètre.

- On se déplace horizontalement et verticalement.
- Les paramètres sont des tableaux à deux éléments représentant la position x et y du départ et de l'arrivé.
- Retourner "Error" Si il n y a pas de chemin entre le départ et l'arrivé. Retourner "Error" si il y a un trou dans le labyrinthe.
- Si plusieurs chemins sont possibles, retourner le chemin le plus court.