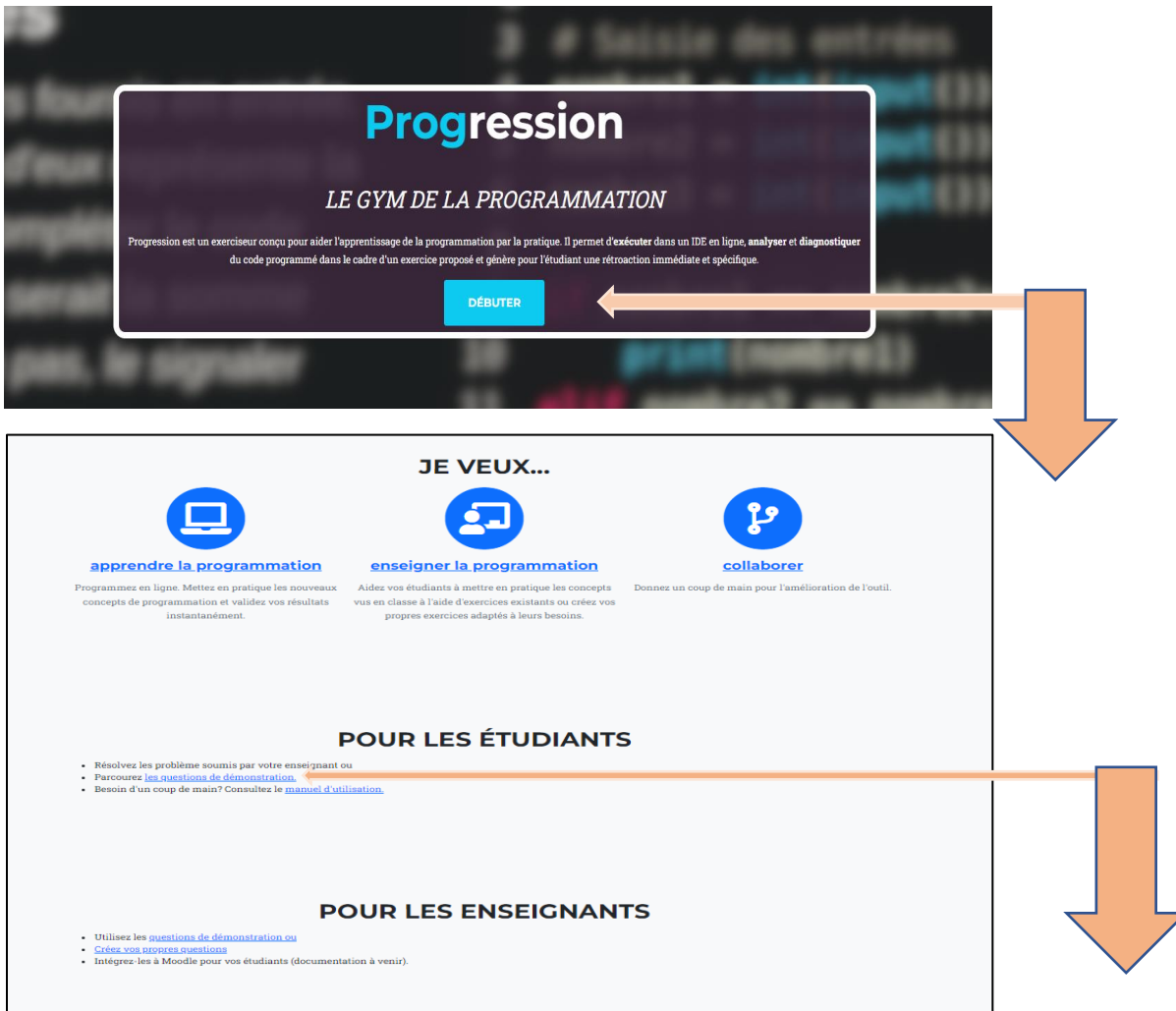


1- Accéder au portail de « Progression »

En suivant le lien : <https://progression.dti.crosemont.quebec/>



2- Accéder à la banque de questions disponible

Il est possible d'y accéder directement avec l'url :

https://progression.pages.dti.crosemont.quebec/contenu/prog_1/liste_questions.html

3- Naviguer dans la liste des questions dans la table de matières de gauche

Table des matières
 1. Les déclarations
 2. Les affectations
 3. Les entrées/sorties
 4. Les structures sélectives
 5. Les structures itératives
 6. Les fonctions
 7. Les chaînes de caractères
 8. Les tableaux

Les fondements de la programmation

1 Les déclarations

1.1 La syntaxe des déclarations

- ▶ Initialisations 1
- ▶ Initialisations 2
- ▶ Initialisations 3
- ▶ Initialisations 4

2 Les affectations

2.1 L'ordre des affectations

- ▶ Ordre des affectations 1

Table de matières organisée en une hiérarchie de liens thématiques incrémentale pour offrir une « progression » dans le niveau de difficulté, et un second niveau de sous-thématiques progressives également en difficulté

- 4- **Choisir une question** à droite, qui sont en ordre croissant dans le niveau de difficulté à l'intérieur de chaque sous-thématique : niveau de *base*, *intermédiaire*, et *défi*, ce dernier niveau identifié par l'icône du *piment*.

Table des matières
 1. Les déclarations
 2. Les affectations
 3. Les entrées/sorties
 4. Les structures sélectives
 5. Les structures itératives
 6. Les fonctions
 7. Les chaînes de caractères
 8. Les tableaux

7 Les chaînes de caractères

7.1 Traitement des caractères

- ▶ Codes numériques
- ▶ Inversion de la casse
- ▶ Caractères des codes numériques

7.2 Traitement des chaînes de caractères

- ▶ Nombre de symboles
- ▶ Texte vertical
- ▶ Comptage majuscules et minuscules
- ▶ Palindrome
- ▶ Sans espace
- ▶ Numéro d'assurance maladie 🌶

Pour chaque thématique, un **second niveau de sous-thématiques « progressives »** dans le niveau de difficulté

Pour chaque sous-thématique, une liste de liens vers des questions de niveau de difficulté croissant

- 5- **S'authentifier si requis** : à la première connexion, une **boîte de connexion** du portail informatique demande un code usager et un mot de passe :
- a. S'authentifier si l'on dispose du code et du mot de passe
 - b. Clic sur *mot de passe oublié* si c'est le cas
 - c. Le code et mot de passe est envoyé aux étudiants par courriel dans leur messagerie du cégep, à chaque début de session. **Important** : Les étudiants doivent d'abord changer leur mot de passe dans le portail informatique : <https://portail.dti.crosemont.quebec>

- 6- **Consulter les consignes** dans l'interface de l'exercice choisi affichée comme suit :

Énoncé du problème

niveau

choix du langage de programmation

Saisie du code, certaines lignes peuvent être masquées par souci de clarté.

Préférences affichage couleur fond et visualisation du code masqué

Entrées

- lettre (caractère)

Sortie attendue

lettre convertie (caractère)

À faire

Compléter le code pour produire en sortie le caractère entré en inversant sa casse.

Exemples

- Caractère majuscule

-Entrée :

A

• Sortie attendue :

a

- Caractère minuscule

• Entrée :

b

• Sortie attendue :

B

exemples d'entrées/sorties attendues.

Jeu de tests

1. Lettre majuscule
2. Lettre minuscule
3. Autre symbole
4. Espace

Entrée

z

Sortie attendue

z

Mode d'affichage par différences

Bouton pour valider le code

Entrées/sorties en mode console

```
1
2 char lettre;
3 char lettreConvertie;
4
5 // Entrée
6 lettre = sc.nextLine().charAt(0);
7 // À faire ...
8
9 System.out.println(lettreConvertie);
```

7- **Compléter le code** selon les consignes de gauche et valider en cliquant sur le bouton de droite.

8- Si Tentative non réussie :

Liste de toutes les tentatives réussies ou non, par langage avec possibilité d'y retourner

Configuration zone de code, et rétroaction éventuelle

Zone de codage. E/S graphique non autorisée

Sortie (affichage) des résultats, ou des erreurs de compilation s'il y a lieu

Jeu de couleurs pour chaque cas de test réussi (vert), ou non réussi (rouge) si la sortie attendue n'est pas identique à la sortie observée au caractère près!

Pour montrer le détail des différences entre sorties attendues et observées

Inversion de la casse

On souhaite inverser la casse d'un caractère, c'est à dire, convertir en majuscule le caractère reçu en minuscule, et inversement, convertir en minuscule le caractère reçu en entrée en majuscule. Pour tout caractère autre qu'une lettre, il n'y aura aucun changement.

Entrées

- lettre (caractère)

Sortie attendue

lettre convertie (caractère)

À faire

Compléter le code pour produire en sortie le caractère reçu en entrée en inversant sa casse.

Exemples

- Caractère majuscule

-Entrée :

A

• Sortie attendue :

a

- Caractère minuscule

• Entrée :

b

Jeu de tests

1. Lettre majuscule
2. Lettre minuscule
3. Autre symbole
4. Espace

Entrée

z

Sortie attendue

z

Sortie observée

z

Mode d'affichage par différences

```
1
2 char lettre;
3 char lettreConvertie;
4
5 // Entrée
6 lettre = sc.nextLine().charAt(0);
7 // À faire ...
8
9 System.out.println(lettreConvertie);
```

- Recommencer une autre tentative en complétant le code, s'aider en consultant les jeux de tests non réussis (au milieu en bas, en rouge) et sur les rétroactions (en haut à droite) ensuite valider

- Il est toujours possible de revenir à une tentative précédente (au milieu en haut) pour la continuer et ensuite la valider comme une nouvelle tentative.

9- Si Tentative réussie : Bravo !

The screenshot shows a Java IDE with a project named "Programme (bêta)". The main editor displays a Java program titled "Majuscule et Minuscule" which converts a character to its opposite case. The program code is as follows:

```
1 //
2 char lettre;
3 char lettreConvertie;
4
5 // Entrée
6 lettre = sc.nextLine().charAt(0);
7
8 if (Character.isUpperCase(lettre))
9     lettreConvertie=Character.toLowerCase(lettre);
10 else
11     lettreConvertie=Character.toUpperCase(lettre);
12
13 System.out.println(lettreConvertie);
```

Annotations and interface elements:

- Indique une tentative réussie:** Points to the green "base" button in the top toolbar.
- Liste des tentatives réussies ou non, par langage:** Points to the "Jeu de tests" (Test Suite) section on the left.
- Tentative de code complétée:** Points to the "Jouer" (Run) button in the top toolbar.
- Rétro-action sur la tentative qu'elle soit réussie ou non:** Points to the "Tout voir" (View All) button in the top toolbar.
- Entrée automatique de chaque cas de tests, Sortie du programme pour chaque cas de test, Validation du code par comparaison de la sortie attendue avec la sortie observée:** A red text box pointing to the test execution area.
- Jeu de couleurs pour chaque cas de test réussi (vert), ou non réussi (rouge) si la sortie attendue n'est pas identique à la sortie observée au caractère près!:** Points to the "Jeu de tests" list where tests are color-coded.
- Pour visualiser les détails de la comparaison tels que les espaces et sauts de lignes ..:** Points to the "Mode d'affichage par différences" (Diff view) toggle.
- Sortie produite par le programme pour le cas de test sélectionné, ou message d'erreur du compilateur:** Points to the "Sortie observée" (Observed output) field.

The "Jeu de tests" section on the left lists four tests:

1. Lettre majuscule
2. Lettre minuscule
3. Autre symbole
4. Espace

The "Sortie attendue" (Expected output) and "Sortie observée" (Observed output) fields show the character 'z' for the selected test.