

Nom	Prénom	Matricule	Programme
Gauthier-Drouin	François	111075877	Certificat en informatique

Travail Pratique 1

Rapport d'investigation:

Cahier des charges

À première vue, le programme sert à convertir une matrice de nombre (qui constitue l'image initiale en matrice de nombres binaires qui est ensuite convertie en étoiles et en espaces vides et transposée (i.e. l'image subit une rotation) pour finalement afficher une nouvelle image.

Stratégie

Voici la stratégie employée pour corriger l'ensemble des erreurs du programme :

- 1 – Corriger les erreurs apparentes dans les trois fichiers
- 2 – Faire un «build» du programme principal et lire dans la console la description des erreurs perçues par le compilateur g++
- 3 – Corriger une erreur
- 4 – Répéter les étapes 2 et 3 jusqu'à l'obtention d'un programme sans erreur

Erreur 1 : Mauvaise incrémentation des boucles «for» (syntaxe)

Localisation

Fichier fonctionsutilitaires.cpp

Lignes 34, 36, 58, 60, 89, 91, 101, 102

Justification:

Cette erreur était très apparente et a été remarquée en lisant simplement le code. L'incrémentation est faite à l'aide de l'expression «i++» et non pas «i++». Le même principe s'applique à la variable «j» lorsque c'est elle qui est utilisée.

```
58 for ((int i = 0; i < p_nbLignes; ++i))  
60 for ((int j = 0; j < p_nbColonnes; ++j))
```

Explications :

L'incrémentation est réalisée seulement si la syntaxe est bonne. De plus, la variable utilisée pour l'incrémentation doit toujours être la même que celle qui définit les limites de la boucle «for» (i.e. l'expression d'incrémentation «j++» ne peut pas être utilisée dans une boucle dont les conditions limites sont «i>0» et «i<p_nbLignes», par exemple.

Solution

```
58 for (int i = 0; i < p_nbLignes; i++)  
60 for (int j = 0; j < p_nbColonnes; j++)
```

Justification:

Comme indiqué dans la capture d'écran ci-dessus, les expressions d'incrémentation ont été modifiées afin de respecter la syntaxe c++. Cette expression a également été modifiée à la ligne 36 parce que l'expression d'incrémentation utilisait la variable «j» alors que la boucle utilisait plutôt la variable «i».

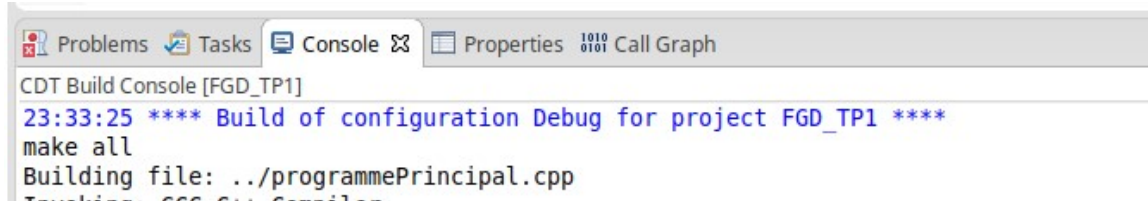
Erreur 2 : Virgule manquante (syntaxe)

Localisation

Fichier programmePrincipal.cpp
Ligne 48

Justification:

La première erreur soulignée par le compilateur concerne une virgule manquante entre deux éléments de la matrice image[HAUTEUR][LARGEUR].



La virgule est utilisée afin de changer de ligne dans la matrice de chiffres servant d'image initiale. Sans cette virgule, le compilateur détecte 16 lignes au lieu de 17 et génère une erreur car la matrice n'est pas complète.

Solution

Justification:

La virgule est ajoutée et un «build» du programme est exécuté de nouveau.

```
48      181, 179, 176, 174, 170, 169 },
```

Erreur 3 : Variable «imageBinaire» non-définie lors de l'appel de la fonction «copie_tableau» (édition de liens)

Localisation

Fichier programmePrincipal.cpp

Ligne 54

Justification:

La figure ci-dessous représente une capture d'écran de la console après avoir corrigé la première erreur et complété un deuxième «build».

```
23:40:33 **** Incremental Build of configuration Debug for project FGD_TP1 ****
make all
Building file: ../programmePrincipal.cpp
Invoking: GCC C++ Compiler
```

Puisque la variable «imageBinaire» était déclarée après l'appel de la fonction «copie_tableau», une erreur était générée lors de l'appel de cette dernière.

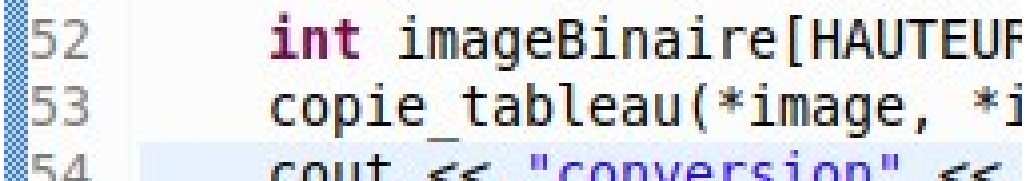


```
51
52 copie_tableau(*image, *imageBinaire);
53 cout << "conversion" << endl;
54 int imageBinaire[HAUTEUR];
```

Solution

Justification:

Pour corriger cette erreur, la ligne dans laquelle la variable «imageBinaire» est définie est placée au-dessus de l'appel de la fonction «copie_tableau».



```
52 int imageBinaire[HAUTEUR];
53 copie_tableau(*image, *imageBinaire);
54 cout << "conversion" << endl;
```

Erreur 4 : Fichier «fonctionsUtilitaires.h» exclues du programmePrincipal.cpp (mauvaise pratique)

Localisation

Fichier programmePrincipal.cpp

Ligne 10

Justification:

Les fonctions définies dans le fichier «fonctionsUtilitaires.cpp» ne sont pas trouvées par le programme principal.

```
23:46:53 *** Incremental Build of configuration Debug for project FGD_TP1 ***
make all
Building file: ../programmePrincipal.cpp
Invoking: GCC C++ Compiler
g++ -std=c++1y -O0 -g3 -Wall -c -fmessage-length=0 -MMD -MP -MF"programmePrincipal.
../programmePrincipal.cpp: In function 'int main()':
../programmePrincipal.cpp:56:2: error: 'copie_tableau' was not declared in this scope
copie_tableau(*image, *imageBinaire, HAUTEUR, LARGEUR);
^~~~~~
../programmePrincipal.cpp:58:2: error: 'conversionBinaire' was not declared in this scope
conversionBinaire(imageBinaire);
^~~~~~
../programmePrincipal.cpp:59:2: error: 'affiche_image' was not declared in this scope
```

Solution

Justification:

Il faut inclure le fichier «header» dans le programme principal afin que le compilateur aie accès à la définition des fonctions lorsqu'elles sont appelées.

```
10 #include "fonctionsUtilitaires.h"
```

Erreur 5 : Déclaration des constantes «HAUTEUR» et «LARGEUR» dans le mauvais fichier (logique)

Localisation

Fichier fonctionsUtilitaires.h

Lignes 11-12

Justification:

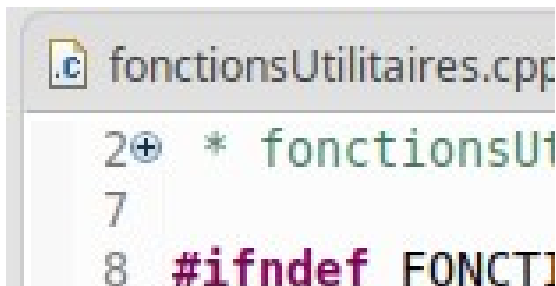
Les variables «HAUTEUR» et «LARGEUR» ne sont pas déclarées dans le fichier «header». Le fichier «header» est inclus dans le programme principal, mais pas l'inverse. C'est donc dire que les variables qui sont déclarées dans le fichier «header» pourront être lues par le compilateur à partir du fichier principal sans erreur.

```
23:48:31 **** Incremental Build of configuration Debug for project FGD_TP1 ****
make all
Building file: ../programmePrincipal.cpp
Invoking: GCC C++ Compiler
g++ -std=c++1y -O0 -g3 -Wall -c -fmessage-length=0 -MMD -MP -MF"programmePrincipal.
In file included from ../programmePrincipal.cpp:10:0:
../fonctionsUtilitaires.h:14:33: error: 'HAUTEUR' was not declared in this scope
void rotation(int p_imageSource[HAUTEUR][LARGEUR], int p_imageDestination[HAUTEUR]
                                ^~~~~~
../fonctionsUtilitaires.h:14:42: error: 'LARGEUR' was not declared in this scope
void rotation(int p_imageSource[HAUTEUR][LARGEUR], int p_imageDestination[HAUTEUR]
                                ^~~~~~
../fonctionsUtilitaires.h:14:50: error: expected ')' before ',' token
void rotation(int p_imageSource[HAUTEUR][LARGEUR], int p_imageDestination[HAUTEUR]
                                ^
```

Solution

Justification:

Il faut ajouter la définition des deux constantes au fichier «header», tel qu'indiqué ci-dessous.



```
fonctionsUtilitaires.cpp
2+  /* fonctionsUtilitaires.h
7
8  #ifndef FONCT
```

Erreur 6 : Fonction «copie_tableau» non-définie (erreur de logique)

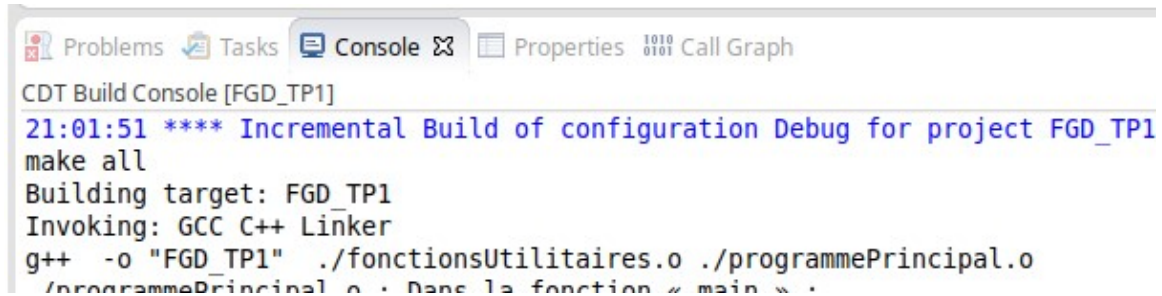
Localisation

Fichier fonctionsutilitaires.h

Lignes 17-18

Justification:

En exécutant un «build» une nouvelle fois, l'erreur «makefile :44» est observée ci-dessous. Cette erreur provient d'une référence indéfinie vers la fonction «affiche_image». Toutefois, aucun numéro de ligne ou d'emplacement spécifique n'est indiqué par le compilateur dans la console. Une recherche visuelle entre les deux fichiers de fonctions permet de constater l'absence de la définition de la fonction «copie_tableau» dans le fichier «header». Cette absence génère une erreur lorsque la fonction est appelée dans le programme principal.



```
CDT Build Console [FGD_TP1]
21:01:51 **** Incremental Build of configuration Debug for project FGD_TP1
make all
Building target: FGD_TP1
Invoking: GCC C++ Linker
g++ -o "FGD_TP1" ./fonctionsUtilitaires.o ./programmePrincipal.o
./programmePrincipal.o : Dans la fonction « main » :
```

Solution

Justification:

La définition de la fonction «copie_tableau» a été ajoutée au fichier «header», qui fait le lien entre le programme principal et le fichier «fonctionsUtilitaires.cpp» dans lequel les fonctions sont écrites.

```
18 void copie_tableau(int* p_tableau)
```

Avertissement 1 : Variable «c_affiche» non-utilisée (mise en garde à corriger)

Localisation

Fichier fonctionsutilitaires.cpp

Ligne 57

Justification:

La variable «c_affiche» est définie, tel qu'indiqué ci-dessous, mais n'est jamais appelée.

```
54 void affiche_image(int* p_image
55 {
```

Ceci génère une mise en garde dans la console.

```
CDT Build Console [FGD_TP1]
00:08:28 **** Incremental Build of configuration Debug for project FGD_TP1 ****
make all
Building file: ../fonctionsUtilitaires.cpp
Invoking: GCC C++ Compiler
g++ -std=c++1y -O0 -g3 -Wall -c -fmessage-length=0 -MMD -MP -MF"fonctionsUtilitaires.d"
../fonctionsUtilitaires.cpp: In function 'void affiche_image(int*, int)':
../fonctionsUtilitaires.cpp:57:7: warning: unused variable 'c_affiche' [-Wunused-variab
    char c_affiche;
        ^~~~~~
Finished building: ../fonctionsUtilitaires.cpp
```

Solution

Justification:

La définition de la variable est supprimée et la mise en garde est enlevée.

```
54 void affiche_image(int* p_image
55 {
56
```

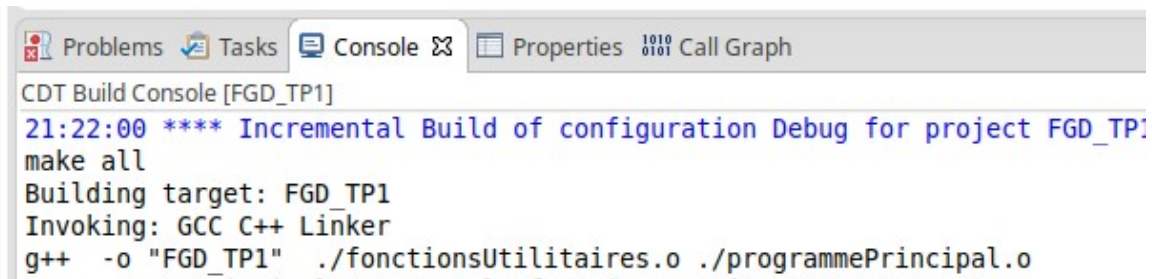

Erreur 7 : Absence de l'argument «p_nbColonnes» dans la définition de la fonction «affiche_image» (édition de liens)

Localisation

Fichier fonctionsUtilitaires.cpp
Lignes 54-56

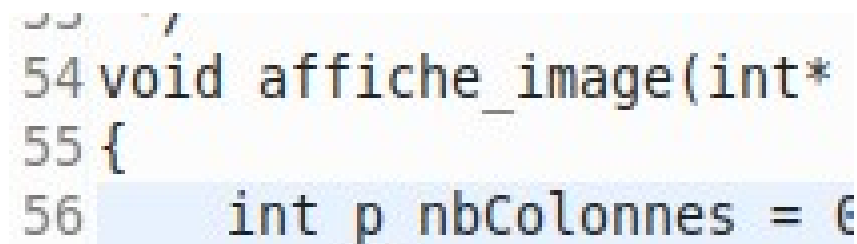
Justification:

Tel qu'énoncé dans la justification de l'erreur 6, une erreur est présente dans les références de la fonction «affiche_image».



```
CDT Build Console [FGD_TP1]
21:22:00 **** Incremental Build of configuration Debug for project FGD_TP1
make all
Building target: FGD_TP1
Invoking: GCC C++ Linker
g++ -o "FGD_TP1" ./fonctionsUtilitaires.o ./programmePrincipal.o
```

Un examen visuel de la définition de la fonction permet de constater l'absence d'un paramètre contenu dans la fonction. C'est ce qui crée une erreur; lorsque le compilateur exécute la ligne 59 du programme (tel qu'indiqué ci-dessous), il rencontre la variable «p_nbColonnes», qui ne fait pas partie des arguments de la fonction et ne peut donc pas être utilisée à l'intérieur de celle-ci.



```
54 void affiche_image(int* p_image)
55 {
56     int p_nbColonnes = 0;
```

Quand int p_nbColonnes est déclarée dans la définition de la fonction, on doit enlever son initialisation. Le contenu de la ligne 56 peut donc être effacé.

Solution

Justification:

L'argument est ajouté et l'initialisation de la ligne 56 («int p_nbColonnes = 0;») est effacé, tel qu'indiqué dans la capture d'écran ci-dessous.

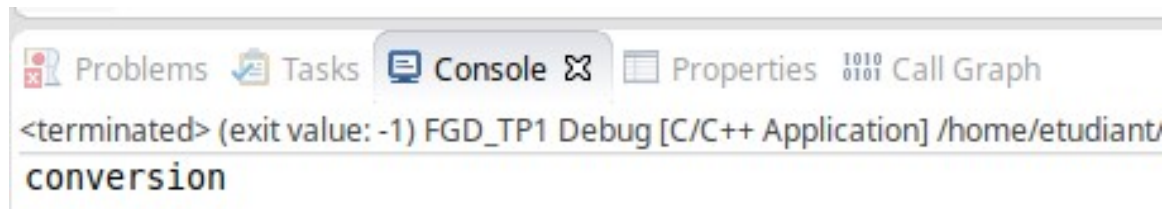


```
54 void affiche_image(int* p_image, int p_nbColonnes)
55 {
```

Après la correction de l'erreur 7, un «build» est effectué et il n'y a plus d'erreur apparente, comme en témoigne la capture d'écran ci-dessous.

```
CDT Build Console [FGD_TP1]
21:44:10 **** Incremental Build of configuration Debug for project FGD_TP1 ****
make all
Building file: ../fonctionsUtilitaires.cpp
Invoking: GCC C++ Compiler
g++ -std=c++1y -O0 -g3 -Wall -c -fmessage-length=0 -MMD -MP -MF"fonctionsUtilitaires.d"
Finished building: ../fonctionsUtilitaires.cpp
```

Il est donc possible d'exécuter le programme. À la suite de l'exécution, voici ce qui est obtenu dans la console :



The screenshot shows the IDE's console window with the following content:

```
<terminated> (exit value: -1) FGD_TP1 Debug [C/C++ Application] /home/etudiant/
conversion
```

Aucun problème n'est détecté. En analysant le programme du «main», on comprend que le programme s'exécute mal (ou arrête complètement) lorsqu'il passe dans la fonction «conversionBinaire».



The screenshot shows three lines of C++ code from a source file:

```
55     int imageBinaire[HAUTEUR]
56     copie_tableau(*image, *j
57     cout << "conversion" <<
```

Erreur 8 : Mauvaise incrémentation (syntaxe) et inversion des index de largeur et hauteur de «p_image» (logique)

Localisation

Fichier fonctionsUtilitaires.cpp
Lignes 36-44

Justification:

Le mauvais incrémenteur est utilisé pour la deuxième boucle.

De plus, les indexs sont inversés, ce qui empêche la fonction de s'exécuter proprement.

```
32 void conversionBinaire(int p_  
33 {  
34     for (int i = 0; i < HAUTE  
35     {  
36         for (int j = 0; j < L  
37         {  
38             if (p_image[i][j]  
39             {  
40                 p_image[i][i]
```

Solution

Justification:

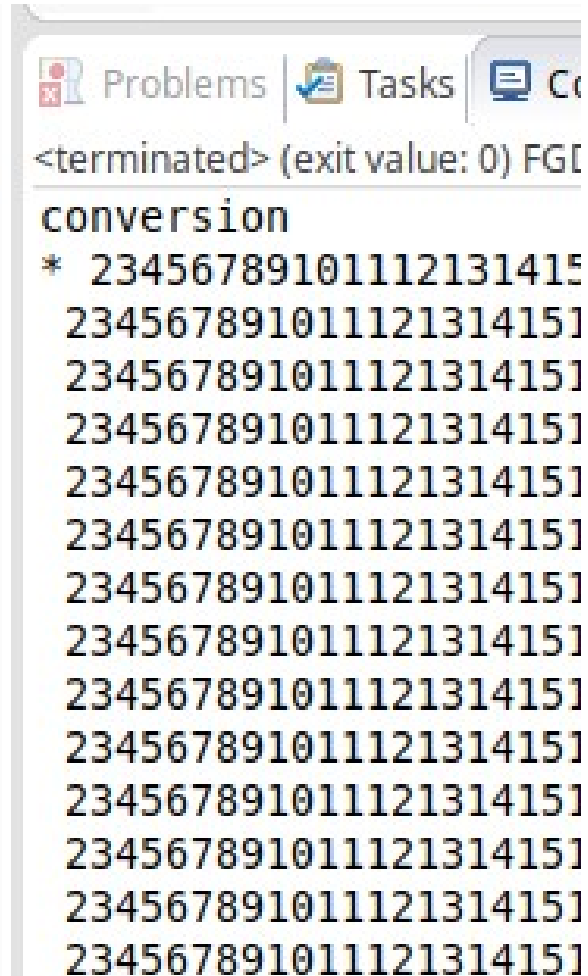
L'index [i] est s'incrémente sur chaque colonne, alors que l'index [j] le fait sur chaque ligne. La correction est donc effectuée.

```
31 */  
32 void conversionBinaire(int p_  
33 {  
34     for (int i = 0; i < HAUTE  
35     {  
36         for (int j = 0; j < L  
37         {  
38             if (p_image[i][j]  
39             {
```

Explications :

Le programme est exécuté, et le résultat de la console est modifié.

Les fonctions «conversionBinaire» et «rotation» semblent mal s'exécuter.



The screenshot shows a C++ IDE console window with tabs for 'Problems', 'Tasks', and 'Console'. The console output is as follows:

```
<terminated> (exit value: 0) FG[
conversion
* 23456789101112131415
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
  234567891011121314151
```

Erreur 9 : Inversion des arguments de la variable «p_imageDestination» dans la fonction «rotation» (logique)

Localisation

Fichier fonctionsUtilitaires.cpp
Ligne 86

Justification:

La fonction «rotation» transpose la matrice «p_imageSource» vers la matrice «p_imageDestination». Puisque la deuxième matrice est transposée, ses arguments doivent être inversés.

```
86 void rotation(int p_imageSource[HAUTEUR][LARGEUR], int p_imageDestination[LARGEUR][HAUTEUR])
```

Solution

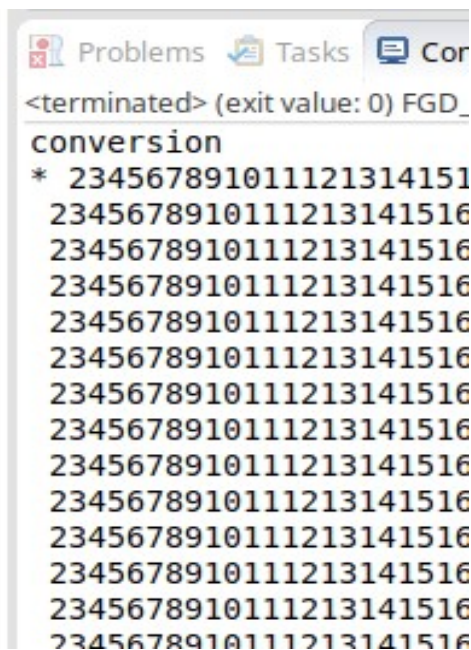
Justification:

La largeur devient la hauteur, et vice-versa dans le deuxième argument de cette fonction.

```
86 void rotation(int p_imageSource[HAUTEUR][LARGEUR], int p_imageDestination[HAUTEUR][LARGEUR])
```

Explications :

La console devrait imprimer «imageBinaire» entre «conversion» et «rotation», mais la conversion en binaires ne se fait pas.



```
<terminated> (exit value: 0) FGD_
conversion
* 2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
2345678910111213141516
```

Erreur 10 : Mauvaise définition du pointeur (logique)

Localisation

Fichier fonctionsUtilitaires.cpp

Lignes 62, 68, 74

Justification:

Ce problème est apparent de façon visuelle. Puisque l'index «j» est à l'extérieur de la définition du pointeur dans la fonction «affiche_image», sa valeur n'est pas prise en compte dans l'assignation des symboles «*» ou «>» aux chiffres binaires.

```
62         if (*(p_image + (i
63             {
64                 cout << ' '; /
65             }
66         else
67         {
68             if (*(n_image
```

Solution

Justification:

L'index «j» est donc inclus dans la définition du pointeur, tel que montré ci-dessous.

```
62         if (*(p_image + (i
63             {
64                 cout << ' '; /
65             }
66         else
67         {
68             if (*(n_image
```

Explications :

À l'exécution du code, la console imprime désormais la matrice imageBinaire (17 colonnes x 21 lignes), suivi du mot «rotation» et de l'image finale «i» (21 colonnes x 17 lignes).

```
Problems Ta
<terminated> (exit va
conversion
*****
**
**
**      *****
**      *****
**      ** *****
**      *****
**      *****
**      *****
**      ** *
**      * *
**      ** * *
**      * *****
**
**
***
*****
```

Erreur 11 : Inversion des arguments de la fonction «affiche_image» (logique)

Localisation

Fichier programmePrincipal.cpp
Lignes 61-63

Justification:

Puisque la fonction «affiche_image» est appelée après que la rotation de l'image soit complétée, ses arguments doivent être inversés.

```
61      cout << "rotat  
62      int i[HAUTEUR]  
63      rotation(image
```

Solution

Justification:

L'argument «HAUTEUR» est donc remplacé par l'argument «LARGEUR».

```
61      cout << "rotat:  
62      int i[LARGEUR]  
63      rotation(image
```


Erreur 12 : Mauvais arguments de la variable «p_imageDestination» (logique)

Localisation

Fichier fonctionsutilitaires.h
Lignes 86-87

Justification:

L'argument doit être modifié parce qu'on réfère présentement à la [HAUTEUR] alors qu'on doit itérer sur la [LARGEUR] en premier lorsque l'image est tournée.

```
15 void rotation(int p_imageSource[HAUTEUR][LARGEUR], int p_imageDestination[LARGEUR][HAUTEUR])
```

Solution

Justification:

La correction est complétée :

```
15 void rotation(int p_imageSource[HAUTEUR][LARGEUR], int p_imageDestination[LARGEUR][HAUTEUR])
```

Erreur 13 : Mauvaise étendue d'itération pour l'index «j» (logique)

Localisation

Fichier fonctionsutilitaires.cpp
Ligne 89

Justification:

```
89 for (int j = 0; j < HAUTEUR; j++)
```

Solution

Justification:

Il faut itérer sur LARGEUR pour aller chercher le nombre entier de lignes (21 au lieu de 17). Autrement, le traitement de l'image est incomplet et les nombres des colonnes 18 à 21 ne sont pas convertis en symboles («*» ou « »).

```
89 for (int j = 0; j < LARGEUR; j++)
```