

# INF2050 – Outils et pratiques de développement logiciel

Normes de codification

Jacques Berger

# Objectifs

Introduire une pratique professionnelle visant l'uniformité du code source

# Prérequis

# Programmation

# Personnalité

Personnalité créatrice

L'art de la programmation

Le style

# Le style

La mise en forme du code

La mise en forme de la documentation dans le code

# Le style

La position des accolades

Les endroits où l'on place des espaces

Dans les parenthèses, après les mot-clés, etc.

Indentation

Nombre d'espaces, tabulation

# Le style

La largeur des lignes de code

La position des déclarations de variables

La mise en page des commentaires

# Le style

L'ordre d'apparition des éléments dans un fichier  
ou une classe

Variables membres, constructeurs,  
destructeurs, méthodes publiques, etc.

# Le style

## Nomenclature

Le choix des noms des identifiants

Préfixes, suffixes, classes, variables, packages,  
la casse des identifiants

# Norme

La lisibilité du code est grandement affectée par le style

Plusieurs styles différents dans un même projet (ou fichier) vont nuire à la lisibilité

Une différence de style nuit à la lecture comme une faute de français dans un texte

# Norme

L'équipe passe avant l'individu

On recommande donc d'uniformiser le code en n'adoptant qu'un seul style pour toute l'équipe

Ça devient une norme de codification (coding convention) ou un standard de programmation

# Norme

Certains langages ont déjà une norme préétablie (ex. Java, Python)

D'autres langages ont un style non-écrit défini par les pratiques du marché (ex. C++)

Peu importe le style, tant que toute l'équipe suit la même norme

# Nomenclature

La nomenclature est l'élément le plus important du style, il touche fortement la lisibilité du code

Choisissez des noms significatifs!

# Nomenclature

La casse est un problème récurrent

camelCase

PascalCase

snake\_case

hyphen-case

# Discussions

Que faire contre ceux qui refusent de suivre la norme?

# Discussions

Si la norme change, devrait-on changer le code également pour respecter la nouvelle norme?

# Discussions

Est-ce pertinent d'avoir un comité qui détermine les normes de codification dans l'entreprise?

# Complexité

Une norme de codification trop élaborée sera difficile à mémoriser et à appliquer

Gardez les normes simples et courtes

# Outils

Mise en forme automatique

Beautifier

Intégré à l'IDE

Vérification automatique du style

Checkstyle

Peut être intégré à l'IDE

# Conclusion

Un style uniforme est important pour améliorer la maintenabilité du code

Cependant, le développeur doit se concentrer sur autre chose que le style durant son travail

# Conclusion

Configurez l'éditeur ou l'IDE de façon à respecter la norme de l'équipe

Utilisez des outils comme Checkstyle ou des Beautifieurs (si possible)

# Conclusion

Autant que possible, suivez les standards du langage, sinon documentez votre norme de codification de façon courte et simple

# Plus loin...

Normes de Java (Sun – document désuet)

<http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>

Code Complete, 2nd edition

Steve McConnell

Microsoft Press, 2004

The Practice of Programming

Brian W. Kernighan, Rob Pike

Addison-Wesley Professional, 1999