Cours de codes collaboratifs

Julien Mathiaud[1,2]

[1]CEA/CESTA - Le Barp
[2]julien.mathiaud@enseirb-matmeca.fr
IMB (UMR 5251),
Université de Bordeaux

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage
Versionnage
Gestion des sources
Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées
Sorties

Non-regression et validation

Déroulé du cours

1 Objectifs du cours

2 Qu'est-ce qu'un code collaboratif ?

3 Outils de développement en commun

4 Outils de debug

5 Outils d'optimisation

6 Entrées/Sorties

7 Non-regression et validation

8 Service aux utilisateurs : dépannage/documentation

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Déroulé du cours

1. Objectifs du cours
2. Qu'est-ce qu'un code collaboratif ?
3. Outils de développement en commun
4. Outils de debug
5. Outils d'optimisation
6. Entrées/Sorties
7. Non-regression et validation
8. Service aux utilisateurs : dépannage/documentation
9. Conclusions

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Objectifs

- Apprendre à gérer un code de façon intelligente
- Touver le bon compromis entre développement et qualité
- Appliquer directement sur le projet code les méthodes étudiées

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Organisation,validation et planning

Organisation

- 1 à 2 séances de cours
- 1 à 2 séances de tp

Evaluation

- Mise en place d'une démarche qualité autour du projet code.

Planning

- cours+tps d'ici fin octobre
- réunions régulières en novembre décembre pour faire un état d'avancement du code

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Déroulé du cours

1 Objectifs du cours

2 Qu'est-ce qu'un code collaboratif ?

3 Outils de développement en commun

4 Outils de debug

5 Outils d'optimisation

6 Entrées/Sorties

7 Non-regression et validation

8 Service aux utilisateurs : dépannage/documentation

9 Conclusions

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun
- Règles de codage
- Versionnage
- Gestion des sources
- Gestion de projet

Outils de debug

Outils d'optimisation
- Exercice de debug et optimisation

Entrées/Sorties
- Entrées
- Sorties

Non-regression et validation

Un premier exemple : le code OpenFoam

Code développé par ESI group : http://www.openfoam.com/

Supporté par la fondation : http://www.openfoam.org/

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun
- Règles de codage
- Versionnage
- Gestion des sources
- Gestion de projet

Outils de debug

Outils d'optimisation
- Exercice de debug et optimisation

Entrées/Sorties
- Entrées
- Sorties

Non-regression et validation

Définition

Code collaboratif

Un code collaboratif est un projet informatique faisant intervenir plusieurs intervenants afin de remplir une tâche.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Les différentes étapes

Les étapes

- création d'un projet
- développement du projet
- mise à disposition des utilisateurs
- support aux utilisateurs

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Création d'un projet scientifique

- Identifier un besoin, une tâche à accomplir,
- Vérifier que ce besoin ne peut être résolu que par la création d'un nouveau code
- Etablir un cahier des charges

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun
Règles de codage
Versionnage
Gestion des sources
Gestion de projet

Outils de debug

Outils d'optimisation
Exercice de debug et optimisation

Entrées/Sorties
Entrées
Sorties

Non-regression et validation

Cahier des charges

Les étapes

- Etablissement de l'objectif final
- Définition des étapes pour obtenir le résultat final
- Estimer la durée de développement et validation du code en hommes/an
- Calibrer l'équipe de développement afin d'obtenir le résultat en temps voulu
- Définir un planning de développement et de livrables

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Création d'une équipe code

Equipe

- Un chef de projet
- Un architecte (informatique)
- Des modélisateurs
- Des développeurs...

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage
Versionnage
Gestion des sources
Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées
Sorties

Non-regression et validation

Déroulé du cours

1 Objectifs du cours

2 Qu'est-ce qu'un code collaboratif ?

3 Outils de développement en commun

4 Outils de debug

5 Outils d'optimisation

6 Entrées/Sorties

7 Non-regression et validation

8 Service aux utilisateurs : dépannage/documentation

9 Conclusions

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Règles de codage

Règles

- Chaque langage de programmation a ses spécificités.
- Mais chaque langage a aussi ses permissivités.
- Il faut que ces dernières soient les mêmes pour toute l'équipe.

En pratique sur un projet Fortran/C/C++ scientifique

- Eviter les goto
- Eviter les optimisations à outrance du C qui nuisent à la lisibilité

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Un exemple de règles

Ecriture du code

- les noms de fonctions et de classes commencent par une majuscule,
- les noms de variables commencent par une minuscule
- les noms de fonctions, de classes et de variables sont en **** (majuscule ou minuscule selon le cas). L'underscore _ est autorisé pour ajouter un suffixe court et pour séparer les éléments syntaxiques
  exemple variable : variableBienNomme_1
- une seule instruction par ligne
- indentation de bloc : tabulations uniquement
- les variables sont typées dès que possible
- limiter la taille des fonctions et procédures hors commentaires.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Autre exemple : règles pour le C

Les points principaux selon GNU

- Formatting : Formatting your source code.
- Comments : Commenting your work.
- Syntactic Conventions : Clean use of C constructs.
- Names : Naming variables, functions, and files.
- System Portability : Portability among different systems.
- CPU Portability : Supporting the range of CPU types.
- System Functions : Portability and standard library.
- Internationalization : Techniques for internationalization.
- Character Set : Use ASCII by default.
- Quote Characters : Use "..." or '...' in the C locale.
- Mmap : How you can safely use mmap.

http://www.gnu.org/prep/standards/standards.html

http://www.gnu.org/prep/standards/standards.html#Writing-C

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Rappels de codage

- Fortran :http://www.math.u-bordeaux1.fr/~lmieusse/PAGE_WEB/ENSEIGNEMENT/cours_f90.pdf
- C++ :http://www.math.u-bordeaux1.fr/~ksantugi/downloads/CPlusPlus/PolyCxx.pdf

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Environnement de travail

Compilateur et librairies

A chaque version de code est associé :

- des distributions compatibles (LINUX/MAC/WINDOWS)
- un compilateur (Intel,GNU),
- des librairies de calcul/paralléilisme (OpenMP,MPI,Petsc)
- des versions de langage (C,C++,Fortran,Html...)
- des versions d'outils de dépouillement et visualisation
- des versions des outils de mise en donnée

exemple pour openfoam

http://www.openfoam.org/download/ubuntu.php

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Gestion des sources

A quoi çà sert ?

La gestion de source permet un suivi temporel de l'évolution des sources au cours du temps en gardant une trace de toute l'historique. Ce faisant cela permet de coordonner le travail de l'équipe code.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Les deux outils principaux de gestion des sources

SVN vs. GIT

- SVN : Apache subversion (outil centralisé)
  https://fr.wikipedia.org/wiki/Apache_Subversion
  https://subversion.apache.org/
  Mode d'emploi : http://svnbook.red-bean.com/nightly/fr/svn-book.html
- GIT (outil décentralisé)
  https://fr.wikipedia.org/wiki/Git
  https://git-scm.com/

Voir https://fr.wikipedia.org/wiki/Logiciel_de_gestion_de_versions pour un panorama plus complet.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Compilation

Différentes options de compilation sont disponibles pour un code :
pour gcc :
urlhttps ://gcc.gnu.org/onlinedocs/gcc/Option-Summary.html

Les options de compilation permettent de produire généralement deux versions :
-une version debug
-une version optimisée (O2/O3)

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Exemple de librairies

Librairies mathématiques

- Petsc (solveurs de système) :http://www.mcs.anl.gov/petsc/
- Scotch (mailleurs) :https://www.labri.fr/perso/pelegrin/scotch/

Librairies informatiques

- open-mpi :http://www.open-mpi.org/
- cuda :https://developer.nvidia.com/gpu-accelerated-libraries

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Outils de suivi :

Définition

L'outil de suivi permet de communiquer entre les différents partenaires de projet en définissant des tâches à accomplir en un temps donné. Il permet de tracer les évolutions dans le planning et d'avoir une vision globale de l'état du projet.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Outils de suivi : exemples

Gestion de projet

https://fr.wikipedia.org/wiki/Logiciel_de_gestion_de_projets

Un exemple : Jira

https://fr.atlassian.com/software/jira

Essai gratuit sur :
https://fr.atlassian.com/software/jira/download?b=j

Démonstrateur : https://jira.atlassian.com/projects/DEMO/issues/DEMO-4589?filter=allopenissues

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Déroulé du cours

1 Objectifs du cours

2 Qu'est-ce qu'un code collaboratif ?

3 Outils de développement en commun

4 Outils de debug

5 Outils d'optimisation

6 Entrées/Sorties

7 Non-regression et validation

8 Service aux utilisateurs : dépannage/documentation

9 Conclusions

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Débuggage : qu'est ce que c'est ?

Qu'est-ce qu'un bug ?

C'est un ensemble de commandes informatiques qui font que le code ne rend pas le résultat escompté. Les bugs informatiques les plus courants :

- mauvais typage de variable (entier/réel)
- mauvaise allocation mémoire
- numérotation C et Fortran confondues
- coquilles dans le code (utilité de travailler à plusieurs...)

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Bugs algorithmiques

Les bugs physico-numériques les plus courants :

- données d'entrée non compatibles avec les modèles
- données d'entrée non compatibles avec le numérique
- erreurs d'arrondi non prévues
- sorties non conformes avec l'état du code
- plein d'autres que vous allez faire....

Solution

tester et encore tester son code...

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Les prints/write

C'est l'outil le plus simple il permet de trouver les bugs les plus simples : le problème est bien localisé et ne nécessite pas de tracer tout le code pour suivre l'historique des variables. C'est généralement suffisant pour les codes d'étude sans structure informatique trop complexe.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Les outils gratuits

Debugger de base

- gdb :http://www.gnu.org/software/gdb/
- idb :https://software.intel.com/en-us/articles/idb-linux

Plus d'infos sur
https://fr.wikipedia.org/wiki/Débogueur

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Valgrind (wikipedia)

Que l'on n'utilise pas de valeurs ou de pointeurs non initialisés ;
Que l'on n'oublie pas de libérer la mémoire allouée. Des options permettent de connaître avec précision les zones de mémoire qui sont perdues ;
Que l'on passe des arguments valides à certaines fonctions de la bibliothèque standard comme la fonction memcpy().

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Les outils payants (publicité totalview)

TotalView est un débogueur pour applications écrites en C/C++ ou Fortran.TotalView permet également de déboguer du code CUDA sous architecture Tesla et Fermi. TotalView permet aussi bien le contrôle d'un processus monothread que le contrôle de plusieurs processus multithreadés (synchrones ou asynchrones) en simultané. De plus, TotalView supporte le débogage d'applications parallèles s'appuyant notamment sur MPI, MPICH, Open MPI et OpenMP, que ce soit en mode interactif ou différé.

http://www.roguewave.com/products-services/totalview

C'est le meilleur outil utilisable sur supercalculateurs polur le moment.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Déroulé du cours

1 Objectifs du cours

2 Qu'est-ce qu'un code collaboratif ?

3 Outils de développement en commun

4 Outils de debug

5 Outils d'optimisation

6 Entrées/Sorties

7 Non-regression et validation

8 Service aux utilisateurs : dépannage/documentation

9 Conclusions

Gprof

Gprof est un outil d'analyse ligne à ligne d'un code en Fortran/C/C++ afin d'en estimer le coût. Il fournit un fichier d'analyse qui guide le développeur pour améliorer les résultats. Il ne dispose pas d'interface graphique. On commence par recompiler le code en mode debug.

https://sourceware.org/binutils/docs/gprof/

Flat profile : Each sample counts as 0.01 seconds.

time	seconds	seconds	calls	ms/call	ms/call	name
33.34	0.02	0.02	7208	0.00	0.00	open
16.67	0.03	0.01	244	0.04	0.12	offtime
16.67	0.04	0.01	8	1.25	1.25	memccpy
16.67	0.05	0.01	7	1.43	1.43	write
16.67	0.06	0.01				mcount
0.00	0.06	0.00	236	0.00	0.00	tzset
0.00	0.06	0.00	192	0.00	0.00	tolower
0.00	0.06	0.00	47	0.00	0.00	strlen
0.00	0.06	0.00	45	0.00	0.00	strchr
0.00	0.06	0.00	1	0.00	50.00	main
0.00	0.06	0.00	1	0.00	0.00	memcpy
0.00	0.06	0.00	1	0.00	10.11	print
0.00	0.06	0.00	1	0.00	0.00	profil
0.00	0.06	0.00	1	0.00	50.00	report

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun
Règles de codage
Versionnage
Gestion des sources
Gestion de projet

Outils de debug

Outils d'optimisation
Exercice de debug et optimisation

Entrées/Sorties
Entrées
Sorties

Non-regression et validation

Gcov

Outil remplaçant gprof avec les dernières versions de gcc.

https://gcc.gnu.org/onlinedocs/gcc/Gcov.html#Gcov

exemple :

gcc -fprofile-arcs -ftest-coverage -g sample.c -o sample

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Outils payants

- Instrumenter le code à la main pour compter le temps passé dans les routines (coûteux en temps) : permet d'analyser le code en optimisé.
- Vtune (Intel) :https://software.intel.com/en-us/intel-vtune-amplifier-xe

Résolution d'une équation stochastique

Contexte

On veut simuler un retour vers l'équilibre en température interne de molécules de gaz diatomiques. L'équation à résoudre est la suivante

$$\partial_t f = \frac{2}{\tau} \nabla_E \cdot ((E - RT) + \partial_E (ERTf))$$

où $f(t, E)$ représente le nombre de nombre de molécules avec l'énergie E à l'instant t. et τ un temps caractéristique du phénomène $R = 280SI$ la constante du gaz parfait et T la température interne moyenne du gaz.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage
Versionnage
Gestion des sources
Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées
Sorties

Non-regression et validation

Algorithme de résolution

Algorithme

1. Tirer au sort 500000 énergies E_p de 100000J à 200000J
2. calculer $T_{int} = moyenne(Energie)/R$
3. Entre t et $t + dt$ l'alogrithme est :

$$E_p^{n+1} = \frac{1}{1 + 2\frac{dt}{\tau}} \left(E_p^n + \frac{RTdt}{\tau}(1 + \sigma^2) + 2\sqrt{\frac{dt}{\tau} RTE_p^n}\sigma \right)$$

où σ est un nombre aléatoire gaussien et dt le pas de temps.

Cas-test

- $\tau = 5s$
- $dt = 1/100s$
- $N_p = 500000$: nombre de particules
- $T = 300, R = 280$

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Programmation

Codage

- Coder les différentes étapes.
- Stocker à chaque itération T_int pour suivre son évolutiion
- A la fin écrire le tableau d'évolution de T_{int}
- Créer un histogramme de taille 100 des énergies internes

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Optimisation

Utilisation de gprof

- Compiler en option -pg -O0
  ex : $gcc - Wall - pg\ test\_gprof.c\ test\_gprof\_new.c - o\ test\_gprof$
- Executer le cas-test
- Analyse grossière :
  $gproftest\_gprofgmon.out > analysis.txt$
- Analyse fine :
  $gprof - l\ test\_gprofgmon.out > analysisfine.txt$

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Déroulé du cours

1 Objectifs du cours

2 Qu'est-ce qu'un code collaboratif ?

3 Outils de développement en commun

4 Outils de debug

5 Outils d'optimisation

6 Entrées/Sorties

7 Non-regression et validation

8 Service aux utilisateurs : dépannage/documentation

9 Conclusions

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Données

Types de données

Deux types principaux de données existent :
-les données utilisateurs,
-les données développeurs.

Les deux coexistent sans forcément être accessibles à tout le monde. Dans l'idéal une interface graphique existe pour générer les fichiers de données comme dans Fluent : elle s'assure de la compatibilité des données. Sinon ce travail devra être effectué par le code à la lecture du fichier de données

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Les données utilisateurs

Données physiques

- les modèles physiques utilisés
- les constantes associés

Données numériques

- schéma numérique
- gestion du pas de temps
- rustines utilisateurs -> obligation d'avoir un résultat

Données d'utilisation

- maillage utilisé
- rythme de sortie/protection
- nombre de processeur
- version d'executable

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Les données développeurs

Données

- version debug/executable en cours de développement
- modèles physico-numériques en cours de développement
- déblocage de certaines options incompatibles jusque là
- rustines de développement
- modèles "fantômes"

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Sorties 1D

Les outils

On peut écrire les données en format brut et les importer dans ces logiciels.

- xmgrace
  http://mintaka.sdsu.edu/reu/grace.tutorial.html
- scilab/matlab
  http://www.scilab.org/fr

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Sorties : visualisation

Les logiciels

- Visit https://wci.llnl.gov/simulation/computer-codes/visit/
- Paraview http://www.paraview.org/

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Sorties : dépouillement

Création d'outils ad-hoc

Si aucun produit disponible ne correspond au besoin il ne reste plus qu'à développer son propre outil....

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Déroulé du cours

1 Objectifs du cours

2 Qu'est-ce qu'un code collaboratif ?

3 Outils de développement en commun

4 Outils de debug

5 Outils d'optimisation

6 Entrées/Sorties

7 Non-regression et validation

8 Service aux utilisateurs : dépannage/documentation

9 Conclusions

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Non régression

Définition

La non-régression est le processus garantissant que l'évolution du code va dans le bon sens. C'est un ensemble de cas tests très simples permettant de tester tous les modèles implémentés dans le code. Ces cas-tests doivent aussi tester les entrées/sorties du code afin de garantir la pérennité du code. La non-régression doit par conséquent aussi être mise sous gestionnaire de sources.

Outil de non régression

C'est un outil permettant de lancer rapidement tous les cas-tests associés à la non-régression et ce afin de permettre un développement sûr tout le long de la vie du code.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Validation

Définition

La validation est le processus permettant de vérifier que les modèles implémentés permettent de résoudre le problème posé. Ce processus se base sur une comparaison à des doonées externes au code (expériences/autres codes).

Outil de validation

C'est un outil permettant de lancer rapidement tous les cas-tests associés à la validation et ce afin de permettre un développement sûr tout le long de la vie du code. Ces cas-tests sont ceux qui sont mis en avant par les edéveloppeurs pour promouvoir leur code.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Déroulé du cours

1 Objectifs du cours

2 Qu'est-ce qu'un code collaboratif ?

3 Outils de développement en commun

4 Outils de debug

5 Outils d'optimisation

6 Entrées/Sorties

7 Non-regression et validation

8 Service aux utilisateurs : dépannage/documentation

9 Conclusions

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Dépannage

Définition

C'est le processus reliant les développeurs et les utilisateurs pour débugger le code. Il est généralement tracé par les outils de suivi de code. Le dépannage peut n'être qu'une demande d'évolution des fonctionnalités du code

Outils associés

Les outils de suvi de code type Jira sont généralement employés pour cela. Pour des raisons de confidentialité, des outils internes peuvent être développés traçant les rapports entre développeurs et utilisateurs. L'essentiel est que chaque demande soit traitée.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Documentation technique

Versionnage/Plan de développement

Le versionnage et le plan de développement doivent faire l'objet de mises à jour documentaires régulières.

Rapports techniques

Dans l'idéal des rapports à usage interne,et/ou externe doivent être produits à chaque nouveau développement (physique/numérique/informatique).

Exposés

Les exposés de présentation et prise en main du code doivent être aussi disponibles.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Documentation informatique : Doxygen

Doxygen

C'est un outil gratuit permettant de créer une documentation automatique de code analysant les variables, les routines et les include associés au projet.

http://www.stack.nl/~dimitri/doxygen/

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Documentation en ligne

Page web

C'est le minimum requis en terme de documentation. Elle peut être produite à partir de convertisseurs de fichiers pdf à html ou autres

Contact

Il faut s'assurer de pouvoir correspondre facilement avec les utilisateurs

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Communication

Formation

Pour promouvoir un code il est essentiel de former les gens.

Communication

Participation et présentation des réultats obtenus à des congrès ad-hoc.

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Déroulé du cours

1 Objectifs du cours

2 Qu'est-ce qu'un code collaboratif ?

3 Outils de développement en commun

4 Outils de debug

5 Outils d'optimisation

6 Entrées/Sorties

7 Non-regression et validation

8 Service aux utilisateurs : dépannage/documentation

9 Conclusions

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Conclusions & Perspectives

L 'objectif de ce cours est de montrer l'étendue des travaux à mener durant la vie d'un code. Le plus dur est de trouver le savant dosage permettant de développer rapidement tout en garantissant une certaine qualité de développement. Ce compromis est essentiel et doit pouvoir être revu en fonction des besoins au cours du temps

Objectifs du cours

Qu'est-ce qu'un code collaboratif ?

Outils de développement en commun

Règles de codage

Versionnage

Gestion des sources

Gestion de projet

Outils de debug

Outils d'optimisation

Exercice de debug et optimisation

Entrées/Sorties

Entrées

Sorties

Non-regression et validation

Objectifs pour le projet code

- versionner le code
- profiler le code et montrer le gain obtenu
- définir des cas tests de validation et non-régression
- créer une documentation associée (page web/guide des sources)