

CORRECTION

|                          |   |                          |   |                          |   |                          |   |                          |   |                          |   |                          |   |                          |   |
|--------------------------|---|--------------------------|---|--------------------------|---|--------------------------|---|--------------------------|---|--------------------------|---|--------------------------|---|--------------------------|---|
| <input type="checkbox"/> | 0 | <input type="checkbox"/> | 0 | <input type="checkbox"/> | 0 | <input type="checkbox"/> | 0 | <input type="checkbox"/> | 0 | <input type="checkbox"/> | 0 | <input type="checkbox"/> | 0 | <input type="checkbox"/> | 0 |
| <input type="checkbox"/> | 1 | <input type="checkbox"/> | 1 | <input type="checkbox"/> | 1 | <input type="checkbox"/> | 1 | <input type="checkbox"/> | 1 | <input type="checkbox"/> | 1 | <input type="checkbox"/> | 1 | <input type="checkbox"/> | 1 |
| <input type="checkbox"/> | 2 | <input type="checkbox"/> | 2 | <input type="checkbox"/> | 2 | <input type="checkbox"/> | 2 | <input type="checkbox"/> | 2 | <input type="checkbox"/> | 2 | <input type="checkbox"/> | 2 | <input type="checkbox"/> | 2 |
| <input type="checkbox"/> | 3 | <input type="checkbox"/> | 3 | <input type="checkbox"/> | 3 | <input type="checkbox"/> | 3 | <input type="checkbox"/> | 3 | <input type="checkbox"/> | 3 | <input type="checkbox"/> | 3 | <input type="checkbox"/> | 3 |
| <input type="checkbox"/> | 4 | <input type="checkbox"/> | 4 | <input type="checkbox"/> | 4 | <input type="checkbox"/> | 4 | <input type="checkbox"/> | 4 | <input type="checkbox"/> | 4 | <input type="checkbox"/> | 4 | <input type="checkbox"/> | 4 |
| <input type="checkbox"/> | 5 | <input type="checkbox"/> | 5 | <input type="checkbox"/> | 5 | <input type="checkbox"/> | 5 | <input type="checkbox"/> | 5 | <input type="checkbox"/> | 5 | <input type="checkbox"/> | 5 | <input type="checkbox"/> | 5 |
| <input type="checkbox"/> | 6 | <input type="checkbox"/> | 6 | <input type="checkbox"/> | 6 | <input type="checkbox"/> | 6 | <input type="checkbox"/> | 6 | <input type="checkbox"/> | 6 | <input type="checkbox"/> | 6 | <input type="checkbox"/> | 6 |
| <input type="checkbox"/> | 7 | <input type="checkbox"/> | 7 | <input type="checkbox"/> | 7 | <input type="checkbox"/> | 7 | <input type="checkbox"/> | 7 | <input type="checkbox"/> | 7 | <input type="checkbox"/> | 7 | <input type="checkbox"/> | 7 |
| <input type="checkbox"/> | 8 | <input type="checkbox"/> | 8 | <input type="checkbox"/> | 8 | <input type="checkbox"/> | 8 | <input type="checkbox"/> | 8 | <input type="checkbox"/> | 8 | <input type="checkbox"/> | 8 | <input type="checkbox"/> | 8 |
| <input type="checkbox"/> | 9 | <input type="checkbox"/> | 9 | <input type="checkbox"/> | 9 | <input type="checkbox"/> | 9 | <input type="checkbox"/> | 9 | <input type="checkbox"/> | 9 | <input type="checkbox"/> | 9 | <input type="checkbox"/> | 9 |

← codez votre numéro d'étudiant ci-contre, et inscrivez votre nom et prénom ci-dessous.

|                |
|----------------|
| Nom : .....    |
| Prénom : ..... |

---

**LIFAP4 Conception et développement d'applications : Examen (CC mi-parcours) du 10 mars 2020 (90 minutes)**

Aucun document autorisé, sans calculatrice, sans téléphone

**Attention : noircir complètement à l'encre la ou les cases cochées (une croix ne suffit pas).** Les questions marquées d'un trèfle peuvent avoir 0, 1 ou plusieurs choix corrects. Ce sera le cas de toutes les questions dans ce sujet.

---

**Question 1 ♣** Segmentation fault (core dumped) signifie

- une erreur de segmentation.
- que cette application à causer une erreur due à la mémoire.
- une erreur à l'exécution.
- que le système d'exploitation a stoppé cette application.
- une erreur à la compilation.
- une erreur de Doxygen.

**Question 2** Avec Git, pour récupérer une copie de travail dans le répertoire courant, la commande est

- git clone http://forge.univ-lyon1.fr/monprojet
- git download http://forge.univ-lyon1.fr/monprojet
- git checkout http://forge.univ-lyon1.fr/monprojet

**Question 3** Avec Git, la commande 'git commit'

- permet d'envoyer les changements des fichiers de travail vers le dépôt local et vers le dépôt distant.
- permet d'envoyer le dépôt local vers le dépôt distant.
- permet d'envoyer uniquement les changements des fichiers de travail vers le dépôt local.

**Question 4** Avec Git, la commande 'git log'

- enregistre mes commentaires dans un journal de log.
- affiche l'historique des commentaires entrés lors des commit.
- enregistre mes changements dans un journal de log.

**Question 5 ♣** La directive ci-dessous

```
#ifndef _CAMION_H_
#define _CAMION_H_
...
#endif
```

- se trouve dans le fichier 'Voiture.h'.
- sert à éviter les redéfinitions de types/classes/fonctions.
- provoquera une erreur de compilation si elle n'est pas respectée.
- provoquera une erreur d'exécution.
- se trouve dans le fichier 'Camion.h'.
- se trouve dans le fichier 'Camion.cpp'.

**Question 6 ♣** Ce code

```
int *a = new int[10];
a[0] = 12;
delete [] a;
if (a[0]==12) cout<<"Vrai";
```

- est un code correcte.
- peut provoquer un 'segmentation fault (core dump)'.
- provoquera une erreur de compilation.
- provoquera une erreur avec Valgrind.

**Question 7 ♣** Ce code

```
0: class Truc
1: {
2:     int a;
3:     int getA() const { return a; }
4: }
```

- provoquera une erreur avec un débogueur.
- provoquera une erreur de compilation.
- provoquera une erreur avec Valgrind.
- provoquera un 'segmentation fault (core dump)'.
- ne provoquera pas d'erreur.

**Question 8 ♣** Avec Git, je crée un nouveau fichier 'src/fic.cpp', quelle(s) commande(s) dois-je entrer pour que les autres développeurs puissent récupérer ce fichier ? (ne vous préoccupez pas de l'ordre, cochez toutes les commandes nécessaire).

- git add
- git commit -a src/fic.cpp
- git commit -m "ajout de fic" .
- git push
- git add src/fic.cpp
- git commit
- git commit fic.cpp
- git a src/fic.cpp

**Question 9 ♣** Avec Git, le dépôt local

- est l'ensemble des fichiers sur mon disque dur sur lesquels je travaille directement.
- se trouve dans le répertoire '.git'.
- se trouve sur le serveur Git.
- contient tout l'historique des versions.

CORRECTION

**Question 10 ♣** Cette commande 'g++ -ggdb -o projet main.o -L/usr/lib -IABC'

- '-L/usr/lib' indique un chemin vers les bibliothèques (.a, .so ou .dll).
- crée un exécutable en assemblant les fichiers objets.
- '-L/usr/lib' indique un chemin vers les fichiers d'entêtes (.h).
- effectue la phase de compilation.
- effectue la phase d'édition de lien.
- demande la production de fichiers objets (.o).
- '-IABC', indique qu'il faut inclure la bibliothèque (library) ABC à l'exécutable.
- '-IABC' indique qu'il faut inclure le ABC.h à la compilation.

**Question 11 ♣** Avec Git, quelle commande demande un accès réseau ?

- git push
- git pull
- git commit

**Question 12 ♣** Doxygen analyse les commentaires commençant par

- /\*\*
- /\*=
- /\*
- ///

**Question 13 ♣** Dans le module Image réalisé en LIFAP4, pour récupérer le Pixel original de coordonnée (x,y), quelle est la bonne manière d'écrire la fonction dans classe Image ?

- Pixel getPix(int x, int y);
- Pixel\* getPix(int x, int y);
- Pixel& getPix(int x, int y);

**Question 14** Avec Git, le dépôt distant

- se trouve sur le serveur Git.
- est l'ensemble des fichiers sur mon disque sur lequel je travaille.
- se trouve dans le répertoire '.git'.

**Question 15** Avec un 'git commit' quelle option permet d'ajouter un commentaire ?

- git commit -m "le commentaire"
- git commit -c "le commentaire"
- git commit -co "le commentaire"
- On ne peut pas inclure de commentaire directement.

**Question 16** Avec Git, la copie de travail

- se trouve sur le serveur Git.
- est l'ensemble des fichiers sur mon disque dur sur lesquels je travaille directement.
- se trouve dans le répertoire '.git'.
- contient uniquement des fichiers gérés par Git.

**Question 17 ♣** Cochez les actions qui sont exécutées classiquement par une boucle d'événement dans une IHM

- Récupère et traite des événements du clavier.
- Récupère et traite des événements de la souris.
- Exécute des actions automatiquement en fonction du temps écoulé.
- Envoie des événements à un autre programme.
- Affiche l'application.
- Sauvegarde toutes les données dans un fichier d'historique.
- Ouvre une fenêtre.
- Pose des questions à l'utilisateur.
- Efface l'écran.

CORRECTION

**Question 18 ♣** SDL permet facilement

- de jouer des sons.
- d'ajouter des menus à une application.
- d'afficher des images.
- de récupérer les touches claviers enfoncées par l'utilisateur.
- de faire saisir à l'utilisateur du texte.

**Question 19 ♣** Même si un programme marche, est-ce que Valgrind peut servir quand même ?

- Non
- Oui

**Question 20 ♣** Dans un Makefile, il y a des règles

- sur 2 lignes.
- comportant la règle de production sur la 2e ligne.
- comportant les dépendances à gauche sur la 1ere ligne.
- comportant des espaces à gauche sur la 2e ligne.
- comportant la cible à gauche sur la 1ere ligne.
- comportant la cible à droite sur la 1ere ligne.
- comportant une tabulation à gauche sur la 2e ligne.
- comportant les dépendances à droite sur la 1ere ligne.

**Question 21 ♣** Pour comprendre, inspecter et déboguer la gestion de la mémoire d'un programme je peux utiliser

- les erreurs de g++.
- un débogueur.
- Doxygen.
- Valgrind.
- Git.
- des 'cout/printf' dans les constructeurs/destructeurs/accesseurs.

**Question 22 ♣** Avec un débogueur je peux

- lancer une série de tests dans un script.
- mettre des points d'arrêts.
- valider des correctifs.
- corriger les erreurs de compilation.
- retrouver l'historique des exécutions.
- obtenir des statistiques d'utilisation de chaque fonction.
- comprendre ce que fait réellement le programme.
- inspecter des variables au cours de l'exécution.

**Question 23 ♣** Le type C++ pour déclarer un entier dont les valeurs seront comprise en 0 et 255 incluse est

- unsigned char
- int
- short
- char

**Question 24** Je veux supprimer un fichier de mon répertoire de travail et que git le prenne en compte (il restera dans l'historique des versions passées) :

- git rm lefichier
- rm lefichier
- je ne peux pas simplement

**Question 25 ♣** La fonction 'SDL\_RenderCopy' de SDL2

- permet de sélectionner une partie de la texture source et une taille du rectangle destination.
- copie une texture sur la fenêtre principale ou tout autre texture cible.
- copie une surface sur la fenêtre principale ou tout autre surface cible.

CORRECTION

**Question 26 ♣** Le cahier des charges

- indique les modalités d'exécution pour la réalisation avec les éléments/versions à livrer.
- est un contrat entre le client et le prestataire de service.
- propose des cycles AGILE de conception.
- définit le nombre de niveau qu'aura le jeu.
- définit exhaustivement les spécifications de l'application.
- indique le nom des développeurs pour chaque tâche.

**Question 27** Avec Git, la commande 'git push'

- permet d'envoyer les changements des fichiers de travail dans le dépôt local et dans le dépôt distant.
- permet d'envoyer le dépôt local dans le dépôt distant.
- permet d'envoyer uniquement les changements des fichiers de travail dans le dépôt local.

**Question 28** Avec Git, la commande git pull

- permet de récupérer les changements du dépôt distant dans le dépôt local et de mettre à jour les fichiers de travail
- permet de répercuter uniquement les changements du dépôt local dans les fichiers de travail
- permet de récupérer uniquement les changements du dépôt distant dans le dépôt local

**Question 29 ♣** Ce code

```
assert(a!=0);
```

- indique au débogueur de tester si la valeur de a est différente de 0.
- indique au compilateur de tester si la valeur de a est différente de 0.
- indique au débogueur de mettre une valeur différente de 0 dans a.
- arrête l'application si 'a' vaut 0.
- s'assure à l'exécution que la variable 'a' est différente de 0.

**Question 30 ♣** Mon programme ne fait pas ce qu'il devrait faire, mais ne plante pas, quelle solution ai-je pour le corriger ?

- Comprendre ce qu'il fait réellement avec des affichages 'cout/printf'.
- Comprendre ce qu'il fait réellement avec un débogueur.
- Le tester avec Valgrind.
- De corriger les erreurs de compilation.

**Question 31 ♣** Ce code

```
0: class Truc
1: {
2:     int a;
3:     void faire() const { a=12; }
4: }
```

- provoquera une erreur de compilation.
- provoquera un 'segmentation fault (core dump)'.
- provoquera une erreur avec Valgrind.
- ne provoquera pas d'erreur.
- provoquera une erreur avec un débogueur.

**Question 32 ♣** La balise \brief sert à

- documenter une structure/class.
- donner une description courte.
- documenter une fonction.

CORRECTION

**Question 33 ♣** Une bonne manière de passer en donnée un argument de type `Camion` (avec `sizeof(Camion)>8`) à une fonction est

- `void bonneFonction(const Camion& c);`
- `void bonneFonction(Camion& c);`
- `void bonneFonction(Camion c);`

**Question 34** Avec Git quelle est l'opération inverse de 'git add'?

- `rm`
- `git reset HEAD`
- `git rm fichier`

**Question 35 ♣** Cochez les balises Doxygen valides.

- `\save` pour sauvegarder toutes les données dans un fichier d'historique.
- `\brief` pour donner une description courte.
- `\fn` pour documenter une fonction.
- `\param` pour documenter un paramètre de fonction/méthode.
- `\ask` pour poser une question à l'utilisateur.
- `\run` pour décrire l'exécution du code.
- `\struct` `\class` pour documenter une structure/class C/C++.

**Question 36 ♣** Un 'callback'

- est représenté par un pointeur de fonction.
- est une fonction récursive.
- est une fonction qui sera appelée par la boucle d'événement.

**Question 37 ♣** La commande `'g++ -Wall -ggdb -c main.cpp -I/usr/include'`

- `'-Wall'` provoque l'affichage de tous les messages de prévention (ambiguïtés, type cast, oublis, etc.).
- `'-ggdb'` indique d'inclure les informations sur le programme pour le débogueur.
- effectue la phase de compilation.
- `'-c'` demande la production de fichiers objets (.o).
- `'-I/usr/include'` indique un chemin vers les bibliothèques (.a, .so ou .dll).
- `'-I/usr/include'` indique un chemin vers les fichiers d'entête (.h).
- crée un exécutable.
- effectue la phase d'édition de lien.
- `'-ggdb'` indique d'exécuter le programme avec gdb.

**Question 38 ♣** 'core dump' signifie que le système a

- sauvegardé tout l'état de la mémoire dans un fichier 'core'.
- sauvegardé l'historique des versions de code dans un fichier 'core'.
- sauvegardé l'historique des instructions dans un fichier 'core'.

CORRECTION

**Question 39 ♣** Pour gérer toutes les notes de LIFAP4 nous écrivons la classe suivante :

```
0: class LIFAP4
1: {
2: public:
3:     LIFAP4() : nb_notes(0), taille_notes(0) {}
4:     int sauver(const string& filename) const;
5:     void ajouterUneNote(int n) { CODE }
6: private:
7:     int *notes;
8:     int nb_notes, taille_notes;
9: };
```

- Ligne 7 : notes est un tableau.
- Ligne 7 : notes est un pointeur.
- Ligne 4 : le 'const' à la fin signifie que la fonction renvoie un entier constant.
- Ligne 4 : le 'const' à la fin signifie que la fonction ne modifie pas les données membres de la classe.
- Un destructeur ne sera pas utile dans cette classe.
- ligne 5 : l'instruction CODE fera une (re)allocation dynamique si le tableau est plein.
- Ligne 3 : il manque une/des initialisation(s).

**Question 40 ♣** Cochez les affirmations appartenant à la méthode de conception AGILE

- On ne revient pas sur un choix quoi qu'il arrive.
- La méthode la plus efficace pour transmettre l'information est une conversation en face à face.
- Les processus complexes sont importants pour résoudre des problèmes complexes.
- Les processus AGILE promeuvent un rythme de développement tranquille mais sûr.
- Bâissez le projet autour de personnes motivées.
- Un logiciel fonctionnel est la meilleure unité de mesure de la progression du projet.

**Question 41 ♣** Doxygen

- est sous licence libre.
- récupère et traite des actions de la souris.
- utilise la grammaire du langage dans lequel est écrit le code source, ainsi que les commentaires.
- est un générateur de documentation capable de produire une documentation à partir du code source d'un programme.
- gère les versions du code d'un programme.

**Question 42 ♣** Qt permet facilement

- de jouer des sons.
- de faire saisir à l'utilisateur du texte.
- d'ajouter des bar d'icônes à une application.
- d'afficher des images.
- d'ajouter des menus à une application.
- de réagir à un événement clavier.

**Question 43 ♣** Cochez les méthodes de conceptions vues durant le cours :

- Cycle en U.
- Méthode AGILE.
- Cycle itératif.
- Cycle procédural.
- Cycle en itératif.
- Cycle en cascade.
- Cycle en V.
- Cycle en spiral.

## CORRECTION

**Question 44 ♣** Dans le module Image

```
Image im1(256,256);
```

```
Image* im2 = new Image(256,256);
```

- Le tableau de Pixel de im1 sera stocké dans le tas.
- im1 est stocké dans la pile.
- Le tableau de Pixel de im2 sera stocké dans le tas.
- Il faudra faire un delete [] im2;
- im1 est stocké dans le tas.
- im2 est stocké sur la pile.

**Question 45 ♣** Cochez les affirmations vraies à propos d'un diagramme de Gantt.

- La ligne d'en-tête représente les unités de temps les plus adaptées au projet (jours, semaines, mois etc.).
- Un diagramme de Gantt permet de visualiser dans le temps les diverses tâches composant un projet.
- Un diagramme de Gantt indique les pénalités en cas de retard dans le développement.
- La colonne de gauche du diagramme énumère toutes les tâches à effectuer.

**Question 46 ♣** La fonction 'SDL\_RenderPresent' de SDL2

- met à jour la fenêtre principale en rendant visible tous les affichages faits depuis l'appel précédent.
- bascule les deux buffers d'affichages.
- recopie le 2e buffer d'affichage dans le buffer visible.

**Question 47 ♣** Cochez les règles d'écriture de code importantes.

- Des lignes ne dépassant pas 80 colonnes.
- Plus de commentaire que de code.
- Code indenté.
- Toujours indenter avec des espaces et non des tabulations.
- Variables, fonctions et classes avec des noms clairs et explicites.
- Le code doit avoir le moins de ligne possible.