

INF2: Travail écrit 09.04.19 / RRH

Nom et prénom: Franchet: Thibard

Travail écrit no. 1

(Durée: 2 périodes)

Directives:

- ECRIVEZ VOS REPONSES DIRECTEMENT SUR LA DONNEE
- Ne pas dégrafer le document
- Vous pouvez écrire au crayon
- Seule documentation autorisée : la Quick Reference Card C++ (non annotée !)





Question 1 (1.25 point)

Compléter la partie notée <à compléter> du programme ci-après, de telle sorte que celui-ci affiche à l'exécution :

```
01.1900
11.2018
01.2019
Exception: mois doit etre un entier entre 1 et 12
```

```
<à compléter>
int main() {
   try {
      cout << Date() << endl;</pre>
   // Date d1 = 1; // Sans les commentaires, serait refusé à la compilation
      Date d2(11, 2018);
      cout << d2++ << endl;
      cout << ++d2 << endl;
     Date d3(13, 2019);
   } catch (const MoisNonValide& e) {
      cout << e.what() << endl;</pre>
   return EXIT SUCCESS;
```

- Tout le code à produire est censé figurer dans le même fichier que main
- Le code de main ne doit en aucun cas être modifié
- Le code doit être écrit sans faire usage d'exceptions prédéfinies donc on consolire

 La classe Date

 o est supposée ne comporter que 2 données-membres : mais (1-12) et année.
- - o est supposée ne comporter que 2 données-membres : mois (1-12) et annee
 - o ne doit comporter qu'une seule fonction *non* membre au plus
 - o ne doit déclarer que les fonctions strictement nécessaires à l'exécution du main proposé, et rien d'autre¹!
 - ¹ cela signifie, en particulier, qu'aucun accesseur ne doit être proposé.
- Aucune fonction de la classe *Date* ne doit être codée en ligne (inline)
- Appliquer l'encapsulation et éviter au maximum d'écrire du code redondant
- Il n'est pas demandé de commenter les diverses fonctions de la classe Date



Votre réponse : *Include Kiostreams *include costallibs * include < iorianips
* Induct strings using nerus pecu std: class Mais NonValide & // Mirite pas de exception voir renarque p. 2 constable + string what () const; private: string what arg; Moiston Valida: Mais Non Valida (const string & what arg): what arg (what arg) & string Pris Nor Valida in what () corst & return what-crs; friend ostreand grandor ex (ostreand this, const Date & rls); Le public explicit Date (unsigned nois = 1, unsigned annex = 1000);/ Détede quester ++ (int); / Manque noexcept ← private: unsigned ruis; unignet anne! V



Votre réponse (suite) :

Dete Dele marigned in tors one ance I tors reist ance ance ? ostreamle operator (lostream & lhs, const Duted who) } this = set fill (10') = set w(2) = nois = 1

reset fill (0') = set w(2) = nois = 1

return this; repetition invite Dated operator + 1 () {

annee += (unsigned) (ruis == 12); mols = (mols % 12) + 1; return * this; Date:

Och proctor + (int) {

Och trp = #this;

++ (*this);

++ (*this); Date: Date (unsigned nois, unsigned annee) { if (nois == 0 11 nois 712) } throw Argonat Non Valida | "Exception: swis doit of m"
"in offer ata 1 et 12"); this > ruis = nois;





0.625) Question 2 (0.75 point)

Le code ci-après est censé afficher à l'exécution :

```
J = 0.5
c1 : i = 1
c2 : i = 2
c3 : i = 1
c3 : i = 2
```

```
1 #include <cstdlib>
 2 #include <iostream>
 3 using namespace std;
 5 class C {
     friend ostream& operator<<(ostream& os, const C& c);</pre>
 7 public:
     C(int i);
     C(const C& c);
    C& operator=(const C& c);
int getI() const;
static double getJ() const;
10
11
13 private:
14
     const int i;
     static const double J = 0.5;
15
16 };
17 3/2/10 cont doble (1) (:) = 0.5;
18 Friend, ostream& operator < (ostream& os, const C& c) {
     return cout << "i = " << c.i;
21
22 C::C(int i) : i(i) {}
23
24 C::C(const C& c) : i(c.i) {}
29 }
30
31 int C::getI() const {return i;}
32 double C::getJ() garst {return J;}
C c1 = 1;
37
     C c2{2};
     C c3(c1);
38
    cout << "c1 : " << c1 << endl;
39
     cout << "c2 : " << c2 << endl;
40
     cout << "c3 : " << c3 << endl;
41
    c3 = c2;
cout << "c3 : " << c3 << endl;
42
4.3
44
     return EXIT SUCCESS;
45 }
```



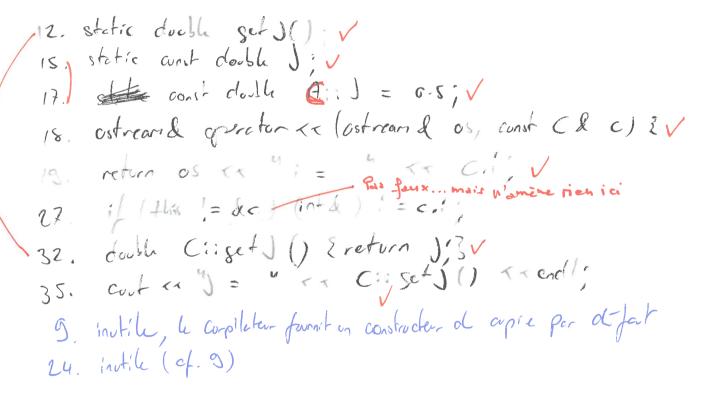
Ce code contient toutefois diverses erreurs ou maladresses.

Pour chacune d'entre elles, indiquer le numéro de la ligne fautive et proposer un correctif C++.

IMPORTANT

- La nature des 2 champs ("usuel constant" pour *i* et "static const" pour *J*) ne doit pas être modifiée
- Aucun champ ne doit être ajouté
- Aucune fonction amie ou fonction-membre ne doit être ajoutée
- Des points seront décomptés si des erreurs ou maladresses supplémentaires sont introduites

Votre réponse :

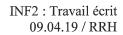




HEIG-VD Département TIC INF2 : Travail écrit 09.04.19 / RRH

Votre réponse (suite) :







Question 3 (1.25 point)



a) (0.45 pts)

Soient les définitions de fonctions suivantes :

```
template <typename T, typename U> void f(T, U) \{...\} // fonction 1 template <typename T> void f(T, T) \{...\} // fonction 2 template <typename T> void f(T, char) \{...\} // fonction 3 template <typename T> void f(T, double) \{...\} // fonction 4 void f(char, double) \{...\} // fonction 5 void f(int, double) \{...\} // fonction 6
```

et soient les déclarations de variables suivantes :

```
char c = 'A';
int i = 1;
float x = 2;
double y = 3;
```

Pour chacun des appels ci-dessous, indiquer soit quelle fonction est appelée, soit que l'appel est ambigu.

- Il n'est PAS demandé de justifier vos réponses ou de proposer un quelconque correctif
- Barème appliqué : +0.075 point pour une réponse correcte; 0 point en cas d'absence de réponse; -0.075 point pour une réponse incorrecte

1)	f(i, c);	foretien 3
	f(c, c);	ambigu
3)	f(i, x);	fonetion 1 V
4)	f<>(i, y);	forchion 4 V
5)	f< double >(c, c);	fonction 3 V
6)	f(i, (double)x);	forchion & V
		6



INF2: Travail écrit 09.04.19 / RRH



b) (0.5 pts)

Soient les définitions de fonctions suivantes :

et soient les déclarations de variables suivantes :

```
char c = 'A';
short s = 1;
int i = 2;
float x = 3;
```

Pour chacun des appels ci-dessous, indiquer soit quelle fonction est appelée, soit que l'appel est ambigu.

- Il n'est PAS demandé de justifier vos réponses ou de proposer un quelconque correctif
- Barème appliqué : +0.125 point pour une réponse correcte; 0 point en cas d'absence de réponse; -0.125 point pour une réponse incorrecte

1)	f(c, i, x);	fonction 7 V
2)	f(s, s, x);	Consige V
3)	f(i, s, x);	Josephon 8V
4)	f<>(c, i, s);	Jenetian 3V
		V.





c) (0.3 pts)

Soient les deux classes génériques suivantes :

```
template <typename T = int> class A {...};
template <typename T, typename U, int n = 10 > class B {...};
```

Pour chacune des instanciations ci-dessous, dire si celle-ci est correcte ou non.

IMPORTANT

- Il n'est PAS demandé de justifier vos réponses ou de proposer un quelconque correctif
- Barème appliqué : +0.1 point pour une réponse correcte; 0 point en cas d'absence de réponse; -0.1 point pour une réponse incorrecte

1)	B <float*,< th=""><th>float></th><th>b;</th></float*,<>	float>	b;

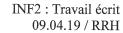
2) size t size = 20; B<int, array<float*, size>> b; Correcte V

Incorrecte V

Correcte V

3) B<float, vector<A<>>, 2*10> b;







Question 4 (1 point)

Compléter les 2 parties notées <*à compléter*> du programme ci-après, de telle sorte que celui-ci affiche à l'exécution :

```
nombre d'occurrences = 4
nombre d'occurrences = 3
```

```
#include <algorithm>
#include <cstdlib>
#include <cstring>
#include <iostream>
#include <vector>
using namespace std;
<à compléter 1>
template <typename T>
size_t nbOcc(const vector<T>& v, const Intervalle<T>& intervalle) {
   return (size_t) count_if(<à compléter 2>);
int main() {
   // Nombre de valeurs dans V1 comprises dans l'intervalle I1 semi-ouvert [1; 3[
  const V<int> V1{0, 1, 2, 3, 4, 3, 2, 1, 0};
const I<int> I1{1, 3};
   cout << "nbre d'occurrences = " << nb0cc(V1, I1) << endl;</pre>
   // Nombre.de valeurs dans V2 comprises dans l'intervalle I2 semi-ouvert ["A"; "D"]
  // ... ou dit autrement, nombre de prénoms commençant par A, B ou C const V<const char*> V2{"Albert", "Jack", "Bob", "Dylan", "Charles"};
  const I<const char*> I2{"A", "D"};
   cout << "nbre d'occurrences = " << nb0cc(V2, I2) << endl;</pre>
  return EXIT SUCCESS;
```

- Hormis les 2 parties notées <à compléter>, aucune ligne de code ne doit être ajoutée, modifiée ou supprimée du code proposé ci-dessus.
- La classe *Intervalle* doit <u>obligatoirement</u> être implémentée sous la forme d'une classe foncteur générique.



xà coptita 13 Votre réponse : terplate of penane Ts using V = vector < T> ; template a typeram TS class Intervalle & V public: Intervalle (const Td Min, cont Td Max): bus operator () (const I sturet) pant; privale:

Non! Telement (cf corrigien classe)

eunat Trin;

Const Trin;

Const Trinx;

Const Charte Charin templete etypenme Ts Intervalle (TS: Intervalle (const Td rin, const Td rin): rin(rin), rangow)8 template Appenance To bul Interallects: operator () (and TS etnat) const { return closet >= min && elvert as mux; Faux! la donnée précissit interelle seni, ouver 3 templete x> bud Intervallex const dur * s :: operator () (const dur * / elevent) const & return *elevent >= * min Id *elevent = * max; templake < typenane 13
Using I = Intervalle < Ts;

Votre réponse (suite): < A complet 2 >

teturn (size +) count-if (v. begin(), v. end(), intervalle); /

HEIG-VD Département TIC



Question 5 (0.75 point)

Que va afficher le programme ci-dessous si l'utilisateur saisit :

- 1) la valeur 1?
- 2) la valeur 2?

...

5) la valeur 5?

```
#include <cstdlib>
#include <exception>
#include <iostream>
#include <stdexcept>
using namespace std;
void onExit()
                      {cout << "onExit" << endl;}
void onTerminate() {cout << "onTerminate" << endl;</pre>
                       exit(EXIT FAILURE);}
void onUnexpected() {cout << "onUnexpected" << endl;}</pre>
void f(int n) throw (int) {throw (short) n;}
void g(int n) noexcept {throw n;}
int main() {
   atexit(onExit);
   set_terminate(onTerminate);
   set_unexpected(onUnexpected);
   cout << "Donnez un entier : ";</pre>
   cin >> n; // On suppose la saisie utilisateur OK
   try {
     switch (n) {
       case 1: throw 1;
       case 2: throw &n;
       case 3: try { f(n); }
                catch (short) {cout << "catch1" << endl;}</pre>
                catch (...) {cout << "catch2" << endl;}</pre>
                break;
       case 4: try { g(n); }
                catch (int) {cout << "catch3" << endl;}</pre>
                catch (...) {cout << "catch4" << endl;}</pre>
                break;
       case 5: try { throw runtime error("Oups!"); }
                catch (const overflow_error& e) {cout << e.what() << endl; throw 'A';}</pre>
                catch (const exception& e) {cout << e.what() << endl; throw 'B';}
catch (...) {cout << "catch5" << endl; throw 'C';}</pre>
                break;
      }
  catch (int&) {cout << "catch6" << endl; throw 'D';}
catch (int) {cout << "catch7" << endl; throw 'E';}</pre>
                        {cout << "catch7" << endl; throw 'E';}
  catch (int)
  catch (const int&) {cout << "catch8" << endl;}</pre>
  catch (const int) {cout << "catch9" << endl;}</pre>
  catch (char& e)
                       {cout << e << endl;}
  cout << "Fin main" << endl;</pre>
  return EXIT SUCCESS;
```



INF2: Travail écrit

09.04.19 / RRH

Vos réponses :

- 1) catch b on Terrinall on Exit V
- 2) on Terringte on Exit V
- 3) On Unexpected on Terminate on Exit
- 4) on Terminate on Exit
- 5) Oups
 B
 Tinnain V
 on Exit