

Nom/prénom :

no camipro :

Programmation 2015

Test final – série 1a + CORR

Durée 1h45.

Mettre votre nom/prénom/no camipro sur toutes les pages.

Aucun document, ni appareil électronique (calculatrice, natel, etc.) n'est autorisé.

Pas de feuille volante reprise, uniquement les feuilles distribuées seront reprises.

Si vous devez faire des hypothèses, notez-les.

Merci de signer cette première page

Signature :

Post-it avec no de place

Nom/prénom :

no camipro :

1. (20pts) La fonction **CommonElements()** trouve les valeurs communes des vector **A** et **B** passés en paramètre et les ajoute au vector **C**. Le vector **C** ne **contient qu'une seule fois** une valeur commune. La fonction retourne un code d'erreur si **A** et/ou **B** sont vides. Le vector **C** est vide lors de l'appel à la fonction.

Exemples

avec A = {1,2,2,3}, B = {1,2,2,0}, C={}
err= CommonElements (A,B,C)
retourne **err=0** et **C = {1,2}**

A = {1,2,2,3}, B = {}, C={}
err= CommonElements (A,B,C)
retourne **err=-2** et **C = {}**

Ecrire en C++ la fonction **CommonElementst ()** suivante:

```
int CommonElements(vector <int> A, vector <int> B, vector <int> &C)
```

retourne :

- 1 si **A** est vide
- 2 si **B** est vide
- 3 si **A** et **B** sont vides
- 0 autrement

3 x 5 pts pour les boucles OK

5 pts pour le reste (syntaxe, err, etc)

- 1 par index boucle faux
- 2 -3 -4, tentative gestion C faux (evt. -1 si err min. i.e. syntax)
- 5 pas de test pour enlever elem de C à double

- 1 gest. Err faux, return C -1, no return. -1
- 2 err return too early
- 1,-2 syntaxe trop éloignée du c++

-3 pts mid-term err (if else test last/first only elem)

```
int CommonElements(vector <int> A, vector <int> B, vector <int> &C) {  
    int found (0);  
    int ret(0);  
    if (A.size()<=0) ret+=-1;  
    if (B.size()<=0) ret+=-2;  
    else  
        for (int i(0);i<A.size();i++)  
            for (int j(0); j<B.size(); j++) {  
                if (A[i]==B[j]) {  
                    found=0;  
                    for (int k(0);k<C.size();k++)  
                        if (A[i]==C[k])  
                            found =1;  
                    if (found==0)  
                        C.push_back(A[i]);  
                }  
            }  
}
```

Nom/prénom :

no camipro :

```
} return ret;
```

Nom/prénom :

no camipro :

2. (10pts) Dans matlab, que valent A, B, C, D, E, F, G, H, J, K. **Les valeurs des variables se réfèrent aux valeurs des points précédents.**

```
A = linspace(10,1,10)
```

```
A = [10 9 8 7 6 5 4 3 2 1]
```

```
B = find(A>5)
```

```
B = [ 1 2 3 4 5 ]
```

```
C = A(B)
```

```
C = [ 10 9 8 7 6 ]
```

```
D = [C 6-B]
```

```
D = [10 9 8 7 6 5 4 3 2 1]
```

```
E = sum(D==A)
```

```
E = 10
```

```
F = B(end-1:end)
```

```
F = [ 4 5 ]
```

```
G = F'*F
```

```
G = [16 20 ; 20 25]
```

```
H = G(:,1)
```

```
H = [ 16 ; 20 ]
```

```
J = C(B(B<3))
```

```
J = [ 10 9 ]
```

```
K = J./J
```

```
K = [ 1 1 ]
```

Nom/prénom :

no camipro :

3. (4Pts) Que calcule la fonction matlab suivante :

Retourne MIN(b)

Donnez une brève explication

```
function a = maFunctionA(b)
    a = b(end);
    for index = 1:length(b)-1
        if b(index) < a
            a= b(index);
        end
    end
end
```

Que retourne `maFunctionA()` pour `b = [8, 7, 6, 7, 1, 4]` ?

1

4. (2 pts) Sans utiliser de boucle, écrire le code matlab équivalent au code C/C++ ci-après pour créer le vector **A**.

```
vector <int> A;
for (int a(0); a < 10 ; a += 2)
    A.push_back(a);
```

A= `0:2:8` ou `linspace(0,8,5)`

5. (4pts) A l'aide d'une boucle while, écrire le code c++ équivalent au code matlab ci-après pour la création du vector **B**.

B=linspace(1,2,5)

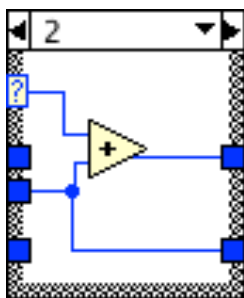
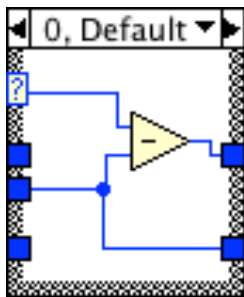
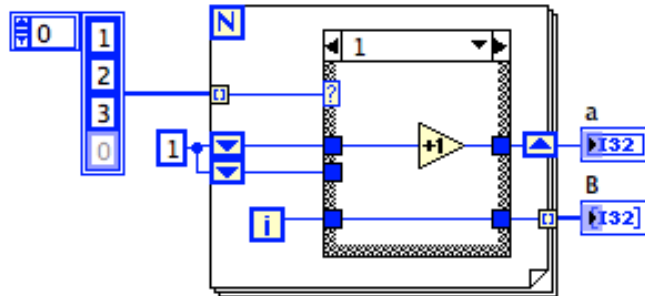
```
vector <intdouble> B;
```

```
for (double d(1) ; d<=2 ; d+=0.25)
    B.push_back(d) ;
```

Nom/prénom :

no camipro :

6. (6 pts) Que valent **a** et **B** après l'exécution du *diagram* ci-dessous, justifiez en notant les étapes intermédiaires?

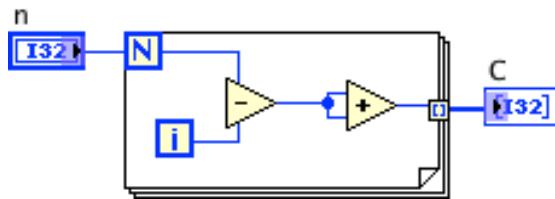


$a=1$
 $B=[0 \ 1 \ 2]$

Nom/prénom :

no camipro :

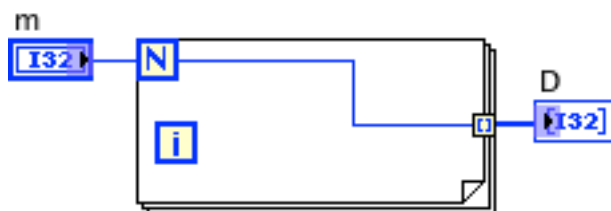
7. (6 pts) Ecrire en matlab (1 ligne) les codes équivalents aux *diagrams* ci-dessous pour la création des vecteur C, D, E.



C fonction de **n**

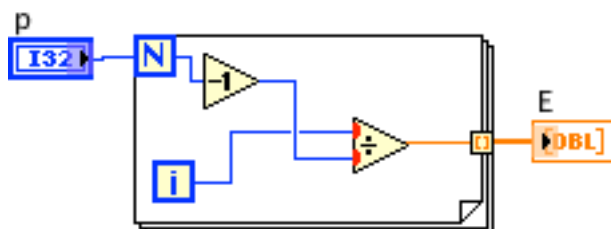
C = `linspace(2*n,2,n)` ou `2* linspace(n,1,n)` ou `2*n :-2 :2`

(1 borne fausse -1pt, 2 bornes fausses 0 pt)



D fonction de **m**

D = `linspace(m,m,m)`



E fonction de **p**

E = `linspace(0,1 ,p)`