

Matric.: Nom: Prénom: Gr: ...

Durée : 30 min

Test de TD

Algorithmique et Structures de Données Statique1

Exercice 1 (6 pts)

Dérouler l'algorithme suivant en donnant la valeur de chaque variable après exécution de chaque instruction et préciser ce qu'il affiche, pour la valeur $n=6$:

```
Algorithme test1;
  var i,j,n: entier;
Debut
  ecrire('Donner une valeur :');
  lire(n);
  i ← -1; j ← n;
  Tant que j >= n faire
    si i <= n alors
      i ← i+2;
      j ← j+3;
    fsi ;
    si i >= n alors
      j ← j-5;
      n ← n-1;
    fsi;
  ftq;
  ecrire('La valeur de i est :', i, ' et celle de j est :',j);
Fin.
```

Exercice 2 (9 pts)

Soient a, b et c trois entiers donnés. Écrire un algorithme ou un programme C qui affiche le plus grand écart entre 2 nombres pouvant être formé à partir de ces trois nombres.

L'écart entre deux nombre est égal à la valeur absolue de leur différence. Par exemple, l'écart entre 2 et 7 est égal à 5. L'écart entre 8 et -6 est égal à 14.

Exemple :

Le plus grand écart pouvant être formé à partir de 2, 7 et -3 est 10.

Réponse:

Exercice 1: Compléter ce tableau, barrer les lignes en plus s'il y en a.

Instruction	n	i	j	Instruction	n	i	j
lire(n)	6	?	?				
i ← -1							

Cet algorithme affichera : La valeur de i est : et celle de j est :

Matric.: Nom: Prénom: Gr: ...

Durée : 30 min

Test de TD

Algorithmique et Structures de Données Statique1

Exercice 1 (6 pts)

Dérouler l'algorithme suivant en donnant la valeur de chaque variable après exécution de chaque instruction et préciser ce qu'il affiche, pour la valeur $n=5$:

```
Algorithme test1;
  var i,j,n: entier;
Debut
  ecrire('Donner une valeur :');
  lire(n);
  i ← -1; j ← n;
  Tant que j >= n faire
    si i <= n alors
      i ← i+2;
      j ← j+3;
    fsi ;
    si i >= n alors
      j ← j-5;
      n ← n+1;
    fsi;
  ftq;
  ecrire('La valeur de i est :', i, ' et celle de j est :',j);
Fin.
```

Exercice 2 (9 pts)

Soient a,b et c trois entiers donnés. Écrire un algorithme ou un programme C qui affiche le plus petit écart entre 2 nombres pouvant être formé à partir de ces trois nombres.

L'écart entre deux nombre est égal à la valeur absolue de leur différence. Par exemple, l'écart entre 2 et 7 est égal à 5. L'écart entre 8 et -6 est égal à 14.

Exemple :

Le plus petit écart pouvant être formé à partir de 2, 8 et -3 est 5.

Réponse:

Exercice 1: Compléter ce tableau, barrer les lignes en plus s'il y en a.

Instruction	n	i	j	Instruction	n	i	j
lire(n)	5	?	?				
i ← -1							

Cet algorithme affichera : La valeur de i est : et celle de j est :

Durée : 30 min

Algorithmique et Structures de Données Statique1

Ecrire un programme C qui lit trois entiers, puis affiche celui qui a la plus grande valeur absolue. Par exemple, pour les nombres -5, 3, -12 votre programme doit afficher -12.

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

Durée : 30 min

Algorithmique et Structures de Données Statique1

Ecrire un programme C qui lit trois entiers, puis affiche celui qui a la plus petite valeur absolue. Par exemple, pour les nombres -5, -3, -12 votre programme doit afficher -3.

[illegible]

Matric.: Bonigé Lype Nom: Version 2 Prénom: 01/12/22 Gr: ...
 Durée : 30 min Test de TD

Algorithmique et Structures de Données Statique1

Exercice 1 (6 pts)

Dérouler l'algorithme suivant en donnant la valeur de chaque variable après exécution de chaque instruction et préciser ce qu'il affiche, pour la valeur $n=5$:

Algorithme test1;

var i,j,n: entier;

Debut

ecrire('Donner une valeur :');
lire(n);

$i \leftarrow -1$; $j \leftarrow n$;

Tant que $j \geq n$ faire

si $i \leq n$ alors

$i \leftarrow i+2$;

$j \leftarrow j+3$;

fsi ;

si $i \geq n$ alors

$j \leftarrow j-5$;

$n \leftarrow n+1$;

fsi;

ftq;

ecrire('La valeur de i est :', i, ' et celle de j est :', j);

Fin.

Exercice 2 (9 pts)

Soient a,b et c trois entiers donnés. Écrire un algorithme ou un programme C qui affiche le plus petit écart entre 2 nombres pouvant être formé à partir de ces trois nombres.

L'écart entre deux nombre est égal à la valeur absolue de leur différence. Par exemple, l'écart entre 2 et 7 est égal à 5. L'écart entre 8 et -6 est égal à 14.

Exemple :

Le plus petit écart pouvant être formé à partir de 2, 8 et -3 est 5.

Réponse:

Exercice 1: Compléter ce tableau, barrer les lignes en plus s'il y en a.

6pts puis chag
erreur -0,25

Instruction	n	i	j
lire(n)	5	?	?
$i \leftarrow -1$	5	-1	nd
$j \leftarrow n$	5	-1	5
$i \leftarrow i+2$	5	1	5
$j \leftarrow j+3$	5	1	8
$i \leftarrow i+2$	5	3	8
$j \leftarrow j+3$	5	3	11
$i \leftarrow i+2$	5	5	11
$j \leftarrow j+3$	5	5	14
$j \leftarrow j-5$	5	5	9
$n \leftarrow n+1$	6	5	9

Instruction	n	i	j
$i \leftarrow i+2$	6	7	9
$j \leftarrow j+3$	6	7	12
$j \leftarrow j-5$	6	7	7
$n \leftarrow n+1$	7	7	7
$i \leftarrow i+2$	7	9	7
$j \leftarrow j+3$	7	9	10
$j \leftarrow j-5$	7	9	5
$n \leftarrow n+1$	8	9	5
ecrire(i,j)	8	9	5

Cet algorithme affichera : La valeur de i est : ...9... et celle de j est : ...5...

Exercice 2:

```
. #include <stdio.h> . . . . .  
. int a,b,c,ecart1,ecart2,ecart3,ecartMin; .  
. int main(){ . . . . .  
.     printf("Donner 3 entiers SVP:");  
.     scanf("%d %d %d",&a,&b,&c);  
.     ecart1=a-b;  
.     if(ecart1<0){ecart1=-ecart1;}  
.     ecart2=b-c;  
.     if(ecart2<0){ecart2=-ecart2;}  
.     ecart3=a-c;  
.     if(ecart3<0){ecart3=-ecart3;}  
.     ecartMin=ecart1;  
.     if(ecart2<ecartMin){ecartMin=ecart2;}  
.     if(ecart3<ecartMin){ecartMin=ecart3;}  
.     printf("\n Le plus petit ecart est:%d",ecartMin);  
.     return 0;  
. }
```

declaration 1pt

lecture 1pt

calcul des ecart 3pt

le ecart min 3pts

Affichage 1pt

Dem: (+1pt) pour la propriété de la famille.

Matric.: Cocuzé Jp Nom: Version 2 Prénom: 01/12/22 Gr:
Durée : 30 min **Test de TP**

Algorithmique et Structures de Données Statique1

Exercice (12 pts)

Ecrire un programme C qui lit trois entiers, puis affiche celui qui a la plus petite valeur absolue.
Par exemple, pour les nombres -5, -3, -12 votre programme doit afficher -3.

Réponse :

```
#include <stdio.h>
int a,b,c,abs1,abs2,abs3,absMin,whosAbs;
int main(){
    printf("Donner 3 entiers SVP:");
    scanf("%d %d %d",&a,&b,&c);
    abs1=a;
    if(abs1<0){abs1=-abs1;}
    abs2=b;
    if(abs2<0){abs2=-abs2;}
    abs3=c;
    if(abs3<0){abs3=-abs3;}
    absMin=abs1;
    whosAbs=a;
    if(abs2<absMin){
        absMin=abs2;
        whosAbs=b;
    }
    if(abs3<absMin){
        absMin=abs3;
        whosAbs=c;
    }
    printf("\n Le nombre ayant la plus petite valeur absolue"
           " est:%d",whosAbs);
    return 0;
}
```

Handwritten annotations:

- declaration 1pt* (next to variable declarations)
- lecture 2pts* (next to scanf)
- Traitement val. absolue 3pts* (bracketed next to the three if-statements for absolute values)
- 1/ Calcul min val absolue 3pts* (bracketed next to the if-statements for finding the minimum)
- 2/ A qui elle appartient 2pts* (bracketed next to the assignments of whosAbs)
- affichage 1pt* (next to the printf statement)

Remarque: Ne pas pénaliser l'oubli des point-virgules.
2/ (+1) pour la propreté de la feuille.

Matric.: Cornigé type Nom: Version 1 Prénom: 01/12/22 Gr: ...
Durée : 30 min

Test de TD

Algorithmique et Structures de Données Statique1

Exercice 1 (6 pts)

Dérouler l'algorithme suivant en donnant la valeur de chaque variable après exécution de chaque instruction et préciser ce qu'il affiche, pour la valeur $n=6$:

Algorithme test1;

var i,j,n: entier;

Debut

ecrire('Donner une valeur :');

lire(n);

$i \leftarrow -1$; $j \leftarrow n$;

Tant que $j \geq n$ faire

si $i \leq n$ alors

$i \leftarrow i+2$;

$j \leftarrow j+3$;

fsi ;

si $i \geq n$ alors

$j \leftarrow j-5$;

$n \leftarrow n-1$;

fsi;

ftq;

ecrire('La valeur de i est :', i, ' et celle de j est :', j);

Fin.

Exercice 2 (9 pts)

Soient a,b et c trois entiers donnés. Écrire un algorithme ou un programme C qui affiche le plus grand écart entre 2 nombres pouvant être formé à partir de ces trois nombres.

L'écart entre deux nombre est égal à la valeur absolue de leur différence. Par exemple, l'écart entre 2 et 7 est égal à 5. L'écart entre 8 et -6 est égal à 14.

Exemple :

Le plus grand écart pouvant être formé à partir de 2, 7 et -3 est 10.

Réponse:

Exercice 1: Compléter ce tableau, barrer les lignes en plus s'il y en a.

Instruction	n	i	j
lire(n)	6	?	?
$i \leftarrow -1$	6	-1	nd
$j \leftarrow n$	6	-1	6
$i \leftarrow i+2$	6	1	6
$j \leftarrow j+3$	6	1	9
$i \leftarrow i+2$	6	3	9
$j \leftarrow j+3$	6	3	12
$i \leftarrow i+2$	6	5	12
$j \leftarrow j+3$	6	5	15
$i \leftarrow i+2$	6	7	15
$j \leftarrow j+3$	6	7	18

Instruction	n	i	j
$j \leftarrow j-5$	6	7	13
$n \leftarrow n-1$	5	7	13
$j \leftarrow j-5$	5	7	8
$n \leftarrow n-1$	4	7	8
$j \leftarrow j-5$	4	7	3
$n \leftarrow n-1$	3	7	3
$j \leftarrow j-5$	3	7	-2
$n \leftarrow n-1$	2	7	-2
ecrire(.)	2	7	-2

Cet algorithme affichera : La valeur de i est : 7 et celle de j est : -2

6pts puis diag
erreur -0,25

Exercice 2: `#include <stdio.h>`
`int a, b, c, ecart1, ecart2, ecart3, ecartMax;`
`int main(){`

`printf("Donner 3 entiers SVP:");`

`scanf("%d %d %d", &a, &b, &c);`

`ecart1=a-b;`

`if(ecart1<0){ecart1=-ecart1;}`

`ecart2=b-c;`

`if(ecart2<0){ecart2=-ecart2;}`

`ecart3=a-c;`

`if(ecart3<0){ecart3=-ecart3;}`

`ecartMax=ecart1;`

`if(ecart2>ecartMax){ecartMax=ecart2;}`

`if(ecart3>ecartMax){ecartMax=ecart3;}`

`printf("\n.Le plus grand ecart est: %d", ecartMax);`

`return 0;`

`}`

même détaillé

Voici version 2

Remar. On peut aussi résoudre ce problème en calculant

$\max(a, b, c) - \min(a, b, c)$, bien sûr sans

utiliser des fonctions autres que celles

de la bib standard.

Matric.: *Corrigé type* Nom: *version 1* Prénom: *01/12/22* Gr:
Durée : 30 min

Test de TP

Algorithmique et Structures de Données Statique1

Exercice (12 pts)

Ecrire un programme C qui lit trois entiers, puis affiche celui qui a la plus grande valeur absolue. Par exemple, pour les nombres -5, 3, -12 votre programme doit afficher -12.

Réponse :

```
..... #include <stdio.h> .....  
..... int a,b,c,abs1,abs2,abs3,absMax,whosAbs;  
..... int main(){  
.....     printf("Donner 3 entiers SVP:");  
.....     scanf("%d %d %d",&a,&b,&c);  
.....     abs1=a;  
.....     if(abs1<0){abs1=-abs1;}  
.....     abs2=b;  
.....     if(abs2<0){abs2=-abs2;}  
.....     abs3=c;  
.....     if(abs3<0){abs3=-abs3;}  
.....     absMax=abs1;  
.....     whosAbs=a;  
.....     if(abs2>absMax){  
.....         absMax=abs2;  
.....         whosAbs=b;  
.....     }  
.....     if(abs3>absMax){  
.....         absMax=abs3;  
.....         whosAbs=c;  
.....     }  
.....     printf("\n Le nombre ayant la plus grande valeur absolue"  
.....         " est:%d",whosAbs);  
.....     return 0;  
..... }
```

barème détaillé

voir version 2