

Travaux Dirigés 1

Notions de base et Structures de contrôle alternatives

Exercice 1 :

Exemple 1 : va afficher le résultat de multiplication la valeur de 'A' par 'B'

Exemple 2 : va afficher un message d'erreur puisque 'A' et 'B' n'ont aucune valeur initiale

Exercice 2 :

Identificateur Valide	Identificateur Invalide
Mat UneValeurEntiere M1546 B3F2_G2Oper cinq	Val-Abs Première Debut debut

Exercice 3 :

Le Type est réel :

Exercice 4 :

Algorithme	Langage C
Algorithme exercice 4; var nombre, carre: entier ; Début Ecrire('Donner nombre') ; Lire(nombre) ; carre ← nombre* nombre; Ecrire('le carre de ',nombre,' est : ',carre) ; Fin.	<pre> 1 #include<stdio.h> 2 3 int main(){ 4 int nombre, carre; 5 printf("Donner un nombre entier :\n"); 6 scanf("%d", &nombre); 7 carre = nombre * nombre; 8 printf("Le carre de %d est: %d\n", nombre, carre); 9 return 0; 10 } 11 </pre>

Exercice 5 :

Algorithme

Algorithme exercice 5;
var n1,n2,somme,produit,difference: entier ;
Début
Ecrire('Donner deux nombre') ;
Lire(n1, n2) ;
somme \leftarrow n1+ n2;
produit \leftarrow n1* n2;
difference \leftarrow n1- n2;
Ecrire('la somme de ',n1,' et ',n2, ' est :',somme) ;
Ecrire('la difference entre ',n1,' et ',n2, ' est :', difference) ;
Ecrire('la multiplication de ',n1,' par ',n2, ' est :', produit) ;
Fin.

Langage C

```

1  #include<stdio.h>
2
3  int main() {
4      int n1,n2,somme,produit,difference;
5      printf("Donner n1 nombre entier :\n");
6      scanf("%d", &n1);
7      printf("\nDonner n2 nombre entier :\n");
8      scanf("%d", &n2);
9      produit = n1 * n2;
10     somme = n1 + n2;
11     difference = n1 - n2;
12     printf("la somme de %d et %d est: %d\n",n1,n2,somme);
13     printf("la difference entre %d et %d est: %d\n",n1,n2,difference);
14     printf("la multiplication de %d par %d est: %d\n",n1,n2,produit);
15     return 0;
16 }
```

Exercice 7 :

Algorithme

Algorithme exercice 7;
var a, b, intermediaire: entier ;
Début
Lire(a, b) ;
intermediaire \leftarrow a;
a \leftarrow b;
b \leftarrow intermediaire;
Fin.

Langage C

```
1  #include<stdio.h>
2
3  int main(){
4      int a,b,intermediaire;
5      printf("Donner a nombre entier :\n");
6      scanf("%d", &a);
7      printf("\nDonner b nombre entier :\n");
8      scanf("%d", &b);
9      printf("les valeurs de a et b avant la permutation sont: %d et %d\n",a,b);
10     intermediaire = a;
11     a = b;
12     b = intermediaire;
13     printf("les valeurs de a et b apres la permutation sont: %d et %d\n",a,b);
14     return 0;
15 }
```

Exercice 8 :

Algorithme

Algorithme exercice 8;

var a, b, c, intermediaire: entier ;

Début

Lire(a, b, c) ;

intermediaire ← a;

a ← c;

c ← b;

b ← intermediaire;

Fin.

Langage C

```
1  #include<stdio.h>
2
3  int main(){
4      int a,b,c,intermediaire;
5      printf("Donner a nombre entier :\n");
6      scanf("%d", &a);
7      printf("\nDonner b nombre entier :\n");
8      scanf("%d", &b);
9      printf("\nDonner c nombre entier :\n");
10     scanf("%d", &c);
11     printf("les valeurs de a, b et c avant la permutation sont: %d, %d et %d\n",a,b,c);
12     intermediaire = a;
13     a = c;
14     c = b;
15     b = intermediaire;
16     printf("les valeurs de a, b et c apres la permutation sont: %d, %d et %d\n",a,b,c);
17     return 0;
18 }
```

Algorithme

Algorithme exercice 8;

var a, b, c, intermediaire: entier ;

Début

Lire(a, b, c) ;

intermediaire \leftarrow a;a \leftarrow b;b \leftarrow intermediaire;intermediaire \leftarrow a;a \leftarrow c ;c \leftarrow intermediaire ;

Fin.

Langage C

```
1  #include<stdio.h>
2
3  int main(){
4      int a,b,c,intermediaire;
5      printf("Donner a nombre entier :\n");
6      scanf("%d", &a);
7      printf("\nDonner b nombre entier :\n");
8      scanf("%d", &b);
9      printf("\nDonner c nombre entier :\n");
10     scanf("%d", &c);
11     printf("les valeurs de a, b et c avant la permutation sont: %d, %d et %d\n",a,b,c);
12     intermediaire = a;
13     a = b;
14     b = intermediaire;
15     intermediaire = a;
16     a = c;
17     c = intermediaire;
18     printf("les valeurs de a, b et c apres la permutation sont: %d, %d et %d\n",a,b,c);
19     return 0;
20 }
```

Algorithme

Algorithme exercice 9;

var x, resultat : entier ;

Début

Lire(x) ;

resultat $\leftarrow (7 * x * x * x) + (5 * x * x) + 15$;

Ecrire('La valeur calculer est : ', resultat) ;

Fin.

Langage C

```
1  #include<stdio.h>
2
3  int main() {
4      int x,resultat;
5      printf("Donner x nombre entier :\n");
6      scanf("%d", &x);
7      resultat = (7*x*x*x)+(5*x*x)+15;
8      printf("La valeur calculer est: %d\n",resultat);
9      return 0;
10 }
```

Exercice 10 :

Algorithme

Algorithme exercice 10;

var x, val : réel ;

Début

Lire(x) ;

val \leftarrow x ;

si (x < 0) alors

 val \leftarrow (-1) * x;

finsi;

Ecrire('La valeur absolue de ', x, ' est : ', val) ;

Fin.

Langage C

```

1  #include<stdio.h>
2
3  int main(){
4      float x, val;
5      printf("Donner x nombre reel :\n");
6      scanf("%f", &x);
7      val = x;
8      if(x < 0){
9          val = (-1) * x;
10     }
11     printf("La valeur absolue de %f est: %f", x, val);
12     return 0;
13 }
```

Exercice 11 :

Algorithme

Algorithme exercice 11;

var x, y, min : réel ;

Début

Lire(x,y) ;

si (x < y) alors

 min \leftarrow x;

sinon

 min \leftarrow y ;

finsi;

Ecrire('Le minimum est ', min) ;

Fin.

Langage C

```

1  #include<stdio.h>
2
3  int main(){
4      float x,y,min;
5      printf("Donner x nombre reel :\n");
6      scanf("%f", &x);
7      printf("Donner y nombre reel :\n");
8      scanf("%f", &y);
9      if(x < y){
10         min = x;
11     }else{
12         min = y;
13     }
14     printf("Le minimum est: %f", min);
15     return 0;
16 }
```


Exercice 12 :

Algorithme	Langage C
Algorithme exercice 12; var x, y, z, min : réel ; Début Lire(x, y, z) ; si (x < y) alors min ← x; sinon min ← y; finsi si (z < min) alors min ← z; finsi Fin.	<pre> 1 #include<stdio.h> 2 int main() { 3 float x,y,z,min; 4 printf("Donner x nombre reel :\n"); 5 scanf("%f", &x); 6 printf("Donner y nombre reel :\n"); 7 scanf("%f", &y); 8 printf("Donner z nombre reel :\n"); 9 scanf("%f", &z); 10 if(x < y){ 11 min = x; 12 }else{ 13 min = y; 14 } 15 if(z < min){ 16 min = z; 17 } 18 printf("Le minimum est: %f",min); 19 return 0; 20 } </pre>
Algorithme exercice 12; var x, y, z, min : réel ; Début Lire(x, y, z) ; si (x < y) et (x < z) alors min ← x; finsi si (y < x) et (y < z) alors min ← y; finsi si (z < y) et (z < x) alors min ← z; finsi Fin.	<pre> 1 #include<stdio.h> 2 int main(){ 3 float x,y,z,min; 4 printf("Donner x nombre reel :\n"); 5 scanf("%f", &x); 6 printf("Donner y nombre reel :\n"); 7 scanf("%f", &y); 8 printf("Donner z nombre reel :\n"); 9 scanf("%f", &z); 10 if((x < y)&&(x < z)){ 11 min = x; 12 } 13 if((y < x)&&(y < z)){ 14 min = y; 15 } 16 if((z < x)&&(z < y)){ 17 min = z; 18 } 19 printf("Le minimum est: %f",min); 20 return 0; 21 } </pre>

Autre solutions

Algorithme exercice 12;

var x, y, z, min : réel ;

Début

Lire(x, y, z) ;

si (x < y) **alors****si** (x < z) **alors**

min ← x;

sinon

min ← z;

finsi**sinon****si** (y < z) **alors**

min ← y;

sinon

min ← z;

finsi**finsi**

Fin.

Algorithme exercice 12;

var x, y, z, min : réel ;

Début

Lire(x, y, z) ;

si (x < y) **et** (x < z) **alors**

min ← x;

sinon**si** (y < z) **alors**

min ← y;

sinon

min ← z;

finsi**finsi**

Fin.