

dev 1.1 BOUCLES

1)

AVEC WHILE

```
#include <stdio.h>
#include <stdlib.h>

int main(void) {
    int x,y;
    printf("insérez un entier naturel\n");
    scanf("%d", &x);
    printf("insérez un deuxième entier naturel\n");
    scanf("%d", &y);
    if (x<y){
        printf("dans l'ordre croissant tous les entiers compris entre ces deux valeurs sont :\n");
        while(x<y){
            printf("%d\n",x);
            x+=1;
        }
    }else if(x==y){
        printf("Il n'y a aucun entier qui soit compris entre ces deux valeurs car ils sont égaux\n");
    }else{
        printf("dans l'ordre croissant tous les entiers compris entre ces deux valeurs sont :\n");
        while(x>y){
            printf("%d\n",y);
            y+=1;
        }
    }
    return EXIT_SUCCESS;
}
```

AVEC DO WHILE

```
#include <stdio.h>
#include <stdlib.h>

int main(void) {
    int x,y;
    printf("insérez un entier naturel\n");
    scanf("%d", &x);
    printf("insérez un deuxième entier naturel\n");
    scanf("%d", &y);
    if (x<y){
        printf("dans l'ordre croissant tous les entiers compris entre ces deux valeurs sont :\n");
        do{
            x+=1;
            printf("%d\n",x);
        }while(x<y);
    }else if(x==y){
        printf("Il n'y a aucun entier qui soit compris entre ces deux valeurs car ils sont égaux\n");
    }else{
        printf("dans l'ordre croissant tous les entiers compris entre ces deux valeurs sont :\n");
        do{
            y+=1;
            printf("%d\n",y);
        }while(x>y);
    }
    return EXIT_SUCCESS;
}
```

boucles suites : 2.

```
#include <stdio.h>
```

```

#include <stdlib.h>

int main(void) {
    int x, diviseur, premier = 1;
    printf("Insérez un entier naturel : ");
    scanf("%d", &x);

    if (x < 2) {
        printf("Ce n'est pas un nombre premier\n");
        return 0;
    }

    for (diviseur = 2; diviseur <= x / 2; diviseur++) {
        if (x % diviseur == 0) {
            premier = 0;
            break;
        }
    }

    if (premier == 1) {
        printf("%d est un nombre premier\n", x);
    } else {
        printf("%d n'est pas un nombre premier\n", x);
    }

    return 0;
}

```

AVEC LE FOR

```

#include <stdio.h>
#include <stdlib.h>

int main(void) {
    int x,y;
    printf("insérez un entier naturel\n");
    scanf("%d", &x);
    printf("insérez un deuxième entier naturel\n");
    scanf("%d", &y);
    if (x<y){
        printf("dans l'ordre croissant tous les entiers compris entre ces deux valeurs sont :\n");
        for(;x<y;x+=1){
            printf("%d\n",x);
        }
    }else if(x==y){
        printf("Il n'y a aucun entier qui soit compris entre ces deux valeurs car ils sont égaux\n");
    }else{
        printf("dans l'ordre croissant tous les entiers compris entre ces deux valeurs sont :\n");
        for(;x>y;y+=1){
            printf("%d\n",y);
        }
    }
    return EXIT_SUCCESS;
}

```

2)

```

#include <stdio.h>
#include <stdlib.h>

```

```

int main(void) {
    int x,y;
    int z=1;
    printf("insérez un entier:\n");
    scanf("%d", &x);
    if (x==0){
        printf("Le chiffre 0 n'a pour résultat dans la table de multiplication 0.\n");
        x*10=y;
    }else{
        while(x!= x*10){
            printf(" x ",z);
            printf("%d\n",x);
            z+=1;
            x*y=x;
        }
    }
    return EXIT_SUCCESS;
}

```

BOUCLES (SUITES)

1)

```
#include <stdio.h>
```

Devinette. Écrivez un programme qui donne cinq tentatives à l'utilisateur pour deviner un nombre entre 0 et 100. À chaque tentative infructueuse, le programme offrira un indice en affichant + ou -.

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <time.h>
```

```

int main(void) {
    srand(time(NULL));
    int random=rand()%100+1,vie=5,rep;

```

```

    printf("Vous avez %d tentatives pour deviner un nombre entre 0 et 100.\n Insérez donc votre choix\n"
, vie);

```

```
    scanf("%d", &rep);
```

```
    while(vie>=0){
```

```
        if(rep<random){
```

```
            printf("c'est + réessayez il vous reste %d vie\n",vie);
```

```
            vie--;
```

```
            scanf("%d", &rep);
```

```
        }else if(rep>random){
```

```
            printf("c'est - réessayez il vous reste %d vie\n",vie);
```

```
            vie--;
```

```
            scanf("%d", &rep);
```

```
        }else if(rep==random){
```

```

printf("BRAVO vous avez trouvez le nombre mystère qui on le rappelle était %d\n", random);
vie = 0;
break;
}else{
printf("nul");
}
}
}
}

```

NOMBRE PREMIER

```

#include <stdio.h>
#include <stdlib.h>

int main(void) {
    int x, diviseur, premier = 1;
    printf("Insérez un entier naturel : ");
    scanf("%d", &x);

    if (x < 2) {
        printf("Ce n'est pas un nombre premier\n");
        return 0;
    }

    for (diviseur = 2; diviseur <= x / 2; diviseur++) {
        if (x % diviseur == 0) {
            premier = 0;
            break;
        }
    }

    if (premier == 1) {
        printf("%d est un nombre premier\n", x);
    } else {
        printf("%d n'est pas un nombre premier\n", x);
    }

    return 0;
}

```

```

#include <stdio.h>

int main(void) {
    int x, i, u;

    printf("Insérez la longueur souhaitée de la table de multiplication : ");
    scanf("%d", &x);

    printf(" X |");
    for (i = 0; i <= x; i++) {
        printf("%4d", i);
    }
    printf("\n-----");

    for (i = 0; i <= x; i++) {
        printf("----");
    }
    printf("\n");

    for (u = 0; u <= x; u++) {
        printf("%4d |", u);
        for (i = 0; i <= x; i++) {
            printf("%4d", u * i);
        }
        printf("\n");
    }

    return 0;
}

```

Bibliothèque math
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

```

int main(void) {
    int t;
    t=log(0.5);
    int y;
    y=fabs(t);
    int x;
    x=sqrt(y);
    printf("%d\n",x);

}

```

sapin

```
#include <stdio.h>
```

```
int main(void) {  
    int n, i, j, k;
```

```
    printf("Quelle hauteur ? : ");  
    scanf("%d", &n);
```

```
    for (i = 1; i <= n; i++) {
```

```
        for (j = 1; j <= n - i; j++) {  
            printf(" ");  
        }
```

```
        for (k = 1; k <= (2 * i - 1); k++) {  
            printf("*");  
        }
```

```
        printf("\n");  
    }
```

```
    return 0;  
}
```

<https://www.programiz.com/online-compiler/3655nuzGRofxz>

```
#include <stdio.h>
```

```
#include <stdio.h>
```

```
int main() {
```

```
    char quitter = 'q', triangle = 't', carre = 'c', x;  
    int n, i, k, e;
```

```
    while(quitter!='q'){  
        printf("_____ \n t) Triangle\n c) Carré\n q) Quitter\nVotre choix ?\n");  
        scanf(" %c", &x);
```

```
        if(x=='q'){  
            printf("Au revoir...\n");  
            quitter = 'p';  
        }else if(x=='t'){  
            printf("Hauteur ? \n");
```

```

scanf("%d",&n);

for(i=1;i<=n;i++){
for (k = 1; k <= (2 * i - 1); k++) {
    printf("*");
}
printf("\n");

}
}else if(x=='c'){
printf("Hauteur ?\n");
scanf("%d",&n);

for(i=0;i<n;i++){
    for(e=0;e<n;e++){
        if (i == 0 || i == n - 1 || e == 0 || e == n - 1) {
            printf("*");
        } else {
            printf(" ");
        }
    }
}

printf("\n");
}
}

return 0;
}
}

```

Remplissage. Écrivez un programme qui déclare un tableau de 10 cases de type int, puis le remplit avec des valeurs choisies aléatoirement entre -50 et 50, et enfin affiche le contenu du tableau.

```

#include <stdio.h>
#include <stdlib.h>
#include<time.h>

```

```

int main(void) {
    srand(time(NULL));

```

```

    int i=0,l,tab[10];

```

```

        for(;i<10;i++){
            l=rand()%101-50;
            tab[i]=l+1;
        }

```

```

for(i=0; i < 10; i++){
    printf("+-----");
}
printf("+\n");

```

```

for(i=0; i<10; i++){

```

```

        printf("| ");
        printf("%d ",tab[i]);
    }
    printf(" |\n");

    for(i=0; i < 10; i++){
        printf("+-----");
    }
    printf("+\n");

    return 0; }
#include <stdio.h>
#include <stdlib.h>

int main(void) {

    int i,t1[2][5],c,l;

        for(i=0;i<5;i++){
            t1[0][i]=i;
            t1[1][i]=i;
        }

        for(i=0;i<5;i++){
            c = t1[0][i];
            printf("| %d ", c);

        }

        printf("\n");

    return 0;
}

```

Tableaux multidimensionnels ex 1.

```

#include <stdio.h>
#include <stdlib.h>

int main(void) {

    int t1[2][5];
    int i,j;

    for(i=0;i<2;i++){
        for(j=0;j<5;j++){
            t1[i][j]=j+1;
            t1[i][j]=j+1;
        }
    }
}

```



```

}
for(i=0;i<2;i++){
    for(j=0;j<5;j++){
        printf("| %d ",t1[i][j]);
    }
    printf("\n");
}

printf("\n");

}

```

```

#include <stdio.h>
#include <stdlib.h>

```

```

int main(void) {

    int t1[3][5];
    int i,j;

    for(i=0;i<3;i++){
        for(j=0;j<5;j++){
            t1[i][j]=j+1;
        }
        t1[i][j]=j*5+1;
    }
    for(i=0;i<3;i++){
        for(j=0;j<5;j++){
            printf("| %d ",t1[i][j]);
        }
        printf("\n");
    }

    printf("\n");

}

```

```

#include <stdio.h>
#include <stdlib.h>

```

```

int main(void) {

    int t1[2][5];
    int i,j;

    for(i=0;i<2;i++){
        for(j=0;j<5;j++){
            t1[i][j]=j+1;
        }
    }
}

```

```

        t1[i][j]=j+1;
    }
}
for(i=0;i<1;i++){
    for(j=0;j<5;j++){
        printf("| %d ",t1[i][j]);
        printf("| %d\n",t1[i][j]);
    }
}

printf("\n");

}

```

ICI ON INVERSE LES COLONNES ET LES LIGNES

```

#include <stdio.h>
#include <stdlib.h>

int main(void) {

    int t1[3][5];
    int i,j;

    for(i=0;i<3;i++){
        for(j=0;j<5;j++){
            t1[0][j]=j+1;
        }
        for(i=1;t1[3][5]!=15){
            t1[i][j]
        }
    }
    for(i=0;i<3;i++){
        for(j=0;j<5;j++){
            printf("| %d ",t1[i][j]);
        }
        printf("\n");
    }

    printf("\n");

}

```

Opérateurs * + -

```

Ex : int n = 23;
    int* p = &n;
    printf(“”%d\n”, *p);
    *p = 7;

```

L'opérateur * permet de manipuler la donnée à l'adresse fournie.

Ex : *(p+1)=8;

Ajouter 1 à une adresse revient à avancer d'une valeur (pas d'un octet).

On peut faire une soustraction entre deux adresses donc faire leur différence soit leur écart qui calcule en nombre de place/ valeur entre elle

```
#include <stdio.h>
#include <stdlib.h>
```

```
int main(void) {
    int* a;
    float* b;
    double* c;
    long double* d;
    char* e;
    short int* f;
    unsigned long int* g;

    printf("a : %p b : %p c : %p d : %p e : %p f : %p %g : %p \n", &a,&b,&c,&d,&e,&f,&g);
}
```

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define LIMIT 26
```

```
int main(int argc, char** argv) {
    char tab,choix;
```

```
    printf("inserez un mot de passe qui ne dépasse pas les 26 caractères. \n");
    tab=getchar();
```

```
    while(choix!='c'){
        printf("Que voulez vous faire? \n |a| essayer le mdp \n |b| Modifier le mdp \n |c| arreter \n");
        choix=getchar();
    }
```

```
    if(choix=='a'){
        if(tab>LIMIT){
            printf("Votre mot de passe dépasse les 26 caractères. \n Veuillez en inserez un autre. \n");
            tab=getchar();
            if(tab<=LIMIT){
```

```

        printf("Le mot de passe rentre dans les critères");
    }
}
}else if(choix=='b'){
    printf("inserez un mot de passe qui ne dépasse pas les 26 caractères. \n");
    tab=getchar();

}else if(choix=='c'){
    printf("Au revoir...");
    break;
}
}
}

```

```

#include <stdlib.h>

```

```

int demander_hauteur(void){
    int x,b;
    do{
        printf("hauteur?\n");
        scanf("%d",&x);
        b = x<0;
        if(b){
            printf("valeur incorrect recommencer\n");
        }
    }while(b);
    return x;
}

```

```

void afficher_carre(int h){
    int i,k;
    for(i=1;i<=h;i++){
        for (k = 1; k <= (2 * i - 1); k++) {
            printf("*");
        }
        printf("\n");

    }
    return;
}

```

```

void afficher_triangle(int h){
    int i,e;
    for(i=0;i<h;i++){
        for(e=0;e<h;e++){
            if (i == 0 || i == h - 1 || e == 0 || e == h - 1) {
                printf("*");
            } else {
                printf(" ");
            }
        }
    }
}

```

```

    }
    }

    printf("\n");
    }
    return;
}

```

```

void mod_triangle(void){
    int h=demander_hauteur();
    afficher_triangle(h);
    return;
}

```

```

void mod_carre(void){
    int h=demander_hauteur();
    afficher_carre(h);
    return;
}

```

```

char menu(void){
    char x;
    int n,b;

    do{
        printf("_____ \n t) Triangle\n c) Carre\n q) Quitter\nVotre choix ?\n");
        scanf(" %c", &x);
        scanf("%*");
        b = x!='q'&& x!='c'&& x != 't';
        if(b){
            printf("valeur incorrect recommencer\n");
        }
    }while(b);
    return x;
}

```

```

int main(void){
    char mod=0;
    while(mod !='q'){
        mod = menu();
        if (mod='t')
        {
            mod_triangle();
        }else if(mod='c'){
            mod_carre();
        }
    }
}

```

```

}
printf("Au revoir...\n");
return EXIT_SUCCESS;
}

```

Ex 3 cm 1.B

```

#include <stdlib.h>
#include <stdio.h>
#define PRXUNIT 0.6
#define PRXPACK 3.39

int main(void) {
    float somme;
    printf("Quelle somme êtes vous prêt à payer? \n");
    scanf("%f", &somme);

    float somxunit = somme*PRXUNIT,somxpack = somme*PRXPACK;

    for(i=0;i!=

    return EXIT_SUCCESS;
}

```

EX 4 CONTROLE MACHINE 1.B (entrainement)

```

#include <stdio.h>

int main() {
    char x,i,cap;
    printf("Entrez une minuscule : ");
    scanf("%c",&x);
    cap=x;

    for(;x<='z';x++){
        printf("%c",x);

    }

    for(i='a';i<cap;i++){
        printf("%c",i);

    }
    return 0;
}

```

```

#include <stdlib.h>
#include <stdio.h>
#define PRXUNIT 0.6
#define PRXPACK 3.39

```

```
int main(void) {
    float somme;
    printf("Quelle somme êtes vous prêt à payer? \n");
    scanf("%f", &somme);

    if(somme>PRIXPACK){
        float somxpack = somme/PRIXPACK;
        somme -= PRIXPACK*
        printf("Vous pouvez acheter %0.f-1 canettes",somxpack);

    }else if(somme==PRIXUNIT){
        printf("\vous pouvez avoir 1 canette\n");

    }

    if(somme>PRIXUNIT){
        float somxunit = somme/PRIXUNIT;

    }

    return EXIT_SUCCESS;
}
```