

```

/*
    Name      :
    Goal      :
    Author    :
    Date      :
*/

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
#define NB_COUP_MAX 15

int main()
{
    void Reprendre();
    int SommeT(int *Tab, int nb_coup);
    double moyenneOlymp(int *Tab, int nb_coup);
    double ecartType(int *Tab, int nb_coup);

    //Déclaration des variables
    int nb_coup = 0; //nombre de coups
    int nb_min = 1;
    int nb_mystere; //nombre aléatoire généré par l'ordinateur
    int nb_max; //borne maximale qui sera saisie par l'internaute
    int user_input; //nombre deviné par l'utilisateur
    int c; //contrôleur de type
    int TabT[NB_COUP_MAX+1]; //Tableau contenant les tentatives
    int i, j; //Compteurs courants
    double ecartType; //L'ecart type
    double moyenneOlymp; //La moyenne olympique
    //titre du jeu
    printf("*****\n");
    printf("*****  JEU DE DEVINETTE  *****\n");
    printf("*****\n\n");

    printf("Veuillez entrer la borne maximale du nombre aleatoire: ");
    do
    {
        c=scanf("%d", &nb_max);
        if(!c)
        {
            printf("Erreur de saisie, veuillez resaisir!!\n");
        }
    }
    while(!c);

    //obtention du nombre mystere
    srand(time(NULL)); //Initialisation du generateur du nombre aleatoire
    nb_mystere = nb_min + (rand()%(nb_max-nb_min + 1));

    //l'utilisateur devine le nombre aléatoire
    printf("Vous allez a present deviner le nombre mystere.\n");
    printf("Quel est le nombre mystere?\n");
    do
    {
        do
        {
            printf("Entrer le nombre mystere: ");
            c=scanf("%d", &user_input);
            fflush(stdin);
            if(!c || user_input > nb_max)
            {
                printf("\nVeuillez revoir votre valeur saisie!\n");
            }
        }
    }
}

```

```

    while(!c || user_input > nb_max);
    TabT[i]=user_input;
    if(user_input < nb_mystere)
    {
        printf("Entrer une valeur plus grande!\n");
    }
    else if(user_input > nb_mystere)
    {
        printf("Entrer une valeur plus petite!\n");
    }
    nb_coup++;
    i++;
}
while(user_input!=nb_mystere && nb_coup < NB_COUP_MAX);

if(user_input == nb_mystere)
{
    printf("\nFelicitation, vous avez trouve le nombre mystere en %d coup(s)\n", nb_coup);
}
else
{
    printf("Game over!!!\n");
    printf("Vous avez atteint le nombre de coups maximal\n");
}

if(nb_coup <= 3)
{
    printf("Vous etes un super voyant.\n");
}
else if(nb_coup <= 6)
{
    printf("Vous etes un sage voyant.\n");
}
else if(nb_coup <= 9)
{
    printf("Vous etes un apprenti voyant.\n");
}
else
{
    printf("Vous etes un voyant pusillanime.\n");
}
printf("Voici l'historique des tentatives\n");

for(j=0; j<i; j++)
{
    printf("%d\t", TabT[j]);
}
printf("\n");
ecartType = ecartType(TabT, nb_coup);
printf("L'ecart type est : %.3lf\n", ecartType(TabT, nb_coup));
printf("\nLa moyenne olympique est: %.3lf\n", moyenneOlymp(TabT, nb_coup));

Reprendre();

return 0;
}

/**ProcEDURE pour reprendre l'opération***/
void Reprendre()
{
    char rep;
    do
    {
        printf("\nVoulez vous reprendre un autre traitement?(O/N) : ");
    }

```

```

scanf("%c",&rep);
fflush(stdin);
if(rep!='o' && rep!='O' && rep!='n' && rep!='N')
{
    printf("\a");
    printf("\nMerci de bien vouloir saisir une bonne reponse.");
}
}
while(rep!='o' && rep!='O' && rep!='n' && rep!='N');
if(rep=='o' || rep=='O')
{
    system("cls");
    main();
}
else
{
    printf("\nMerci!");
}
}

```

/\*Fonction somme\*/

```

int SommeT(int *Tab, int nbcoup)
{
    int s=0;
    int i;
    for(i=0; i<nbcoup; i++)
        s+=Tab[i];
    return s;
}

```

```

double moyenneOlymp(int *Tab, int nb_coup)
{
    double moy=0;
    int i,j,tmp,s;
    for(i=0; i<nb_coup-1; i++)
    {
        for(j=i+1; j<nb_coup; j++)
        {
            if(Tab[i]>Tab[j])
            {
                tmp = Tab[i];
                Tab[i] = Tab[j];
                Tab[j] = tmp;
            }
        }
    }

    for(i=1,s=0; i<nb_coup-1; i++)
    {
        s+=Tab[i];
    }

    moy = (double)s/(nb_coup-2);

    return moy;
}

```

```

double ecartType(int *Tab, int nb_coup)
{
    int i,som=0;
    double ecart=0;
    double moyenne=0;

    moyenne = SommeT(Tab,nb_coup)/nb_coup;

    for(i=0,som=0; i<nb_coup; i++)
    {

```

```
        som += pow((Tab[i]-moyenne),2);  
    }  
    ecart = sqrt(som/nb_coup);  
    return ecart;  
}
```