

Algorithme complet:

```
Sous programme initialisation
    entier nombre_de_points <--- 0
    tableau d'entiers tableau_initial <--- 0
fin sous programme initialisation
```

```
Sous programme tirage_tableau_initial
    Pour entier indice allant de 1 à 10
        Faire
            Booléen sortie <--- VRAI

            Entier alea <--- nombre_aleatoire (modulo 500 )
            Pour valeur allant de 1 à ( indice max )
                Si alea = tableau_initial[valeur]
                    faire sortie <--- FAUX
                Fin si
            Fin Pour
        Fin tant que
        Tant que (sortie vaut FAUX )
            tableau_initial[indice] <--- alea
        Fin Pour
    Fin Sous programme tirage_tableau_initial
```

```
Sous programme generer_numero_element_du_tableau
    Pour entier indice2 allant de 1 à 10
        Faire
            Booléen sortie2 <--- VRAI
            Entier alea2 <--- nombre_aleatoire (modulo 10 )
            Pour valeur allant de 1 à 10
                Si alea2 = tableau_initial[valeur]
                    faire sortie2 <--- FAUX
                Fin si
            Fin Pour
        Tant que (sortie2 vaut FAUX )
            tableau_initial[indice2] <--- alea2
        Fin Pour
    Fin Sous programme generer_numero_element_du_tableau
```

```
Sous programme afficher_question
    Si (alea2 + 1 est different de 1)
        suffixe <--- ième
    Sinon
        suffixe <--- ier
    Fin Si
    Ecrire ( donner le (alea2 +1) suffixe élément du tableau )
Fin sous programme afficher_question
```

```
Sous programme lire_reponse
```

Lire (reponse)
Effacer l'écran
Fin sous programme lire_reponse

Sous programme affichage_resultat_question
Si (reponse = nombre_tableau[alea2])
 Ecrire (c'est juste)
 nombre_de_points <--- nombre_de_points + 2
Sinon
 Ecrire (c'est faux)
Fin Si

Fin Sous programme affichage_resultat_question

Sous programme effacer element_tableau
 /* il faut remplacer la valeur lue par un zero */
 element_tableau[alea2] <---- 0
Fin sous programme affichage_resultat_question

Sous programme affichage_resultat_final

Ecrire (Nombre de points = nombre_de_points)

Fin sous programme affichage_resultat_final

Sous programme affichage_tableau_initial
 Pour entier indice2 allant de 1 à 10
 Afficher tableau_initial[indice2]
 Fin pour
Fin sous programme affichage_tableau_initial

Programme principal
 Initialisation
 Tirage tableau initial
 affichage_tableau_initial
 Ecrire (appui entree pour continuer)
 Lire (appui touche utilisateur)
 Pour compteur allant de 1 a 10
 generer_element_du_tableau
 afficher_question
 lire_reponse
 affichage resultat question

effacer element_tableau

Fin pour

affichage resultat final

Fin programme principal

Suggestion:

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

int main(int argc, char** argv)
{
    int tirage;
    time_t valeur_temps;
    (void)time(&valeur_temps);
    srand((long)valeur_temps); /*initialisation du random*/
    for(tirage = 0; tirage < 10; tirage++)
        fprintf(stdout, "%d\n", rand()%488);
    return EXIT_SUCCESS;
}
```