

CheckPoint Algo

Première tâche :

```
program CompterMotsEtVoyelles;

var
  phrase: string;
  nombreMots, nombreVoyelles, i: integer;
  caractere: char;

begin
  // Initialisation des compteurs
  nombreMots := 0;
  nombreVoyelles := 0;

  // Lecture de la phrase
  write('Entrez une phrase se terminant par un point : ');
  readln(phrase);

  // Parcours de la phrase caractère par caractère
  for i := 1 to length(phrase) do
    begin
      caractere := phrase[i];
      // Vérifier si le caractère est alphabétique
      if caractere in ['a'..'z', 'A'..'Z'] then
        begin
          // Compter les voyelles (en minuscule)
          if caractere in ['a', 'e', 'i', 'o', 'u', 'y', 'A', 'E', 'I', 'O', 'U', 'Y'] then
            nombreVoyelles := nombreVoyelles + 1;
        end
      end
    end
  end
```

```

else if caractere = ' ' then
begin
    nombreMots := nombreMots + 1;
end;
end;

// Incrémenter le nombre de mots pour le dernier mot
nombreMots := nombreMots + 1;

// Affichage des résultats
writeln('Nombre de mots dans la phrase : ', nombreMots);
writeln('Nombre de voyelles dans la phrase : ', nombreVoyelles);
end.

```

Deuxième tâche :

```

program SommeElementsDistincts;

type
    Ensemble = array of Integer;

function SommeElementsDistincts(ensemble1, ensemble2: Ensemble): Integer;
var
    somme, element: Integer;
    i: Integer;
    presentDansEnsemble2, presentDansEnsemble1: Boolean;
begin
    // Initialisation de la somme
    somme := 0;

    // Parcourir chaque élément de l'ensemble 1

```

```

for i := 0 to Length(ensemble1) - 1 do
begin
    element := ensemble1[i];
    // Vérifier si l'élément n'est pas présent dans l'ensemble 2
    presentDansEnsemble2 := False;
    for element2 in ensemble2 do
    begin
        if element2 = element then
        begin
            presentDansEnsemble2 := True;
            Break;
        end;
    end;

    // Ajouter l'élément à la somme s'il n'est pas présent dans l'ensemble 2
    if not presentDansEnsemble2 then
        somme := somme + element;
    end;

    // Parcourir chaque élément de l'ensemble 2
    for i := 0 to Length(ensemble2) - 1 do
    begin
        element := ensemble2[i];
        // Vérifier si l'élément n'est pas présent dans l'ensemble 1
        presentDansEnsemble1 := False;
        for element1 in ensemble1 do
        begin
            if element1 = element then
            begin

```

```

    presentDansEnsemble1 := True;

    Break;

end;

end;

// Ajouter l'élément à la somme s'il n'est pas présent dans l'ensemble 1
if not presentDansEnsemble1 then
    somme := somme + element;
end;

// Retourner la somme
SommeElementsDistincts := somme;
end;

// Exemple d'utilisation
var
    ensemble1, ensemble2: Ensemble;
    resultat: Integer;
begin
    // Initialisation des ensembles
    ensemble1 := [3, 1, 7, 9];
    ensemble2 := [2, 4, 1, 9, 3];

    // Calcul de la somme des éléments distincts
    resultat := SommeElementsDistincts(ensemble1, ensemble2);

    // Affichage du résultat
    writeln('Resultat : ', resultat);
end.

```