

## Communication et sobriété numérique

---

### Contexte

Afin de regagner de la souveraineté sur le secteur stratégique des réseaux sociaux pour collégiens, votre société a pour mission de développer une nouvelle application de communication rapide et bienveillante : Bikkok.

Votre équipe est chargée d'écrire un démonstrateur capable d'envoyer les messages. Une autre équipe se charge de la réception et de l'affichage. Les types des messages sont :

- Texte
- Photo ?
- Vocal
- Vidéo

Pour la démonstration prévue dans deux semaines, le message envoyé sera : `Pour ce soir (@Kollok) : pâtes ou pizza ?`. Malgré le caractère confidentiel du message, il est envoyé en clair – il n'est pas chiffré. Ce message comprend 41 caractères, dure 3 secondes en vocal et en vidéo.

### 1 Message texte

```
package Texte is
    — Renvoie le texte tapé par l'utilisateur
    function Lire_champ_texte return String ;

    — Envoie un message.
    — Pos_Destinataire indique à quelle position se trouve le '@' du destinataire dans le message.
    procedure Envoyer_texte(Message : String ; Pos_Destinataire : Integer) ;
end Texte ;
```

- Ne rien envoyer si le message est vide / signaler s'il n'y a pas de destinataire ?
- Coût numérique : afficher le nombre d'octets utilisés
- Évaluer le coût CO2 ? ... ou plutôt par rapport à un mail 'moyen'

### 2 Message vocal

- ★ Écrire un programme qui enregistre un son et l'envoie (utiliser l'acteur Vocal ci-dessous).
- Calculer la taille des données envoyées (en octets) pour le message de test, sachant que le son est échantillonné à 44000 Hz et un échantillon occupe 2 octets.
- coût CO2 et comparaison par rapport au message texte ????

La musique ou le son se compresse en général très bien. Pour ne pas gaspiller les ressources, cherchons à transmettre moins d'octets.

- ★ En préambule, écrire le corps de la procédure Copier ci-après. Cette procédure doit copier une partie du tableau Source dans le tableau Dest. La partie copiée commence à la position SPos et sa longueur est Len. Elle est copiée à partir de la position DPos du tableau Dest.

```
procedure Copier(Source : T_Tab ; SPos, Dpos, Len : Integer ; Dest : in out T_Tab)
```

☆☆ En utilisant l'acteur Compression, améliorez votre programme pour qu'il transmette moins d'octets.

```
package Vocal is

  — Un son non compressé est une suite d'échantillons (un tableau d'entiers)
  type T_Tab is array(Integer range <>) of Integer ;

  — Cette fonction enregistre un message vocal et le renvoie sous forme de tableau T_Son
  — Les cases du tableau sont numérotées à partir de 0.
  — Par chance, la taille du tableau est toujours un multiple de Taille_segment_max
  — (voir l'acteur Compression).
  function Enregistrer_Vocal return T_Tab ;

  — Envoie un son au destinataire prévu.
  — Le son peut être compressé ou non compressé.
  — En version compressée, le tableau Son doit contenir tous les segments compressés à la suite
  — (voir l'acteur Compression).
  procedure Envoyer_Son(Son : T_Tab) ;

end Vocal ;
```

---

```
with Vocal ; use Vocal ;

— Ce package permet de compresser des morceaux audio dans le format OGG.
package Compression is

  — La fonction de compression prend un tableau d'entrée appelé segment
  — et renvoie un segment compressé 15 fois plus petit.

  Taux_Compression : constant Integer := 15 ;

  — Le segment d'entrée a une taille de 4096 entiers.
  Taille_Segment_Entree : constant Integer := 4096 ;

  Taille_Segment_Sortie : constant Integer := Taille_Segment_Entree / Taux_Compression ;

  — Comprime les données du segment d'entrée.
  — Le segment d'entrée doit avoir une taille égale à Taille_Segment_max ;
  — Renvoie un nouveau segment compressé, numéroté à partir de 0.
  — La taille du segment compressé est Taille_Segment_Sortie
  function Compresser( Entree : T_Tab ) return T_Tab ;

end Compression ;
```

### 3 Message vidéo

Format 1920x1080

highlight box example