**BTS SERVICES INFORMATIQUES AUX ORGANISATIONS**

**ÉPREUVE PONCTUELLE E22**

**ALGORITHMIQUE APPLIQUÉE**

**SESSION 2019**

## L’usage de tout modèle de calculatrice, avec ou sans mode examen, est autorisé.

Il est **formellement interdit** d’utiliser toute connexion à un réseau interne ou externe au centre d’examen, quel qu’en soit le procédé.

*L’épreuve est constituée d’une préparation d’une heure suivie d’une interrogation orale de vingt minutes.*

*La préparation comporte deux parties :*

*- une première partie, sur table, d’une durée de trente minutes au cours de laquelle le candidat répond au problème posé sur sa copie qu’il conservera jusqu’à la fin de l’épreuve ;*

*- une seconde partie, sur ordinateur, d’une durée de trente minutes, au cours de laquelle le candidat enregistre sa production sur la clé USB fournie et uniquement sur celle-ci.*

*À l’issue des vingt minutes d’interrogation, le candidat restituera à l’examinateur :*

*- le sujet ;*

*- sa copie sur laquelle figureront son nom et son prénom et ses brouillons ;*

*- la clé USB.*

**OCTET CACHÉ**

On considère une image en nuances de gris composée de huit pixels alignés horizontalement.

À chaque pixel est associé un entier compris entre 0 et 255 qui correspond à la couleur du pixel :

* 0 pour un pixel noir,
* 255 pour un pixel blanc,
* les entiers de 1 à 254 pour les différentes nuances de gris, des plus foncées (entiers proches de 0) aux plus claires (entiers proches de 255).

Voici un exemple d’une telle image, modélisée par la variable :

***image*** = [0, 36, 72, 109, 145, 182, 218, 255].

Sous chaque pixel est indiqué l’entier correspondant.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| 0 | 36 | 72 | 109 | 145 | 182 | 218 | 255 |

Notre objectif est de dissimuler un octet, c’est-à-dire une succession de huit bits, à l’intérieur de l’image.

**PARTIE A – Algorithme sur papier**

On vous fournit l’algorithme « A » suivant :

|  |  |
| --- | --- |
| 0  1  2  3  4  5 | ***image*** ← [0, 36, 72, 109, 145, 182, 218, 255]  Pour k allant de 0 à 7 faire  Si ***image***[k] est impair alors  ***image***[k] ← ***image***[k] – 1  Fin Si  Fin Pour |

|  |
| --- |
| **1.** Listez les variables utilisées et précisez leur type.  **2.** Quel traitement réalise cet algorithme « A » ?  **3.** Que contient la variable ***image*** à la fin de l’algorithme ? |

Pour dissimuler un octet (modélisé par une variable ***octet*** de type tableau dont les éléments sont 0 ou 1) dans l’image, on commence par appliquer l’algorithme « A ». Ensuite, on ajoute à chaque terme de ***image*** le terme du tableau ***octet*** de même rang.

|  |
| --- |
| **4.** Écrire un algorithme « B » qui, à partir des variables ***image*** et ***octet***, renvoie la variable ***image*** modifiée dans laquelle ***octet*** a été dissimulé.  Par exemple, pour ***image*** = [0, 36, 72, 109, 145, 182, 218, 255] et  ***octet*** = [0, 1, 1, 0, 1, 0, 1, 1]  On doit obtenir finalement ***image*** = [0, 37, 73, 108, 145, 182, 219, 255]. |

Si on compare l’image initiale et l’image modifiée contenant l’octet caché, on ne distingue aucune différence à l’œil nu. C’est pour cela qu’on dit avoir caché l’octet à l’intérieur de l’image.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| image initiale |  |  |  |  |  |  |  |  |
|  | 0 | 36 | 72 | 109 | 145 | 182 | 218 | 255 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| image modifiée |  |  |  |  |  |  |  |  |
|  | 0 | 37 | 73 | 108 | 145 | 182 | 219 | 255 |

Pour retrouver l’octet caché à l’intérieur d’une image, il suffit de parcourir les pixels de l’image, et de remplacer chaque nombre entier pair par 0 et chaque nombre entier impair par 1.

|  |
| --- |
| **5.** Écrire un algorithme « C » qui, à partir d’une variable ***image***, renvoie la variable ***octet*** correspondant à l’octet dissimulé dans l’image.  Par exemple, pour ***image*** = [0, 37, 73, 108, 145, 182, 219, 255], on doit obtenir  ***octet*** = [0,1,1,0,1,0,1,1]. |

**PARTIE B – Implémentation sur machine**

Implémenter les algorithmes précédents dans un même script. Vous enregistrerez le fichier sous le nom NOM\_Prenom sur la clé USB fournie.