|  |
| --- |
| **Détection et extraction de panneaux** |
| **Nom** |  |
| **Prénom** |  |
| **Classe** |  |

|  |  |
| --- | --- |
| **Question : 1** | **Note** |
| Ce sont des array à 3 dimensions : Lignes, Colonnes, Triplets |  |

|  |  |
| --- | --- |
| **Question : 2** | **Note** |
| Entiers codés sur 8 bits |  |

|  |  |
| --- | --- |
| **Question : 3** | **Note** |
| 3 entiers par pixel, soit 3 fois 1 octet (8 bits)Soit 3 octets |  |

|  |  |
| --- | --- |
| **Question : 4** | **Note** |
| Lignes x Colonnes x 3 octets600 x 800 x 31 440 000 octets1.4 Mo |  |

|  |  |
| --- | --- |
| **Question : 5** | **Note** |
| t = 0.000001T = 600\*800\*3\*tTemps de traitement estimé: 1.44 s |  |

|  |  |
| --- | --- |
| **Question : 6** | **Note** |
| **import** matplotlib**.**pyplot **as** plt**def** Affiche**(**fig**,**im**):** plt**.**figure**(**fig**)** plt**.**imshow**(**im**)** plt**.**axis**(**'off'**)** plt**.**show**()** |  |

|  |  |
| --- | --- |
| **Question : 7** | **Note** |
| Nom\_Image **=** "Image\_1.bmp"Image **=** plt**.**imread**(**Nom\_Image**)**Image **=** Image**[:,:,:**3**]**Affiche**(**1**,**Image**)** |  |

|  |  |
| --- | --- |
| **Question : 8** | **Note** |
| **import** numpy **as** np |  |

|  |  |
| --- | --- |
| **Question : 9** | **Note** |
| **def** NB**(**im**,**alpha**):** Nl**,**Nc **=** im**.**shape**[**0**:**2**]** imnb **=** np**.**copy**(**im**)** **for** c **in** range**(**Nc**):** **for** l **in** range**(**Nl**):** R**,**G**,**B **=** im**[**l**,**c**]** R**,**G**,**B **=** float**(**R**),**float**(**G**,),**float**(**B**)** Cond\_1 **=** R **>** alpha**\*(**G**+**B**)** Cond\_2 **=** R **-** max**(**G**,**B**)** **>** 2**\***alpha**\*(**max**(**G**,**B**)-**min**(**G**,**B**))** **if** Cond\_1 **and** Cond\_2**:** Pix **=** **[**0**,**0**,**0**]** **else:** Pix **=** **[**255**,**255**,**255**]** imnb**[**l**,**c**]** **=** Pix **return** imnb |  |

|  |  |
| --- | --- |
| **Question : 10** | **Note** |
| alpha **=** 0.75Image\_NB **=** NB**(**Image**,**alpha**)**Affiche**(**2**,**Image\_NB**)** |  |

|  |  |
| --- | --- |
| **Question : 11** | **Note** |
| l et c les nombres de lignes et colonnesO(l\*c) |  |

|  |  |
| --- | --- |
| **Question : 12** | **Note** |
| **def** Seuil**(**imng**,**k**):** Nl**,**Nc **=** imng**.**shape**[**0**:**2**]** imnb **=** np**.**copy**(**imng**)** **for** c **in** range**(**Nc**):** **for** l **in** range**(**Nl**):** N**,**\_**,**\_ **=** imng**[**l**,**c**]** **if** N **<=** k**:** Pix **=** **[**0**,**0**,**0**]** **else:** Pix **=** **[**255**,**255**,**255**]** imnb**[**l**,**c**]** **=** Pix **return** imnb |  |

|  |  |
| --- | --- |
| **Question : 13** | **Note** |
| K **=** **(**1**/**9**)\***np**.**array**([[**1**,**1**,**1**],[**1**,**1**,**1**],[**1**,**1**,**1**]])**Image\_NG **=** Convolution**(**Image\_NB**,**K**)**Image\_NB **=** Seuil**(**Image\_NG**,**127**)**Affiche**(**3**,**Image\_NB**)** |  |

|  |  |
| --- | --- |
| **Question : 14** | **Note** |
| **def** Liste\_voisins**(**l**,**c**,**Nl**,**Nc**):** vg **=** l**,**max**(**0**,**c**-**1**)** vd **=** l**,**min**(**Nc**-**1**,**c**+**1**)** vb **=** min**(**Nl**-**1**,**l**+**1**),**c vh **=** max**(**0**,**l**-**1**),**c **return** **[**vg**,**vd**,**vb**,**vh**]** | **def** Liste\_voisins**(**l**,**c**,**Nl**,**Nc**):** GDBH **=** **[[**l**,**c**-**1**],[**l**,**c**+**1**],[**l**+**1**,**c**],[**l**-**1**,**c**]]** **for** i **in** range**(**len**(**GDBH**)):** L**,**C **=** GDBH**[**i**]** **if** **not** **(**0**<=**L**<=**Nl**-**1 **and** 0**<=**C**<=**Nc**-**1**):** GDBH**[**i**]** **=** **[**l**,**c**]** **return** GDBH |  |

|  |  |
| --- | --- |
| **Question : 15** | **Note** |
| **def** Explorer**(**imnb**,**l**,**c**,**T**,**k**):** Nl**,**Nc**,**\_ **=** imnb**.**shape Pile **=** **[[**l**,**c**]]** Val **=** imnb**[**l**,**c**,**0**]** # Inutile de mettre dans while **while** len**(**Pile**)** **>** 0**:** l**,**c **=** Pile**.**pop**()** T**[**l**,**c**]** **=** k Lv **=** Liste\_voisins**(**l**,**c**,**Nl**,**Nc**)** **for** v **in** Lv**:** lv**,**cv **=** v T\_v **=** T**[**lv**,**cv**]** Val\_v **=** imnb**[**lv**,**cv**,**0**]** **if** T\_v**==-**1 **and** Val\_v **==** Val**:** Pile**.**append**([**lv**,**cv**])** |  |

|  |  |
| --- | --- |
| **Question : 16** | **Note** |
| **def** Zones**(**imnb**):** Nl**,**Nc **=** imnb**.**shape**[**0**:**2**]** T **=** **-**np**.**ones**([**Nl**,**Nc**],**dtype**=**'int64'**)** k **=** 0 **for** l **in** range**(**Nl**):** **for** c **in** range**(**Nc**):** **if** T**[**l**,**c**]** **==** **-**1**:** Explorer**(**imnb**,**l**,**c**,**T**,**k**)** k **+=** 1 **return** T |  |

|  |  |
| --- | --- |
| **Question : 17** | **Note** |
| Table **=** Zones**(**Image\_NB**)**Affiche**(**4**,**Table**)**Nb\_Zones **=** np**.**amax**(**Table**)** **+** 1**print(**"Nombre de zones: "**,**Nb\_Zones**)** |  |

|  |  |
| --- | --- |
| **Question : 18** | **Note** |
| Zone 1 | Nl**-**1**,**Nc**-**1**,**0**,**0 |  |
| Zone 2 | T**[**l**,**c**]** |
| Zone 3 | **(**Ml**\***Taille**+**l**)/(**Taille**+**1**)** |
| Zone 4 | **(**Mc**\***Taille**+**c**)/(**Taille**+**1**)** |
| Zone 5 | imnb**[**l**,**c**,**0**]** |
| Zone 6 | l\_min **=** l |
| Zone 7 | l\_max **=** l |
| Zone 8 | c\_min **=** c |
| Zone 9 | c\_max **=** c |
| Code | Donnees\_Zones **=** Donnees**(**Table**,**Nb\_Zones**,**Image\_NB**)** |

|  |  |
| --- | --- |
| **Question : 19** | **Note** |
| **def** LC\_Bilin**(**l**,**c**,**LO**,**LC**):** O1**,**\_**,**\_**,**O4 **=** LO C1**,**C2**,**C3**,**C4 **=** LC l12**,**c13 **=** O1 l34**,**c24 **=** O4 Vl **=** **[**l34**-**l**,**l**-**l12**]** Mat **=** **[[**C1**,**C2**],[**C3**,**C4**]]** Vc **=** **[**c24**-**c**,**c**-**c13**]** Res **=** Prod\_MV**(**Mat**,**Vc**)** Res **=** Prod\_VV**(**Res**,**Vl**)** Cst **=** 1**/((**l34**-**l12**)\*(**c24**-**c13**))** Res **=** **[**int**(**round**(**t**\***Cst**,**0**))** **for** t **in** Res**]** **return** Res |  |

|  |  |
| --- | --- |
| **Question : 20** | **Note** |
| **def** Coins**(**im**):** Nl**,**Nc **=** im**.**shape**[**0**:**2**]** hg **=** **[**0**,**0**]** hd **=** **[**0**,**Nc**-**1**]** bg **=** **[**Nl**-**1**,**0**]** bd **=** **[**Nl**-**1**,**Nc**-**1**]** **return** **[**hg**,**hd**,**bg**,**bd**]** |  |

|  |  |
| --- | --- |
| **Question : 21** | **Note** |
| **def** Resize**(**im**,**dim**):** Nl**,**Nc **=** dim im\_rec **=** 255**\***np**.**ones**([**Nl**,**Nc**,**3**],**dtype**=**'uint8'**)** LO **=** Coins**(**im**)** LC **=** Coins**(**im\_rec**)** **for** l **in** range**(**Nl**):** **for** c **in** range**(**Nc**):** L**,**C **=** LC\_Bilin**(**l**,**c**,**LC**,**LO**)** im\_rec**[**l**,**c**]** **=** im**[**L**,**C**]** **return** im\_rec |  |

|  |  |
| --- | --- |
| **Question : 22** | **Note** |
| **def** Distance\_uv**(**u**,**v**):** n **=** len**(**u**)** Dst **=** 0 **for** i **in** range**(**n**):** di **=** u**[**i**]-**v**[**i**]** Dst **+=** di**\*\***2 Dst **=** Dst**\*\*(**1**/**2**)** **return** Dst |  |

|  |  |
| --- | --- |
| **Question : 23** | **Note** |
| **def** Analyse**(**im**):** Nl**,**Nc **=** im**.**shape**[**0**:**2**]** L\_RGB **=** **[]** **for** l **in** range**(**Nl**):** **for** c **in** range**(**Nc**):** R**,**G**,**B **=** im**[**l**,**c**]** R **=** float**(**R**)** G **=** float**(**G**)** B **=** float**(**B**)** L\_RGB **+=** **[**R**,**G**,**B**]** **return** L\_RGB |  |

|  |  |
| --- | --- |
| **Question : 24** | **Note** |
| **def** Recadrage\_k**(**im**,**LD**,**T**,**k**):** \_**,**\_**,**\_**,**\_**,**l\_min**,**c\_min**,**l\_max**,**c\_max **=** LD**[**k**]** dl**,**dc **=** l\_max **-** l\_min **+** 1**,**c\_max **-** c\_min **+** 1 im\_rec **=** 255**\***np**.**ones**((**dl**,**dc**,**3**),**dtype**=**'uint8'**)** **for** l **in** range**(**dl**):** **for** c **in** range**(**dc**):** ll**,**cc **=** l**+**l\_min**,**c**+**c\_min **if** T**[**ll**,**cc**]==**k**:** im\_rec**[**l**,**c**]** **=** im**[**ll**,**cc**]** **return** im\_rec |  |

|  |  |
| --- | --- |
| **Question : 25** | **Note** |
| **from** math **import** sqrt**def** Etude\_Motif**(**imnb**,**imc**,**T**,**LD**,**a**):** imc **=** Resize**(**imc**,(**a**,**a**))** imc\_nb **=** NB**(**imc**,**alpha**)** L\_RGB\_c\_nb **=** Analyse**(**imc\_nb**)** Res **=** **[]** **for** k **in** range**(**len**(**LD**)):** im\_loc\_nb **=** Recadrage\_k**(**imnb**,**LD**,**T**,**k**)** im\_loc\_nb **=** Resize**(**im\_loc\_nb**,(**a**,**a**))** L\_RGB\_im\_loc\_nb **=** Analyse**(**im\_loc\_nb**)** d **=** Distance\_uv**(**L\_RGB\_c\_nb**,**L\_RGB\_im\_loc\_nb**)** dn **=** d**/(**255**\***a**\***sqrt**(**3**))** Res**.**append**([**dn**,**k**])** Res**.**sort**()** **return** Res |  |

|  |  |
| --- | --- |
| **Question : 26** | **Note** |
| Im\_Cercle **=** plt**.**imread**(**"Cercle.bmp"**)**a **=** 100Distances **=** Etude\_Motif**(**Image\_NB**,**Im\_Cercle**,**Table**,**Donnees\_Zones**,**a**)** |  |

|  |  |
| --- | --- |
| **Question : 27** | **Note** |
| **def** Selection**(**Ldst**,**dmax**,**LD**):** Res **=** **[]** **for** d**,**k **in** Ldst**:** **if** d **<=** dmax**:** Res**.**append**(**LD**[**k**])** **return** Res |  |

|  |  |
| --- | --- |
| **Question : 28** | **Note** |
| dmax **=** 0.4 |  |

|  |  |
| --- | --- |
| **Question : 29** | **Note** |
| Zones\_Sel **=** Selection**(**Distances**,**dmax**,**Donnees\_Zones**)** |  |

|  |  |
| --- | --- |
| **Question : 30** | **Note** |
| **def** Recadrage**(**im**,**l\_min**,**l\_max**,**c\_min**,**c\_max**):** dl**,**dc **=** l\_max **-** l\_min **+** 1**,**c\_max **-** c\_min **+** 1 im\_rec **=** np**.**zeros**((**dl**,**dc**,**3**),**dtype**=**'uint8'**)** **for** l **in** range**(**dl**):** **for** c **in** range**(**dc**):** ll**,**cc **=** l**+**l\_min**,**c**+**c\_min im\_rec**[**l**,**c**]** **=** im**[**ll**,**cc**]** **return** im\_rec |  |

|  |  |
| --- | --- |
| **Question : 31** | **Note** |
| ind **=** 1**for** zone **in** Zones\_Sel**:** \_**,**\_**,**\_**,**\_**,**l\_min**,**c\_min**,**l\_max**,**c\_max **=** zone Panneau **=** Recadrage**(**Image**,**l\_min**,**l\_max**,**c\_min**,**c\_max**)** Panneau **=** Resize**(**Panneau**,(**100**,**100**))** Affiche**(**100**+**ind**,**Panneau**)** plt**.**imsave**(**"Panneau "**+**str**(**ind**)+**".bmp"**,**Panneau**)** ind **+=** 1 |  |