UE[HT-P1-INFRES-002-M] :

*Nom: ........................................*

*Prénom: ....................................*

*Groupe: .....................................*

Programmation - Bases et Algorithmique – G4

|  |  |
| --- | --- |
| **Question théorique :** | **Sous total : /30** |

Questionnaire à choix multiples, entourez **la** bonne réponse. Une bonne réponse vaut 2 points, une mauvaise réponse vaut -1, ne rien répondre ne vaut aucun point. Pour le code on considère qu’il est dans une fonction et les bibliothèques sont incluses. Vous avez 50 minutes pour la théorie.

|  |  |
| --- | --- |
| **Questions** | **Réponses** |
| 1. lequel de ces langages n’est pas inspiré du C ? | 1. PHP 2. Java 3. HTML 4. Python |
| 1. Quel caractère n'est pas disponible dans le code ASCII ? | 1. $ 2. \* 3. à 4. % |
| 1. Lequel de ces noms de variables n’est pas permis en C? | 1. VaR 2. annee2015 3. 2015annee 4. annee\_2015 |
| 1. Le code suivant entraîne une erreur de compilation. Le compilateur affiche le message "expected primary-expression before 'int'". Quelle est l'erreur ?   int main() {  int long=10 ;  return 0 ; } | 1. Il manque un point-virgule à la première ligne 2. Il est interdit d'affecter une variable lors de sa déclaration 3. La variable long ne peut pas être de type entier 4. Une variable ne peut pas s'appeler 'long' car c'est un mot réservé. |
| 1. Qu’affichera le code suivant ?   int main() {  int x=11,y=0;  y=x%3;  printf("%d",y);  return 0;  } | 1. 11 2. 2 3. 1 4. 3 |
| 1. Quel convertisseur est employé pour afficher un nombre réel de taille double ? | 1. %f 2. %d 3. %s 4. %c |
| 1. Quel est l'opérateur de comparaison ? | 1. != 2. = 3. == 4. % |
| 1. Qu'affichera le code suivant ?   int main() {  int x=0,y=100;  y=x%10;  printf("%d",y);  return 0;  } | 1. 100 2. 0 3. 10 4. 1 |
| 1. De quelle bibliothèque a-t-on besoin pour utiliser strcpy() ? | 1. math.h 2. stdlib.h 3. time.h 4. string.h |
| 1. Le code suivant :   int i =1;  switch (i)  {  case 1 : printf("1");  case 2 : printf("2") ;  case 3 : printf("3");break ;  default : printf("4");  }  Affichera : | 1. 3 2. 123 3. 12 4. 1234 |
| 1. Qu'affichera le script suivant ?   float x=2.5 ;  if (x>3) printf("%f", x) ;  else if (x/2>1) printf("%f",2\*x) ;  printf("ok") ; | 1. 5.000000ok 2. 5.000000 3. 2.500000 4. ok |
| 1. A quoi sert l’instruction BREAK ? | 1. à sauter une instruction 2. à mettre fin à un bloc d'instructions 3. à mettre fin à un programme 4. à détruire une variable afin d'en libérer l'espace mémoire |
| 1. Soit un programme contenant les lignes suivantes :   int i = 0;  int j = 0;  for (i = 0; i < 2; i = i + 1)  {  for (j = 0; j < 3; j = j + 1)  {  printf("%d ", i);  }  }  qu'est ce qui sera affiché ? | 1. 1 2 1 2 3 2. 0 1 2 0 1 2 3. 0 0 0 1 1 1 4. 0 1 0 1 0 1 0 1 |
| 1. Qu'affichera le code suivant ?   int i,x=1;  for (i=1 ; i<5 ; i++) x=(i+1)\*2;  printf("%i",x); | 1. 5 2. 11 3. 1 4. 10 |
| 1. Soit prix[], un tableau de 20 réels contenant des prix d'articles. Quel code permet d'inverser l'ordre de ces prix (de façon à échanger le 1er avec le dernier, le 2e avec l'avant-dernier, etc.) ? | 1. for (i=0 ; i<10 ; i++) {   prix[i]=prix[i]+prix[19-i];  prix[19-i]=prix[i]-prix[19-i];  prix[i]=prix[19-i]-prix[i];  }   1. for (i=0 ; i<10 ; i++) {   prix[i]=prix[i]+prix[19-i];  prix[19-i]=prix[i]-prix[19-i];  prix[i]=prix[i]-prix[19-i];  }   1. for (i=0 ; i<10 ; i++) {   prix[i]=prix[i]+prix[19-i];  prix[19-i]=prix[19-i]-prix[i];  prix[i]=prix[i]-prix[19-i];  }   1. for (i=0 ; i<10 ; i++) {   prix[i]=prix[i]+prix[19-i];  prix[19-i]=prix[19-i]-prix[i];  prix[i]=prix[19-i]-prix[i];  } |
| 1. Quel est le nombre de dimensions maximum d'un tableau en langage C ? | 1. 3 2. 2 3. 1 4. Pas de limite précise |
| 1. Qu'affichera le script suivant ?   char mot[32]="boa" ;  int i=0 ;  while (mot[i]!='\0') {  mot[i]=mot[i]+i ;  i++ ;  }  printf("%s",mot) ; | 1. bpc 2. cpb 3. bpb 4. boa |
| 1. Soit la déclaration de type de structure suivante, comment déclarer une variable structurée de type joueur ?   struct joueur {  char pseudo[32];  int score;  } ; | 1. joueur j1; 2. struct j1; 3. int joueur; 4. struct joueur j1; |
| 1. Quel mot-clé permet de déclarer une procédure ? | 1. INT 2. VOID 3. PROC 4. BREAK |
| 1. Dans le code source suivant, quelle est l'erreur ?   float A;  float fct(float A) {  float C;  C = A\*A;  return C ;  }  int main() {  float B=6.2;  int C=0;  C=fct(B);  } | 1. La double déclaration de C 2. La fonction fct attend un paramètre réel et reçoit un entier 3. La double déclaration de A 4. La fonction fct renvoie un paramètre réel qui est stocké dans un entier |

*Nom: ........................................*

*Prénom: ....................................*

*Groupe: .....................................*

|  |  |
| --- | --- |
| **Question sur ordinateur :** | **Sous total: /70** |

Le programme à réaliser doit permettre l’encodage d’une suite de nombres entiers dans un tableau, l’affichage de la suite de nombre mémorisée au format matriciel et le produit des deux diagonales.

Le programme retournera également le nombre de valeurs paires et impaires au sein du tableau.

Les nombres doivent être mémorisés dans un tableau 4x4 (16 valeurs) et le tableau doit être déclaré au niveau du programme principal.

Le programme principal doit se présenter de la manière suivante :

***main()***

***{***

***// déclaration du tableau destiné à contenir les valeurs entières***

***// affichage de votre nom, prénom, classe***

***// appel de la procédure d’encodage des valeurs***

***// appel de la procédure d’affichage des valeurs au format matriciel***

***// appel de la fonction de calcul du produit des valeurs des deux diagonales principales***

***(l’affichage de la valeur calculée doit se faire à l’intérieur du programme principal)***

***// appel de la ou les fonction(s) de calcul permettant de retourner le nombre de valeurs paires (l’affichage de la valeur calculée doit se faire à l’intérieur du programme principal)***

***// appel de la ou les fonction(s) de calcul permettant de retourner le nombre de valeurs impaires (l’affichage de la valeur calculée doit se faire à l’intérieur du programme principal)***

***}***

Particularités souhaitées au niveau du comportement des différents sous-programmes :

\* La procédure d’encodage des valeurs doit posséder des contrôles de saisie (16 valeurs au maximum et chaque valeur entrée doit être comprise entre 0 et 10)

\* Décrivez le fonctionnement de votre procédure d’encodage des valeurs (Insérer en commentaire une explication/justification de la technique d’échange de paramètre choisie pour cela).

Exemple de résultat souhaité à la page suivante :

|  |
| --- |
| #################################  Code Jean 1IRT  #################################  Remplissage de la matrice  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Entrer la valeur de la ligne 1, colonne 1 : 1  Entrer la valeur de la ligne 1, colonne 2 : 1  Entrer la valeur de la ligne 1, colonne 3 : 2  Entrer la valeur de la ligne 1, colonne 4 : 3  Entrer la valeur de la ligne 2, colonne 1 : 4  …  Affichage de la matrice  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  **1**  1 2 **3**  4 **5 6** 7  8 **9 10** 5  **2**  4 2 **8**  Calcul du produit des deux diagonales  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Produit diagonal : 129.600  Quantité de valeurs paires  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Quantité : 9  Quantité de valeurs impaires  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Quantité : 7 |
|  |

***Lorsque vous aurez terminé, vous devrez me remettre un dossier portant vos nom et prénom dans lequel se trouveront le code source commenté de votre programme ainsi que le fichier exécutable créé. Durée maximale de l’épreuve : 2H30***

***Bon travail.***