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

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

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

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

Programmation - Bases et Algorithmique – G3

|  |  |
| --- | --- |
| **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. Le langage C est un langage ? | 1. Composé 2. Compilé 3. Ecrit 4. Interprété |
| 1. Qu’est-ce que une instruction ? | 1. La traduction du programme en langage machine 2. L'espacement situé avant une ligne de code 3. Une ligne de code 4. Un symbole compris par la machine |
| 1. Quel mot-clé permet de déclarer une variable de type réel ? | 1. real 2. short 3. float 4. short double |
| 1. Qu’est-ce qu’affecter une variable ? | 1. Stocker une valeur dans une variable 2. Effacer une variable 3. Copier une variable 4. Allouer un espace mémoire à la variable |
| 1. Avec la fonction printf, comment affiche-t-on le caractère ‘%’ | 1. printf(« % ») ; 2. printf(« %% ») ; 3. printf(« \% ») ; 4. printf(«/% ») ; |
| 1. Qu’affichera le code suivant ?   int main() {  int x=1,y=2;  y=x\*y;  x=x\*y;  x=x\*y;  printf("%d %d",x,y);  return 0;  } | 1. 3 1 2. 7 2 3. 7 6 4. 4 2 |
| 1. Parmi ces instructions, l'une ne donne pas le même résultat que les autres. Laquelle ? | 1. X+=1; 2. X++; 3. X=1+X; 4. X=X+1; 5. X=1+1; |
| 1. Qu'affichera le code suivant ?   int main() {  int x=5,y=100;  y=x%15;  printf("%d",y);  return 0;  } | 1. 15 2. 0 3. 5 4. 1 |
| 1. De quelle bibliothèque a-t-on besoin pour utiliser cos() ? | 1. stdio.h 2. stdlib.h 3. math.h 4. string.h |
| 1. Qu'affichera le code suivant ?   int x=10;  if (x<10) printf("1");  else printf("2");  printf("3"); | 1. 13 2. 1 3. 123 4. 23 5. 3 |
| 1. Soit le code suivant, pour quelle valeur de X, le message « hello » s'affichera ?   if ((X>5) && (X%2==0)) printf("hello"); | 1. 2 2. 10 3. 5 4. 11 |
| 1. A quoi sert l’instruction CONTINUE ? | 1. à passer à l’itération suivante 2. à mettre fin à un bloc d'instructions 3. à continuer le programme 4. à détruire une variable afin d'en libérer l'espace mémoire |
| 1. Que vaudra x après ce code ?   int x=0,i,j;  for (i=0 ; i<10 ; i++) {  for (j=0 ; j<=10 ; j++) {  x++;  }  } | 1. 20 2. 110 3. 10 4. 1000 5. 90 |
| 1. Comment traduire en français le code suivant ?   while (i<7) i++; | 1. Jusqu'à ce que i soit plus petit que 7, l'incrémenter 2. Si i est plus petit que 7, l'incrémenter 3. Tant que i est plus petit que 7, l'incrémenter 4. Pour que i soit plus petit que 7, l'incrémenter |
| 1. Soit year[], un tableau de 25 entiers contenant l'année de naissance de 25 étudiants. Quel code permet d'afficher toutes les années de naissance inférieures ou égales à 1990 ? | 1. for (i=0 ; i<25 ; i++) {   if (year[i]<=1990) printf("%i",year[i]);  }   1. for (i=0 ; year[i]<1990 ; i++) {   printf("%i",year[i]);  }   1. if (year[i]<=1990) {   for (i=0 ; i<25 ; i++) printf("%i",year[i]);  }   1. for (i=0 ; i<25 ; i++) {   printf("%i",year[i]<=1990);  } |
| 1. Qu'affichera le code suivant ?   char mot[32]="castor";  int i,x;  for (i=0 ; mot[i]!='\0' ; i++) {  printf("%c",mot[i+1]);  } | 1. astor 2. castor 3. dbtups 4. bdtups |
| 1. Qu'affichera le code suivant ?   char mot[32]="boa" ;  int i=0 ;  while (mot[i]!='\0') {  mot[i]=mot[i]+i ;  i++ ;  }  printf("%s",mot) ; | 1. boa 2. cpb 3. bpc 4. bpb |
| 1. Soit la déclaration suivante, combien d'octets occupera en mémoire la variable nom ?   char nom[32]; | 1. 256 2. 8 3. 128 4. 64 5. 32 |
| 1. Laquelle de ces différences entre tableaux et structures est vraie ? | 1. Les structures peuvent contenir des fonctions, pas les tableaux. 2. Les structures ne nécessitent pas de déclaration, les tableaux oui. 3. Les structures peuvent contenir des valeurs de types différents, pas les tableaux. 4. Les structures ne peuvent contenir qu'une seule valeur, pas les tableaux. |
| 1. Qu'affichera le code suivant ?   int x=5;  int \*p;  p=&x;  printf("%d",p); | 1. 5 2. x 3. L'adresse de p 4. L'adresse de x |

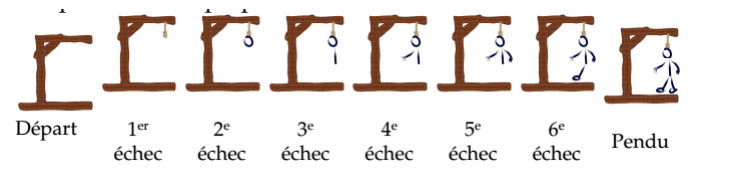
|  |  |
| --- | --- |
| *Nom: ........................................*  *Prénom: ....................................*  *Groupe: .....................................*  **Question sur ordinateur :** | **Sous total: /70** |

Écrire un programme reprenant le « jeu du pendu ». Pour cela, on suppose qu’on dispose d’un fichier mots.txt contenant une liste de mots, à raison d’un par ligne.

Au démarrage, le programme utilise une variable RANDOM pour prendre un mot au hasard dans le fichier mots.txt.

Le script doit alors écrire à l’écran le symbole du pendu et un «undersocre» pour chaque lettre du mot choisi.

Une lettre est alors demandée à l’utilisateur. Si cette lettre figure parmi les caractères du mot non encore découverts, alors le joueur vient de découvrir une lettre qui sera par la suite toujours affichée à la place du (ou des) «underscore(s) » correspondant(s) avec les autres caractères découverts. Sinon, le joueur a fait une erreur. Le joueur a droit à 7 erreurs. Le dessin du pendu doit se dessiner au fur et à mesure des échecs comme sur la figure suivante (en simplifié bien sûr).



S’il découvre la totalité du mot, alors il a gagné et le script se termine avec un code de retour affichant un texte de victoire.

S’il fait plus de 7 erreurs, il a perdu et le code de retour est l’affichage d’un texte de défaite.

Un deuxième fichier contenant les règles du jeu devra aussi exister. Un menu au départ proposera de rentrer dans le jeu, le quitter ou lire l’aide.

Votre programme doit être impérativement décomposé en plusieurs de sous programmes (procédures ou fonctions). Le programme principal doit impérativement se présenter de la manière suivante :

***main()***

***{***

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

***//Appel du menu***

***// Appel de la fonction jouer***

***// Appel de la procédure règles***

***// Appel de la procédure d’affichage du résultat***

***}***

***En fin de test, me remettre un dossier portant votre nom suivi de votre prénom dans lequel se trouveront le code source commenté de votre programme ainsi que le fichier exécutable.***

***L’utilisation des clés USB, la mise en réseaux (wifi ou autres) sont interdits durant l’interrogation. Toute tentative de fraude entraînera automatiquement une cote de 0/20 pour cette interrogation. Durée maximale de l’épreuve : 2H30***

***Bon travail.***