QCM : Langage C++ complément

Nom : ……………………………….

Prénom : ………………………….

**Attention, plusieurs bonnes réponses sont parfois possibles !**

|  |  |  |
| --- | --- | --- |
|  | **Question** | **Réponse** |
| 1 | Soit le programme suivant :  int A = 5;  int B(4);  int \*P = &A;  B = (\*P) + 2;  cout << "B = " << B;  Il affiche : | 🞏 B = 4 |
| 🞏 B = 7 |
| 🞏 B = 3 |
| 🞏 B = 6 |
| 2 | Si P est un pointeur que représente cette déclaration ?  cout << &P; | 🞏 L'affichage de la valeur de P |
| 🞏 L'affichage de la valeur pointée par P |
| 🞏 L'affichage de l'adresse de P |
| 🞏 Le caractère P |
| 3 | Soit l'extrait de programme suivant :  int i = 3, j = 6;  int \*p1, \*p2;  p1 = &i;  p2 = &j;  \*p1 = \*p2;  Que peut-on dire ? | 🞏 P1 =3 et pointe sur l'entier i |
| 🞏 La valeur de P2 est égale à l'adresse de J |
| 🞏 La valeur pointée par P1 est 6 |
| 🞏 P2 pointe sur i |
| 4 | Soit l'extrait de programme suivant :  struct Client  {  int Age;  };    Client structure1;  Client \*pstr = &structure1;  ……………………………………;  Avec la syntaxe de la flèche "->" comment peut-on compléter le programme et affecter 35 à l'Age de la structure1 via le pointeur? | 🞏 Age -> pstr = 35 |
| 🞏 -> structure1.Age =35 |
| 🞏 Age -> structure1 = 35 |
| 🞏 structure1->Age = 35 |
| 🞏 pstr -> Age = 35 |
| 5 | Soit l'extrait de programme suivant :  void test1(int \*i){\*i = 2;}  void test2(int &i){i = 4;}  int main(){  int A, B;  test1(………);  test2(………);  }  Donnez les bons appels des fonctions ? | 🞏 test1(&A); et test2(B) |
| 🞏 test1(&A); et test2(\*B) |
| 🞏 test1(A); et test2(B) |
| 🞏 test1(\*A); et test2(&B) |
| 6 | Soit le programme suivant :  int A = 7;  float &var = A;  Ce programme est-il correcte ? | 🞏 oui |
| 🞏 non |
| 🞏 seulement si float était un type équivalent à int |
| 🞏 Parfois oui |
| 7 | Soit le programme suivant :  class A{  public:  A();  ~A();  void SetA(short val);  private:  short membre;  };  Que peut-on dire ? | 🞏 A est une classe. |
| 🞏 membre est un attribut de type float. |
| 🞏 ~A est un destructeur. |
| 🞏 SetA est une méthode de surcharge d'opérateur. |
| 🞏 A() est une méthode particulière appelée constructeur. |
| 8 | Un attribut de type public peut-il être accessible directement de l'extérieur de la classe ? | 🞏 Oui |
| 🞏 Non |
| 🞏 Parfois |
| 9 | Soit le programme suivant :  class A{  public:  A();  A(int I,int j);  void SetA(short val);  private:  int age;  int taille;  };  Comment peut-on instancier un objet de type A nommé monA initialisé avec les deux valeurs 3 et 7 ? | 🞏 A monA = 3, 7; |
| 🞏 A monA(3, 7); |
| 🞏 A monA(3)(7); |
| 🞏 monA A(3, 7); |
| 10 | Soit le programme suivant :  int tableau[100];  int \*pi=tableau;  int A(2);  int B(2);  A++;  \*(tableau+B) = 4;  \*(tableau+A) = 7;  cout << tableau[3] << endl;  cout << \*(pi + B);  Il affiche ? | 🞏 2  2 |
| 🞏 4  7 |
| 🞏 7  4 |
| 🞏 2  3 |