

1. Pour chaque famille de programme donnez un exemple de langage.

| Programme | Exemple de langage |
|------------------|---------------------------|
| Compilé | |
| Semi Compilé | |
| Interprété | |

2. On dispose de 2 machines informatiques différentes et d'un programme source. Expliquez comment ce programme est rendu portable pour être exécutable sur ces 2 machines lorsque le programme est de type compilé ou de type semi compilé.
3. Donnez les différentes étapes qui permettent de produire un exécutable à partir d'un programme écrit en langage C++. Vous préciserez les extensions des fichiers produits.

| Etape | Extensions des fichiers produits |
|--------------|---|
| | |
| | |
| | |

4. Quel est la fonction principale d'un programme écrit en langage C++ ?
5. Soit un programme écrit en langage C++. Que fait le précompilateur lorsqu'il rencontre la directive `#include` ?
6. Quelle est la classe utilisée qui gère les entrées sorties standards comme des flux de données ?
7. Comment se désignent les entrées sorties standards, en langage C++, pour être gérées comme des flux de données ?
8. Lorsque l'on gère les périphériques standards comme des flux de données, on utilise des opérateurs particuliers.
Pour les opérateurs représentés ci-dessous, donnez leur nom et les opérations qu'ils réalisent.

| Opérateur | Nom | Opération |
|-----------------------|------------|------------------|
| <code>>></code> | | |
| <code><<</code> | | |

9. Pour les types ci-dessous donnez la taille, en octets, occupée en mémoire par une variable.

| Type | Taille en octets | Type | Taille en octets |
|----------------------|-------------------------|-----------------------|-------------------------|
| <i>char</i> | | <i>unsigned short</i> | |
| <i>unsigned char</i> | | <i>long</i> | |
| <i>float</i> | | <i>short</i> | |
| <i>bool</i> | | <i>double</i> | |

10. Identifiez les noms de variables dont la syntaxe est correcte du point de vue du langage C++.

```
distance    vitesse5    vitesseMaxi    2Distance    magasin%    main
valeurStock    étatIntialisation    _vitesse    vitesse+    int    magasin1
include    stock    20Magasin    référenceStock    float    _stock
```

11. Quelle est la particularité, sur la taille, d'une variable de type int ?

12. Soit l'extrait de programme :

```
float hauteur = 10.5 ;
double volume = 5000;
-----
t4 = sizeof(volume) ;
cout << t4 ;
```

11.1 Donnez le type de base pour déclarer la variable t1 et t4.

```
t1 = sizeof(hauteur) ;
cout << t1 ;
```

11.2 Donnez les valeurs qui sont affichées.

13. Donnez la plage de valeur couverte par une variable déclarée de type short

14. Donnez la plage de valeur couverte par une variable déclarée de type long

15. Complétez le tableau lors de la déclaration dans un programme C++.

| Donnée | Valeurs | Déclaration |
|-------------------------|---------------------------|-------------|
| <i>age</i> | 0 à 99 (entier) | |
| <i>kilometrage</i> | 0 à 200 000 (entier) | |
| <i>autonomie</i> | 10 à 100 (entier) | |
| <i>hauteurRelative</i> | - 1000 à 1000(réel) | |
| <i>seuil</i> | - 500 à + 500 (entier) | |
| <i>MOITIE</i> | 0.5 (constante) | |
| <i>offsetRts</i> | - 100 à 100 (réel) | |
| <i>seuilRst</i> | 0 à 10 (entier) | |
| <i>TEMPERATURE_MAXI</i> | -5 (constante) | |
| <i>volumeEssence</i> | 0 à 60 (réel) | |
| <i>MIN_TEMPERATURE</i> | -25 (constante) | |
| <i>TVA</i> | 19.6 (constante) | |

16. Soit l'opération $(c+s)*f$ dont le résultat vaut 789.34
Donnez le résultat de l'exécution des instructions `cout .. ;`

```
int var=0 ;
-----
var=(c+s)*f;

cout << ((c+s)*f) ;
cout << var ;
```

17. Donnez une écriture équivalente des instructions.

| Instruction | Instruction équivalente |
|---|-------------------------|
| <code>indCellule++ ;</code> | |
| <code>nbre -=50 ;</code> | |
| <code>ptrTab-- ;</code> | |
| <code>ddp *= 15.5 ;</code> | |
| <code>ddp = ++uSecteur * calibre ;</code> | |
| <code>couleur = ++rouge + base ;</code> | |

18. Donnez le résultat (vrai ou faux) des évaluations suivantes :

Lorsque vitesse vaut 50 ; VITESSE_MAXI vaut 110 ; VITESSE_MINI vaut 50

| Expression | Evaluation |
|---|------------|
| <code>vitesse == VITESSE_MAXI</code> | |
| <code>(vitesse >= VITESSE_MINI)&&(vitesse <= VITESSE_MAXI)</code> | |
| <code>(vitesse < VITESSE_MINI) (vitesse > VITESSE_MAXI)</code> | |

Lorsque hauteur vaut 410 ; HAUTEUR_MAXI vaut 650 ; HAUTEUR_MINI vaut 410

| Expression | Evaluation |
|---|------------|
| <code>hauteur <= HAUTEUR_MINI</code> | |
| <code>(hauteur != HAUTEUR_MINI) (hauteur != HAUTEUR_MAXI)</code> | |
| <code>(hauteur > HAUTEUR_MINI)&&(hauteur < HAUTEUR_MAXI)</code> | |

Lorsque mesure vaut 5; MESURE_REF vaut 5 ; MESURE_MAX vaut 10

| Expression | Evaluation |
|--|------------|
| <code>(mesure >= MESURE_REF)&&(mesure < MESURE_MAX)</code> | |
| <code>mesure == MESURE_REF</code> | |
| <code>(mesure < MESURE_REF) (mesure > MESURE_MAX)</code> | |

19. Analyse

| | |
|--|--|
| Soit la séquence : | Donnez les valeurs prises par valA et valB à l'issue de la séquence. |
| <pre> unsigned char valA= 0 ; unsigned char portA = 0x01 ; unsigned char valB = 0 ; unsigned char portB = 0x10 ; valA = portA & 0xFE ; valB = portB << 1 ; </pre> | <pre> valA = valB = </pre> |

| | |
|---|--|
| Soit la séquence : | Donnez les valeurs prises par varA et varB à l'issue de la séquence. |
| <pre> unsigned char varA = 0x00 ; unsigned char port = 0xf0 ; unsigned char varB = 0x00 ; unsigned char dir = 0x08 ; varA = port 0x01 ; varB = dir >> 1 ; </pre> | <pre> varA = varB = </pre> |

20. Codage

| Soit le programme décrit ci-dessous | Etablissez le codage en C++ de cette séquence |
|--|---|
| <p><i>COUPLE_SEUIL = 100 mN 'constante</i></p> <p><i>choixCouple : entier ' 0 à 250 mN</i></p> <p><i>coupleMesuré : réel ' 0 à 300 mN</i></p> <p><i>Saisir au clavier choixCouple</i></p> <p><i>Afficher choixCouple</i></p> <p><i>Calculer la somme</i> <i>COUPLE_SEUIL + choixCouple</i> <i>et Afficher le résultat.</i></p> | |

21. Soit l'extrait d'un programme source :

```

unsigned short nbre=0xf703; //1
unsigned short masque=0x8000; //2
char out[16], *ptr=out; //3

cout << nbre <<" en binaire = "<<endl; //4

for (int i=0; i<=15; i++){ //5
    *ptr = ((masque & nbre) ? '1' : '0') ; //6
    masque >>=1; //7
    ptr++; //8
    cout << out[i] ; //9
}

```

Questions :

- Identifiez les variables qui sont utilisées dans cet extrait.
Donnez la taille, en octets, des types de variables utilisées dans ce programme ?
- Quelle est le résultat de l'exécution de l'instruction en 4 ?
- De quel type est l'opération (arithmétique, comparaison, .) réalisée en 8 ?
Quelle opération est réalisée en 8 ?
- De quel type est l'opération (arithmétique, comparaison, .) réalisée en 7 ?
Quelle opération est réalisée en 7 ?
- Proposez une autre forme d'écriture pour les instructions en 7 et 8.
- De quel type (au sens algorithmique) est la structure de contrôle qui est utilisée en 5 ?
Dans ce programme, quelle est la condition pour sortir de cette structure ?
- L'opérateur ternaire ((condition) ? val1 : val2) Si condition = 1 alors retourne val1, sinon retourne val2. Expliquez ce que fait l'instruction 6, lorsque masque vaut 0x8000 et 0x0008.