

Partie I

QCM : La feuille de réponse est annexée au sujet. Vous la détacherez et la glisserez dans votre copie. Pour chacune des questions ci-dessous, indiquez, sur la feuille de réponse, la lettre correspondant à la bonne réponse, de manière claire et lisible (les réponses illisibles seront annulées).

Points : 3

1) Quelle sera la valeur de `j` après l'exécution du bloc d'instructions suivant :

```
j = 0;
switch (i) {
    case 3:
        j++;
    case 2:
        j+=2;
    case 1:
        j+=3;
}
```

- a) 3 si `i=3`, 2 si `i=2`, 1 si `i=1`
- b) 1 si `i=3`, 2 si `i=2`, 1 si `i=3`
- c) 6 si `i=3`, 5 si `i=2`, 3 si `i=1`
- d) 6 quelque soit la valeur de `i`
- e) Aucune des réponses précédentes n'est correcte

2) L'instruction « `if (size=0) i++ ; else i--;` »

- a) contient une erreur
- b) incrémente la variable `i` si `size` est positif
- c) décrémente la variable `i` si `size` est négatif
- d) incrémente la variable `i` si `size` est différent de 0
- e) décrémente la variable `i` si `size` est différent de 0

3) Le programme « `int main(){ int b=5, x=12, z ; z = (2+5*x+4)/b-3 ; printf ("%d", z); }` » affiche la valeur :

- a) 10
- b) 10.2
- c) 17
- d) 17.6
- e) Aucune des réponses précédentes n'est correcte

4) L'opérateur `==` permet en langage C de :

- a) réaliser une affectation
- b) tester une égalité
- c) comparer deux variables
- d) convertir un `int` en `float`
- e) Aucune des réponses précédentes n'est correcte

- 5) L'expression « $e * e$; » :
- a) contient une erreur de syntaxe
 - b) équivaut à l'expression « $e = e * 2$; »
 - c) calcule le carré de e (e^2)
 - d) aucune des réponses précédentes n'est correcte
- 6) Les instructions suivantes « `if (h>5 && h<=18) printf("Bonjour") ; else printf("Bonsoir") ;` »
- a) affichent toujours « Bonjour »
 - b) affichent toujours « Bonsoir »
 - c) affichent « Bonjour » si la valeur de h est égale à 5
 - d) affichent « Bonsoir » uniquement si h est inférieur à 5 ou si h est supérieur ou égale à 18
 - e) Aucune des réponses précédentes n'est correcte
- 7) L'instruction « `if (a%2 == 0) printf("bonjour") ;` »
- a) contient une erreur de syntaxe
 - b) n'affiche rien, quelque soit la valeur de a
 - c) affiche « bonjour » quand a est un entier impair
 - d) affiche « bonjour » quand a est un entier pair
 - e) Aucune des réponses précédentes n'est correcte
- 8) Le programme
- ```
int main () {
 int ASCII;
 ascii = getchar() ;
 printf ("%d\n", ascii);
}
```
- a) contient une erreur
  - b) va afficher le code ASCII du caractère fourni par l'utilisateur
  - c) va afficher le caractère fourni par l'utilisateur
  - d) Aucune des réponses précédentes n'est correcte
- 9) Que fait l'expression : « `if (e%2 != 0) p*=y;` » ?
- a) Rien, elle contient une erreur de syntaxe
  - b) Elle multiplie la valeur de  $p$  par celle de  $y$
  - c) Elle multiplie la valeur de  $p$  par celle de  $y$  uniquement si  $e$  est impair
  - d) Elle multiplie la valeur de  $p$  par celle de  $y$  uniquement si  $e$  est pair
  - e) Aucune des réponses précédentes n'est correcte
- 10) Le programme
- ```
int main() { int i=0 ; while(i<10) printf("%d ",i); i++; }
```
- a) va afficher 9 nombres
 - b) va afficher 11 nombres
 - c) va afficher 10 nombres
 - d) va boucler indéfiniment
 - e) contient une erreur de syntaxe
 - f) Aucune des réponses précédentes n'est correcte

Partie II

Questions à lacunes : les lignes 8, 10, 13, 15 doivent être complétées par vos soins. Pour chacune d'elles, indiquez le n° de la ligne et écrivez la ligne complétée. Les autres lignes ne doivent pas être recopiées. Les réponses illisibles seront annulées.

Points : 2

11) Remplissez les lacunes soulignées dans le programme ci-dessous :

1	#include <stdio.h>
2	
3	int main ()
4	{
5	float x,y,t;
6	
7	printf ("Entrez x : ");
8	_____ ("____", &x);
9	printf ("Entrez y : ");
10	_____ ("____", &y);
11	
12	if (x<y) {
13	t = _____;
14	x = y;
15	y = _____;
16	}
17	printf ("Ecart : %f", (x-y));
18	}

Partie III

Traces : Réalisez la trace d'exécution des programmes indiqués sur votre copie. Ecrivez de manière claire et lisible selon le modèle ci-dessous (réponses illisibles seront annulées).

(Question 12) Pont d'observation	Variable	Variable	...
Pont d'observation 1			
...			

Points : 5

12) Tracez l'exécution du programme suivant.

#include<stdio.h>
main(){
int min, max;
int reste;
int i, n;
min = 2;
max = 5;
printf ("Les premiers entre %d et %d sont : ", min, max);
n = min;
//pont d'observation 1
while (n<=max) {
i=2;
reste = 1;
//pont d'observation 2

```
do {
    if (n != i)
        reste = n%i;
    i++;
    //pont d'observation 3
} while (i<=n && reste != 0);
//pont d'observation 4
if (reste != 0) {
    printf (" %d ", n);
}
n += 1; //pont d'observation 5
}
//pont d'observation 6
}
```

Partie IV

Pour chacune des questions ci-dessous, écrivez le programme pour le problème indiqué, suivant les indications données. Ecrivez de manière claire et lisible (réponses illisibles ne seront pas prises en considération).

Points : 10 (question 13 : 6 points, question 14 : 4 points)

13) Un nombre est parfait quand il est égal à la somme de ses diviseurs. Par exemple :

6	=	1 + 2 + 3	est un nombre parfait.
28	=	1 + 2 + 4 + 7 + 14	est aussi un nombre parfait.
8	≠	1 + 2 + 4	n'est pas un nombre parfait.

Rappel : les diviseurs d'un nombre entier positif n sont de tous les nombres entiers compris entre 1 et $n-1$ qui divisent n .

Ecrire un programme qui calcule et affiche les nombres parfaits entre 5 et 100.

14) La compagnie d'assurance « MegaAssur » veut calculer le bonus/malus des conducteurs, ce calcul va tenir compte du nombre d'accidents survenus dans l'année précédente ($nbacc$), ainsi que du bonus/malus de l'année précédente ($cbmprec$).

Pour cela « MegaAssur » calcule un coefficient d'accident ($coef$) de la manière suivante :

- si le conducteur a eu un ou des accidents $coeff = 1 + 0,05 \times nbacc$
- si le conducteur n'a pas eu d'accident, il sera de 0,95

Le bonus/malus de l'année (cbm) correspond à celui de l'année précédente ($cbmprec$) multiplié par le coefficient ($coef$). Mais la valeur du bonus/malus est limitée 3,5 (valeur maximale). Donc, si le bonus/malus calculé pour un conducteur dépasse cette valeur maximale, il sera égal à 3,5.

Ecrire un programme qui calcule et affiche le bonus/malus d'un conducteur. Ce programme doit demander à l'utilisateur le bonus/malus correspondant à l'année précédente ($cbmprec$), ainsi que le nombre d'accidents ($nbacc$), et après la lecture de ces données il doit calculer $coef$ puis calculer le bonus/malus pour l'année courante et l'afficher.

Feuille réponse pour la partie I (QCM)

<i>Question</i>	<i>Réponse</i>
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	