## Questions QCM

Que font les instructions suivantes:

e/=2;

y\*=y;

p\*=y;

Récrivez-les différemment.

Que ferait l'instruction: if (e%2) p\*=y;

5. printf("%i", A)

a. Affiche le caract`ere dont le code ASCII est stock´e dans la variable A

b. Affiche le code ASCII du caract`ere ’A’

c. Affiche le code ASCII du caract`ere stock´e dans la variable A

d. Affiche le caract`ere ’A’

Les instructions i=0 ;

while(i<10)

printf("%i ", i) ;

i++ ;

a. vont afficher 9 nombres

b. vont afficher 11 nombres

c. vont afficher 10 nombres

d. vont boucler ind´efinimment

10. if (a<5) printf("Bonjour") ; a=a+1 ;

a. Affiche bonjour et augmente la valeur de a quelque soit a

b. N’affiche pas bonjour et n’augmente pas la valeur de a quelque soit a

c. Augmente la valeur de a quelque soit a

d. Affiche bonjour quelque soit a

13. printf("%c", ’A’)

a. Affiche le caract`ere ’A’

b. Affiche le code ASCII du caract`ere stock´e dans la variable A

c. Affiche le code ASCII du caract`ere ’A’

d. Affiche le caract`ere dont le code ASCII est stock´e dans la variable A

14. if (a%2 == 0) printf("bonjour") ;

a. N’affiche rien (quelque soit la valeur de a)

b. Affiche bonjour quand a est un entier impair

c. D´eclenche le message d’erreur invalid lvalue in assignment

d. Affiche bonjour quand a est un entier pair

5. == permet en langage C de :

a. r´ealiser une affectation

b. tester une ´egalit´e

c. comparer deux identificateurs

d. convertir un int en float

main(){while(1);}

Evaluer les expressions suivantes en supposant

**a=20   b=5   c=-10   d=2   x=12   y=15**

Notez chaque fois la valeur rendue comme résultat de l'expression et les valeurs des variables dont le contenu a changé.

**(1)   (5\*X)+2\*((3\*B)+4)**

**(2)   (5\*(X+2)\*3)\*(B+4)**

**(3)   A == (B=5)**

**(4)   A += (X+5)**

**(5)   A != (C \*= (-D))**

**(6)   A \*= C+(X-D)**

**(7)   A %= D++**

**(8)   A %= ++D**

**(9)   (X++)\*(A+C)**

j = 0;

switch (i) {

case 3:

j++;

case 2:

j++;

case 1:

j++;

}

if (size = 0)

int i;

for (i = 0; i < 10; i++);

printf ("%d\n", i);

for (i < 10; i = 0; i++);

typecast et /

## Questions trous

Echange des variables

Trouver deux nombres entiers tels que la différence de leur carré donne juste 7 000.

Diviseurs de n

Conversion km – Mille

nb\_blancs = nb\_autres = 0;

for (i = 0; i < 10; )

nb\_chiffre[i++] = 0;

while ((c = getchar()) != EOF)

switch (c) {

case '0':

case '1':

case '2':

case '3':

case '4':

case '5':

case '6':

case '7':

case '8':

case '9':

nb\_chiffre[c - '0']++;

break;

case ' ':

case '\n':

case '\t':

nb\_blancs++;

break;

default:

nb\_autres++;

}

## Exercice de conversion :

Programme avec do…while Calculer la factorielle N! = 123...(N-1)N d'un entier naturel N en respectant que 0!=1.

~~a) Utilisez~~ **~~while~~**~~,~~

b) Utilisez **for**.

Calculez la somme des N premiers termes de la série harmonique :

1 + 1/2 + 1/3 + ... + 1/N

#include <stdio.h>

main(){

const int N=10;

int i;

int som;

float harmo;

harmo = 0.0; som = 0; i = 1; /\* initialisation des variables \*/

//Point d’observation 1

while (i<=N){

som = som + i;

harmo = harmo + 1/(float)i;

i++;

// Point d’observation 2

}

// Point d’observation 3

printf("La somme des %d premiers entiers est : %d\n", N, som);

printf("La somme des %d premiers termes de la"

" serie harmonique vaut : %f\n", N,harmo);

}

## Questions trace

Amortissement

#include <stdio.h>

/\*Programme de calcul d'échéancier simplifié:

A partir:

du taux d'intérêt mensuel

du taux des frais mensuels par rapport à la la somme empruntée

du montant des mensualités de remboursement

On considère que la mensualité d'un mois donné est composée

d'une part d'intérêts sur le mois écoulé, d'une part de frais

(assurance et divers frais) et d'une part d'amortissement.

\*/

main()

{ const float tauxfrais = 0.000725;

int i;

float S,Sact,Rembparmois,interet,capitalremb;

float fraisparmois,tauxmensuel,totalremb;

/\*REMARQUES\*/

/\*tauxfrais = taux des frais mensuels par rapport à  la somme empruntée\*/

/\*i = numéro de la mensualité en cours\*/

/\*S = somme empruntée\*/

/\*Sact = somme actuelle, restant à  rembourser\*/

/\*Rembparmois = montant des mensualités de remboursement\*/

/\*intérêt = montant des intérêts du mois en cours\*/

/\*capitalremb = montant de l'amortissement du mois en cours\*/

/\*fraispar mois = montant des frais mensuels = S \* tauxfrais\*/

/\*tauxmensuel = taux mensuel des intérêts (1% sera donné par 0.01)\*/

/\*totalremb = montant total des remboursements\*/

/\*les calculs ne sont pas arrondis à  2 décimales : le programme\*/

/\*pourrait facilement être amélioré en effectuant ces arrondis\*/

printf("Somme empruntée ? ");

scanf("%f",&S);

printf("Remboursements mensuels souhaités ? ");

scanf("%f",&Rembparmois);

printf("Taux mensuel de l'emprunt (0.01 par exemple) ?");

scanf("%f",&tauxmensuel);

printf("\n");

if (Rembparmois<=S\*(tauxmensuel+tauxfrais))

printf("IMPOSSIBLE : remboursement trop faible\n");

else

{ Sact=S; i=0;fraisparmois=S\*tauxfrais;

totalremb=0;

printf("durée Paiement capital remboursé intérêt");

printf(" frais capital restant\n");

do

{ i=i+1;

interet=Sact\*tauxmensuel;

if (Rembparmois > Sact+interet+fraisparmois)

{ Rembparmois=Sact+interet+fraisparmois;

capitalremb=Sact;

Sact=0;

}

else

{ capitalremb=Rembparmois-interet-fraisparmois;

/\* Sact=Sact-Rembparmois+interet+fraisparmois; soit:\*/

Sact -= capitalremb;

}

totalremb += Rembparmois;

printf("%4d %14.2f %14.2f %15.2f %8.2f %13.2f\n",

i,Rembparmois,capitalremb,interet,fraisparmois,Sact);

} while (Sact>0);

printf("\nle montant total des remboursements est : %14.2f\n",

totalremb);

}

}

Suite un

#include<stdio.h>

main(){

int p, s, i, t, n;

n=4;

p=0; // Valeur de u1

s=0; // Valeur de u2

for(i=3; i<=n; i++){

t = s;

s = 2\*p+s+1;

p = t;

// Point d’observation 1

}

printf("La valeur de u%i est : %i\n", i-1, s);

}

#include<stdio.h>

main(){

int n, B, c;

n=31; //n comme nombre

B=2; //B comme base

//Point d’observation 1

while(n!=0){

c=n%B; //c comme chiffre

n=n/10; //Point d’observation 2

printf("%i", c);

}

printf("\n");

}

Afficher les nombres premiers entre deux numéros entiers x et y fournis par l’utilisateur.

## Questions Algorithme

Ecrire un programme qui calcule le bonus / malus d’un conducteur pour son assurance.

program bonus\_malus;

const CBMMIN=0.5;

CBMMAX=3.5;

COEFFMAX=1.25;

COEFFMOY=1.125;

COEFFMIN=0.95;

var CBM : real;

nbaccidents : integer;

n : integer; {compteur de 1 … n accidents}

typeresp : char;

begin

write('Donnez le CBM du d‚but de l''ann‚e : ');

readln(CBM);

write('Combien d''accidents dans l''ann‚e ? ');

readln(nbaccidents);

n:=0;

while n<nbaccidents do

begin

write('accident no ',n,': responsabilit‚ EntiŠre ou Partielle (E/P) ? ');

readln(typeresp);

if typeresp in ['P','p']

then CBM := trunc(CBM\*100\*COEFFMOY)/100

else CBM := trunc(CBM\*100\*COEFFMAX)/100;

if CBM>CBMMAX then CBM:=CBMMAX;

n:=n+1

end;

if nbaccidents=0 then

begin

CBM := trunc(CBM\*100\*COEFFMIN)/100;

if CBM<CBMMIN then CBM := CBMMIN

end;

writeln('Le nouveau CBM est : ',CBM:5:2)

end.

1. Un nombre est parfait quand il est égal à la somme de ses diviseurs.

exemple : 6 = 1 + 2 + 3 est un nombre parfait.

28 = 1 + 2 + 4 + 7 + 14 est aussi un nombre parfait.

Ecrire un programme qui lit un nombre entier n au clavier, puis qui décide et affiche s'il est parfait ou non.

Aide : dans un premier temps, il s'agit de faire la somme de tous les diviseurs de n, c'est à dire de tous les nombres compris entre 1 et n-1 qui divisent n. Ensuite il suffit de comparer cette somme avec n.