

Healer

GNG1506 Automne 2019 – Devoir 1

Barème (total de 35 points)

1. Question 1 : 10 points
2. Question 2 : 10 points
3. Question 3 : 15 points

Question 1 (10 points)

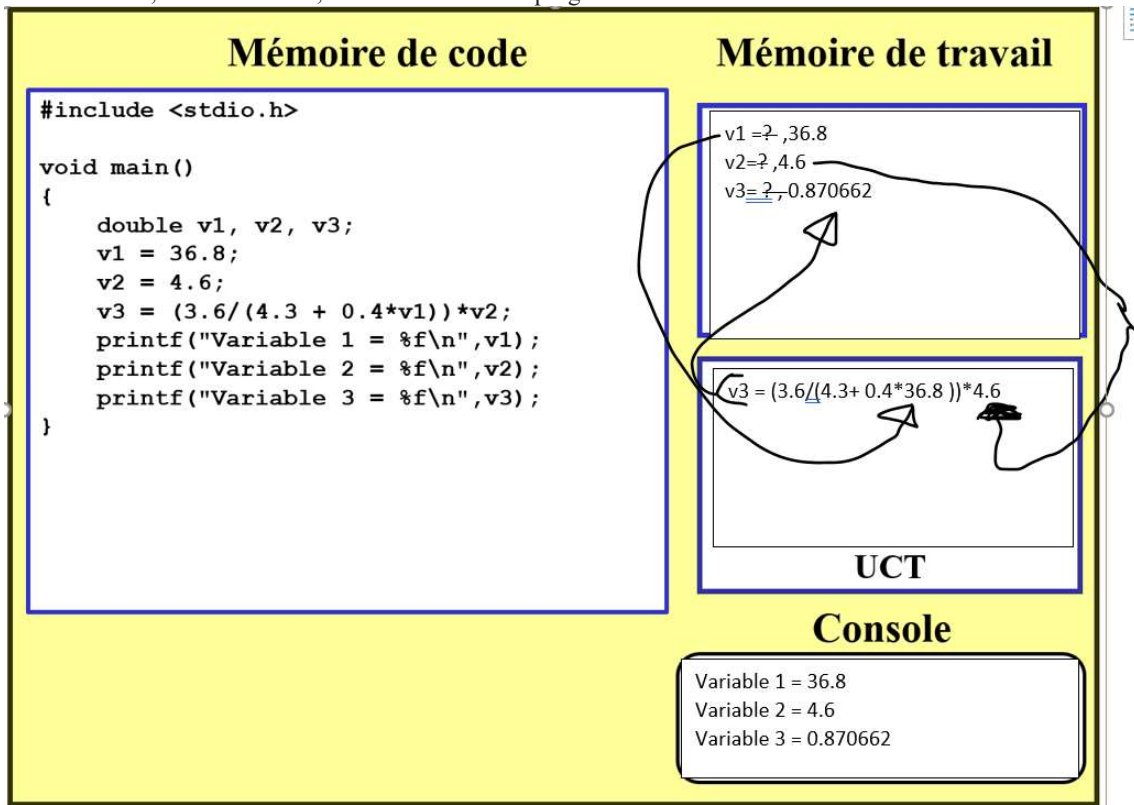
a) (5 points) Le modèle de programmation ci-dessous contient le programme C montré dans la mémoire de code. Vous devez montrer comment la mémoire de travail est utilisée durant l'exécution de ce programme et comment l'UCT évalue l'expression arithmétique qui donne une valeur à la variable v3.

Montrez les variables créées dans la mémoire de travail et comment leurs valeurs changent durant l'exécution du programme.

- Montrez comment les valeurs sont affectées aux variables. Soyez certain de montrer toutes les valeurs qui sont affectées et remplacées. Représenter les affectations successives des valeurs comme suit :

Nom de variable ~~z~~, ~~z~~, ~~z~~, ~~z~~, 10

- Pour l'opération arithmétique, montrez comment le contenu de la mémoire de travail est déplacé vers l'UCT pour calculer la valeur affectée à la variable z. Montrez **toutes** les opérations effectuées dans l'UCT, c.-à-d. **une ligne** par opération.
- Enfin, dans la console, montrez la sortie du programme.



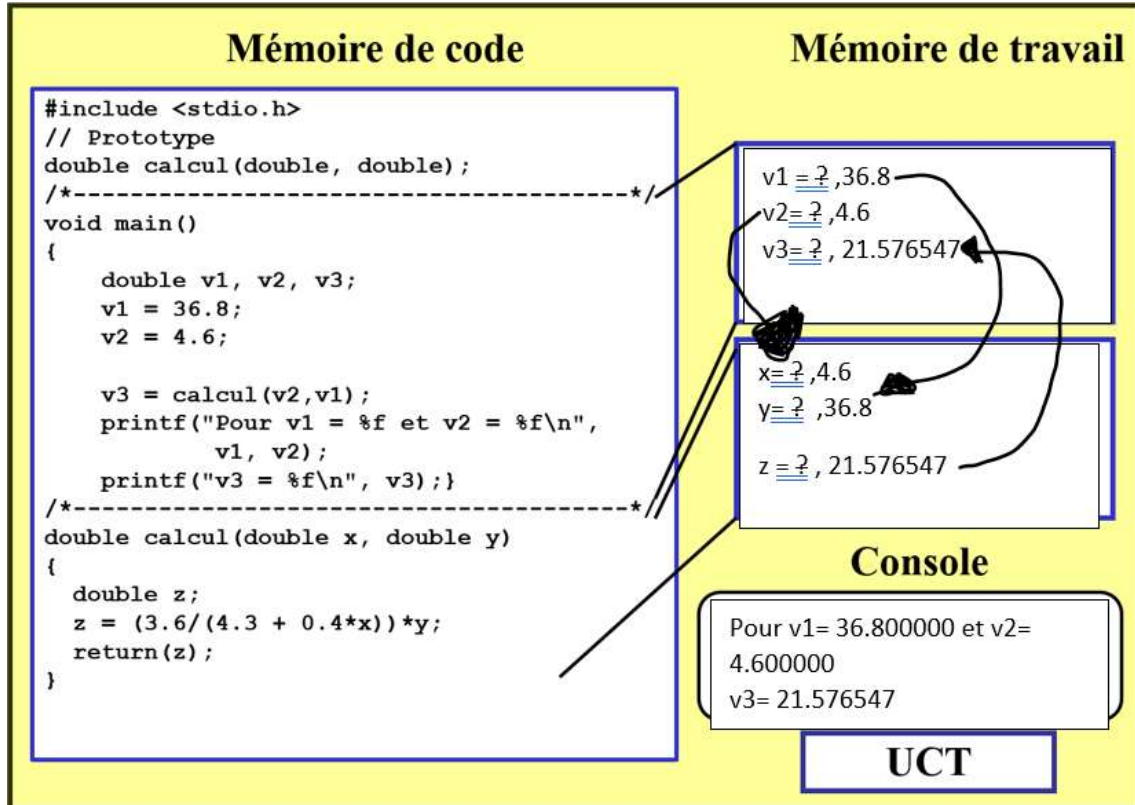
b) (5 points) Le modèle de programmation ci-dessous contient dans la mémoire de code le programme C avec 2 fonctions. Vous devez montrer comment la mémoire de travail est utilisée durant l'exécution des deux fonctions de ce programme. Chaque portion de mémoire de travail est associée à une fonction avec une paire de lignes. (Note : la première paire de lignes indique la portion de mémoire de travail alloué à la fonction **main** et la deuxième paire indiquent celle allouée à la fonction **calcul**).

Montrez les variables (et paramètres) créées dans chaque parcelle de mémoire durant l'exécution des fonctions. Il n'est **pas** nécessaire de montrer comment l'UCT évalue les expressions dans les différentes instructions de ce programme comme vous l'avez fait dans la partie (a).

- Montrez comment les valeurs sont assignées aux variables. Soyez certain de montrer toutes les valeurs qui sont assignées et remplacées. Représenter les affectations successives des valeurs comme suit :

Nom de variable ~~7~~ ~~2~~, ~~6~~, ~~4~~, 10

- Utilisez des flèches pour montrer comment les valeurs sont copiées entre la mémoire de travail allouée à la fonction **main** et la mémoire de travail allouée à la fonction **calcul**.
- Enfin, dans la console, montrez la sortie du programme.



Question 2 (10 points)

Vous êtes impliqués dans le développement d'un engin d'avion, la soufflante non carénée (voir https://fr.wikipedia.org/wiki/Soufflante_non-carénée). Un avion de masse 20 000 kg atteint une vitesse de 180 m/s dont la puissance de l'engin applique une force de 40 000 newtons. Lorsque le pilote augmente cette puissance à 60 000 newtons et l'avion accélère, l'équation suivante donne le changement de la vitesse après ce changement de puissance (une nouvelle vitesse sera atteinte après environ 120 s)

$$v = 0.00001t^3 - 0.00488t^2 + 0.75795t + 181.3566$$

où v est la vitesse de l'avion en m/s et t est le temps en second ($t = 0$ au moment où la puissance de l'engin est augmentée à 60 000 newtons).

Développez un programme qui calcule la vitesse de l'avion à un temps donné t . L'utilisateur devra fournir la valeur du temps. Testez votre programme en utilisant les cas indiqués dans le tableau suivant :

Temps t (s)	Vitesse v (m/s)
0.00	181.35660
2.00	182.85306
10.00	188.45810
50.00	208.30410
100.00	218.35160
120.00	219.31860

Lignes directrices:

Logique/Stratégies :

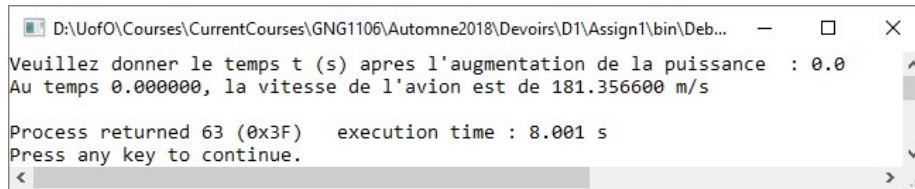
- Dans la fonction **main**,
 - Penser à utiliser les variables locales suivantes :
 - v : pour la vitesse de l'avion.
 - t : pour le temps t .
 - Demandez à l'utilisateur de donner une valeur du temps t . Utilisez des messages courts et appropriés pour faire la demande à l'utilisateur. Et enfin, lisez les valeurs à partir du clavier et affecter la valeur à la variable t .
 - Faites un appel à une fonction, par exemple **calculeVitesse**, pour calculer la vitesse de l'avion. Stockez le résultat retourné par la fonction **calculeVitesse** dans la variable t . Vous aurez besoin de définir au préalable cette fonction dans votre programme.
 - Affichez les résultats de la façon suivante :
 - Au temps 0.0, la vitesse de l'avion est de 181.3566 m/s.
- Pour la fonction **calculeVitesse**,
 - Pensez à utiliser le paramètre suivant :
 - t : pour la valeur du temps.
 - Pensez à utiliser la variable locale suivante :
 - **vitesse** : pour stocker la vitesse de l'avion au temps t (noter que cette variable contiendra la valeur retournée par la fonction).
 - La fonction calcule la vitesse de l'avion (voir l'équation donnée au début de la question), la stocke dans la variable **vitesse** et retourne la valeur stockée dans **vitesse**.

Testez le programme en utilisant les valeurs fournies dans le tableau présenté plus haut. Dans votre fichier de devoir, présenter la sortie de votre programme pour **tous** les cas tests.

Healer

La réponse à cette question doit fournir :

- 1) Le code source de votre programme (insérer **également** le code source dans le fichier de devoir).
- 2) Les sorties montrant les résultats de tous les cas de test; insérer la sortie dans le fichier de devoir. Ce qui suit est un exemple de la sortie pour le premier cas de test.



```
D:\UofO\Courses\CurrentCourses\GNG1106\Automne2018\Devoirs\D1\Assign1\bin\Deb...
Veuillez donner le temps t (s) apres l'augmentation de la puissance : 0.0
Au temps 0.000000, la vitesse de l'avion est de 181.356600 m/s

Process returned 63 (0x3F) execution time : 8.001 s
Press any key to continue.
```

1) Code source

```
#include <stdio.h>

#include <stdlib.h>
#include <math.h>

double calcul_vitesse(int x);

void main()
{
    int t;
    double vitesse;
    t=0;
    vitesse= 180;
    printf("temps inicial a %d, et vitesse
inicial %f. ",t,vitesse );
    printf("Suite a un passage de puissance de 40
000 N a 60 000N du moteur,\n\n Veuillez indiquez
le temps ecouler apres ce changement de
puissance:\n");
    printf("temps=");
    scanf("%d",&t);

    vitesse=calcul_vitesse(t);

    printf("\nAu temps %d , la vitesse est:
%f\n",t,vitesse);
}

double calcul_vitesse(int x)
{
    double v;
    v=0.00001*pow(x,3)-0.00488*pow(x,2)+(0.75795*x)+
181.3566;
    return (v);
}
```

2) Consoles

Cas 1

temps inicial a 0, et vitesse inicial 180.000000. Suite a un passage de puissance de 40 000 N a 60 000N du moteur,

Veillez indiquer le temps ecouler apres ce changement de puissance:
temps=0

\u temps 0 , la vitesse est: 181.356600

rocess returned 41 (0x29) execution time : 2.568 s
ress any key to continue.

Question 3 (15 points)

La résistance électrique d'un matériel dépend de la température. Le coefficient de température de résistance, α , permet d'ajuster la résistance d'un matériel pour une température donnée. En plus du coefficient, la résistance du matériel doit être connue à une température standard (normalement 20 degrés Celsius). La résistance, R , d'un matériel à une température T , se trouve avec

$$R = R_{ref} [1 + \alpha(T - T_{ref})] \quad (1)$$

où

R = la résistance du matériel en ohms à la température T en °C,

R_{ref} = la résistance du matériel à la température T_{ref} (normalement à 20 °C),

α = coefficient de température de résistance (/°C) pour le matériel,

T = la température du conducteur en degré Celsius,

T_{ref} = température de référence à laquelle α est donnée pour le matériel,

(référence: <https://www.allaboutcircuits.com/textbook/direct-current/chpt-12/temperature-coefficient-resistance/>)

Pour les métaux, le coefficient est positif, ce qui indique que la résistance augmente à mesure que la température augmente. Dans le cas des éléments carbone, germanium, et le silicium, le coefficient est négatif, ce qui indique que la résistance diminue à mesure que la température augmente. Pour certains alliages métalliques, le coefficient est presque zéro et donc leur résistance varie très peu avec le changement de température. La table suivante donne des exemples de valeurs pour le coefficient de température de résistance.

Matériel	Élément/Alliage	α (/°C)
Nickel	Élément	0.005866
Fer	Élément	0.005671
Tungstène	Élément	0.004403
Aluminium	Élément	0.004308
Cuivre	Élément	0.004041
Acier (99.5% fer, 0.5% carbone)	Alliage	0.003
Constantan	Alliage	-0.000074
Silicium	Élément	-0.075
Germanium	Élément	-0.048

Vous êtes impliqué dans un projet de développement de transformateurs électriques qui contient des bobines de champ. Une bobine est formée d'un conducteur enroulé autour d'un noyau de fer. Votre rôle est de développer un logiciel qui permet de calculer la résistance du conducteur de la bobine à différentes températures.

La résistance est calculée avec les étapes suivantes :

Healer

1. Avec la valeur de résistance par unité de longueur, R_L (ohms/m) et la longueur du conducteur, L (m), de la bobine calcule la résistance à la température de référence, T_{ref} , (20 °C) comme étant $R_{ref} = R_L L$.

2. Pour ajuster la résistance à une température donnée, T (°C), l'équation (1) est utilisée étant donné le coefficient de température de résistance, α (/°C), à la température de référence, T_{ref} (20 °C).

Suivez les consignes suivantes pour répondre à cette question :

- 1) Tout d'abord, élaborer un ensemble de cas de test (Excel est un logiciel pratique qui vous permet de créer des cas de test). Au minimum, présentez 5 cas tests. Assurez-vous que les cas de test couvrent des larges gammes des données :
 - a) Variez la résistance par unité de longueur, R_L , entre 0.00327 ohm/m et 3.55 ohms/m.
 - b) Utilisez les valeurs du coefficient α d'aluminium et de cuivre données dans le tableau ci-dessus.
 - c) Variez la longueur du fil conducteur entre 0.01 mètre et 100 mètres.
 - d) Variez les températures du conducteur entre -40 et +40 °C.
- 2) Développer votre programme en utilisant le gabarit C GNG1506 (GNG1506gabarit.c), c'est-à-dire, votre programme doit contenir une fonction `main` et une fonction qui calcule la résistance du conducteur de la bobine. La fonction `main` contient des instructions pour obtenir les données de l'utilisateur, appelle la fonction pour obtenir la résistance, et affiche les résultats à l'écran (afficher aussi les valeurs d'entrées de l'utilisateur). Documentez bien votre programme et suivez les conventions de programmation
- 3) Un indice : Vous pouvez représenter 5.65×10^{-8} dans un programme C par « 5.65e-8 ».
- 4) Inclure dans votre fichier de devoir, un tableau contenant tous vos cas de test, votre code source, et les sorties montrant les résultats de tous les cas de test. Soumettez également votre fichier de code source.

TABLEAU de CAS de TEST :

Cas de Test	Résistance par unité de longueur, R_L (0.00327 - 3.55 ohms/m)	Coefficient α	Longueur du fil conducteur L (0.01 - 100 mètres)	Température T du conducteur [-40 ; + 40 °C].	Resultat- Sortie La valeur de la résistance
Aluminium					
1	0,0032700	4,308E-03	0,0100	-40	0,0000242
2	0,8899525	4,308E-03	25,0075	-20	18,4204216
3	1,7766350	4,308E-03	50,0050	0	81,1861242
Cuivre					
4	2,6633175	4,041E-03	75,0025	20	199,7554708
5	3,5500000	4,041E-03	100,0000	40	383,6911000

Healer

CODE SOURCE :

```
/*-----*/
Fichier: D1_Question3
Auteur: Healer
Description: Ce programme calcule la resistance d'un conducteur pour une temperature donnee
selon le materiel et les caracteristique du conducteur
-----*/
#include <stdio.h>
#include <math.h>

// Prototypes de fonctions
double Calcul_Resistance(double w,double x,double y, double z);
/*-----*/
Fonction: main
Description: Demande les donnee du conducteur: Sa longueur, sa resistance par unite de
longueur, le coeficient de resistance de son materiel.
Demande la temperature a laquelle on veut connaitre la resistance du conducteur.
Appelle la fonction pour calculer la resistance: Calcul_Resistance.
Affiche le resultat soit la resistance recherchee
-----*/
void main(void)
{

// Déclarations des variables
double L; // Longueur du conducteur (m)
double R_L; //resistance par unite de longueur pour le conducteur(Ohm/m)
double R; // resistance qu'on veut connaitre, a la temperature T (C)
double a; // le coeficient de temperature de resistance (/c) pour le materielle
double T; // la temperature (C) ou l'etude est effectue
char element[256];

//demande des variables:

printf("Temperature (C) du milieu T = ");
scanf("%lf",&T);
fflush(stdin);
printf("%f\n",T);

printf("Longueur(m) du conducteur L=");
scanf("%lf",&L);
printf("Longueur(m) L= %f\n",L);
fflush(stdin);

printf("Resistance par unite de longueur pour le materiel (Ohm/m)\n R_L=" );
scanf("%lf",&R_L);
printf("%f\n",R_L);
fflush(stdin);

printf ("\n\nCoeficient de resistance du Cuivre (/C) = 4.041e-3 \nCoeficient de resistance
de l'Aluminium (/C) = 4.308e-3\n\n");
```

Healer

```
printf("Entrer le coefficient correspondant au materiau d'interet:\nCopier et coller parmi
ceux ci-dessus:\n");
scanf("%lf",&a);
fflush(stdin);
printf("%f\n",a);

// Calculer la resistance
R= Calcul_Resistance(L,R_L,a,T);
//Impression du resultat
printf("\n\nLa resistance de ce conducteur a T=%f C est %f Ohm\n",T,R);
}
/*-----*/
Fonction: Calcul_Resistance
Paramètres:
w - est la longueur du conducteur (m)
x - est la resistance par unite de longueurdu materiel (Ohm/m)
y - est le coefficient de temperature de resistance du materiel (/C)
z - est la temperature a laquelle on veut connaitre la resistance du conducteur (C)

Retourne: C'est la valeur de la resistance a la temperature donnee
Description: C'est la fonction qui peut calculer la resistance a partir des variables de la
longueur, la resistance par unite de longueur du conducteur.
Aussi du coefficient de temperature de resistance ainsi que la temperature d'etude.
Bien note que R_ref est la resistance de reference. On la calcule à partir du produit
de la resistance par unite de longueur et la longueur du conducteur

-----*/
double Calcul_Resistance(double w,double x,double y, double z)
{
// Déclarations des variables

double R_ref;
double R;
// Instructions
R_ref= x*w;

R= (double)R_ref*(1+y*(z-20));
return(R);
}
```

Healer

Sorties des CAS de TEST :

1.