

Licence 2, Partiel LC4, Développement en C
15 mars 2008, durée 2h

Le sujet comporte 4 pages. Documents manuscrits et tous les documents de cours sont autorisés (tds, tps, corrections, cours). Les livres ne sont pas autorisés. Tous les exercices sont indépendants. Il n'est pas demandé de faire #include pour les fichiers en-tête de la bibliothèque standard.

Votre code doit être écrit de façon lisible, avec des indentations et des accolades appropriées mettant de distinguer les fins de blocs de code (fins de boucles, ...).
Le barème est donné seulement à titre indicatif.

Exercice 1

Il affichera le programme suivant:

```
#include <stdio.h>
#include <stdlib.h>
int main(void){
    double tab[] = { 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0 };
    double *d = tab;
    double *f = d + sizeof tab / sizeof(double);
    int i = 4;
    while (*p = &tab[i];
    double x;
    printf("%3.1f\n", *(d+2));
    printf("%3.1f\n", *d + 2);
    printf("%d\n", p-d);
    printf("%3.1f\n", *(p-2)+3);
    f = *p++;
    printf("f-p = %d\n", f-p);
    printf("x = %3.1f, *p= %3.1f\n", x, *p);
    return EXIT_SUCCESS;
```

Exercice 2

Écrire une fonction

```
char * coder(const char *text, const char *from, const char *to)
```

La fonction retournera une nouvelle chaîne de caractères de même longueur que la chaîne text (il faut allouer la mémoire pour la nouvelle chaîne). Les chaînes from et to donnent une correspondance de lettres dans le codage. Par exemple from="abcd" et to="xyzx".

→ x,
→ y,
→ x,

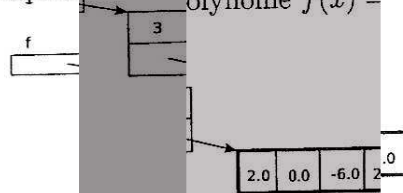
Il s'agit de dire la première lettre de from correspond à la première lettre de to, la deuxième lettre de from correspond à la deuxième lettre de to, etc.

Noter toutefois que seule la première occurrence d'une lettre dans from compte, par exemple la deuxième occurrence de a dans from ne donne pas lieu à une correspondance.

s from
 ettre d dan e de lan'a pas de corre spondance p parce que la chaîne to
 a construit qui ser, façon suivante: chaque caractère qui n'a pas de
 opié sans ch acés res, si un caractère dans from a une correspondance
 caractère codé dification recopié dans la chaîne codée. Dans notre exemple
 ont rempl respectivement par x,y,x tandi s que tous les autres
 és sans mo on.

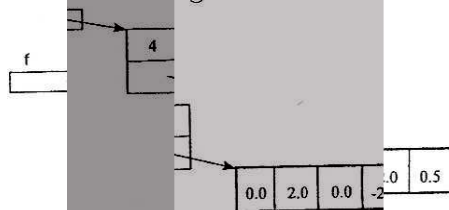
$= a_0 + a_1$
 ôme $p(x) = x + a_2x^2 + \dots + a_nx^n$ par le type polynome défini

; e;
 *polynom une str
 est un po
 teur vers ent du taicture dont le premier chan ap donne le degré du
 e champ ente le pointeur vers le tableau de coe efficients, le coefficient
 ème élém tableau.
 e 1 représ polynôme $f(x) = 2.0 - 6.0x^2 + 2.0x^3$.



Polynôme
 igure 1: F $f(x) = 2.0 - 6.0x^2 + 2.0x^3$.
 ynome i

polie de l'integral(polynome p)
 $+ \frac{a_1}{2}x^2 -$
 e polynôm $\frac{a_{i-1}}{i}$ potégrale de $p(x)$. Pour $p(x) = a_0 + a_1x + a_2x^2 + \dots +$
 $(x) = a_0x$ e de la $+ \frac{a_2}{3}x^3 + \dots + \frac{a_{n+1}}{n+1}x^{n+1}$, c'est -à-dire le coefficient de
 de x^i est ar $i > 0$. on doit construire le polynôme du
 e polynôm figure 1 la foncti
 ous:



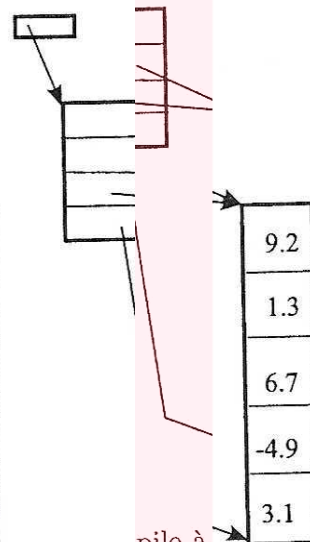
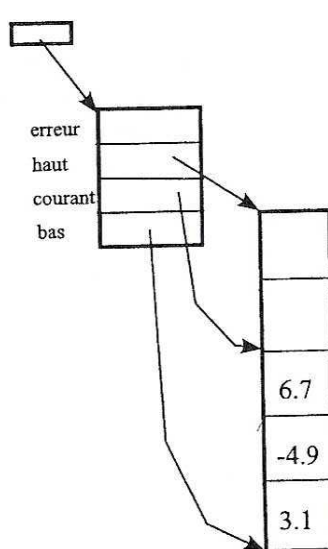
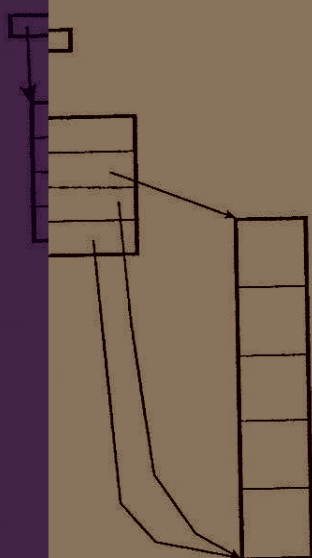
a représe
 oubles ser entée par le type suivant:

t e
 b {
 int
 erreur;
 double
 le *haut;
 double
 le *courant;
 lib
 a *bas;
 double
 def struct ep *pile;

implantée par un tableau de nombres doubles géré avec des
l'adresse du premier élément tableau, le pointeur haut donne
l'adresse du premier élément du tableau, courant donne l'adresse du premier
le champ erreur sera utilisé pour signaler des erreurs lors de
etc.

pointe
e l'adr
élémen
s opér

urs: le pointeur
esse juste après
t libre dans le
ations empiler,



pile à
7 au s
nt dan

Figure 2: Le dessin représente trois états d'une pile de capacité 5, la pile au milieu contient 3 éléments, 3.1 est au fond de la pile et 6.7 est au-dessus. La pile de gauche contient déjà 5 éléments, elle est pleine. Pour ajouter un élément, il faut agrandir le tableau utilisé pour stocker les éléments.

gauche est vide,
ommet, la pile à
s la pile à droite

Les questions qui suivent implémentent les opérations suivantes. Ces questions forment un ensemble, elles peuvent être traitées dans quel ordre. En particulier, si vous êtes bloqué sur une question, ne passez pas à la question suivante. De plus, il est possible de proposer que les fonctions demandées dans les questions ne soient disponibles même si vous ne les avez pas implantées.

te pile. Même
traitées dans
une question
ne question on
écédentes sont

Question 1: Écrire une fonction

```

* pile faire_pile(int capacite)

```

construit une pile vide qui peut stocker jusqu'à capacite d'éléme

Question 2: Écrire une fonction

```
int pile_vide(pile p)
```

retourne 1 si la pile est vide et 0 sinon.

Question 3: Écrire une fonction

```
double depiler(pile p)
```

qui dépile et retourne l'élément au sommet de pile. Si la pile est vide, la fonction retourne 0 et met 1 dans le champ erreur. Si la pile était non-vide, la fonction met la valeur dépilee dans le champ erreur.

Question 4: Écrire une fonction

```
int empiler(pile p, double d)
```

qui met d au sommet de la pile.

Si la pile était pleine avant de mettre d sur la pile, il faut doubler la capacité de la pile (realloc).

La fonction met la valeur 0 dans le champ erreur sauf si elle a tenté d'agrandir la pile et échoué, auquel cas il faut mettre 2 dans le champ erreur.

La fonction retourne la valeur mise dans erreur.

Question 5: Écrire une fonction

```
int operation(pile p, char op)
```

Le paramètre op peut prendre une des quatre valeurs: '+', '-', '*', '/'. La fonction effectue l'opération indiquée sur les deux nombres doubles du sommet de la pile, effectue la dépile et retourne le résultat. Si le paramètre op n'est pas correct, la fonction met 5 dans le champ erreur. Si la pile contenait moins de 2 éléments et que l'opération ne peut pas être effectuée, la fonction met 6 dans le champ erreur. Si la pile contenait moins de 2 éléments et que l'opération ne peut pas être effectuée, la fonction met 6 dans le champ erreur.

Si la pile contenait moins de 2 éléments et que l'opération ne peut pas être effectuée, la fonction met 6 dans le champ erreur. Si la pile contenait moins de 2 éléments et que l'opération ne peut pas être effectuée, la fonction met 6 dans le champ erreur.

La fonction operation retourne la valeur mise dans erreur.

Question 6: Écrire une fonction

```
void detruire(pile p)
```

qui libère la mémoire occupée par la pile p (tableau et structure).