

M1103 : Structures de Données & Algorithmes fondamentaux

Feuille TP n° 4 – distribuée en semaine DUT n°14

Utilisation des Piles

OBJECTIFS PEDAGOGIQUES :

1.- Apprendre les manipulations de base d'un Type abstrait de Données Pile

ÉVALUATION :

- Les exercices abordés dans cette feuille TP feront l'objet d'une évaluation en fin de module

RESSOURCES A VOTRE DISPOSITION POUR REALISER CE TP :

- Sur le WebCampus, dans la zone associée à ce module APL-M1103 :
 - tp4.pdf : le présent sujet de tp
 - dans l'archive ressourcesTP4.zip :
 - pile.h, pile.cpp : les fichiers implémentant le type UnePile.
 - sousProgrammes.h, sousProgrammes.cpp, main.cpp : les fichiers associés à l'exercice n°1 (expressions entières)
 - sousProgrammes.h, sousProgrammes.cpp, main.cpp : les fichiers associés à l'exercice n°2 (tours de Hanoï)

DIRECTIVES GENERALES – ORGANISATION DES REPERTOIRES DE TP

Néant

DIRECTIVES PARTICULIERES POUR CETTE SEANCE DE TP

- Créer un répertoire tp4.
- À l'intérieur de ce répertoire, créer un projet pour chaque exercice de la feuille de TD n°4 (projets Hanoi et projet ExpressionsEntieres).
- Copier les fichiers pile.h, pile.cpp dans le répertoire tp4.
- Copier les fichiers main.cpp, sousProgrammes.h et sousProgrammes.cpp correspondants à chaque exercice dans le répertoire adéquat.

TRAVAIL A FAIRE SUR CETTE FEUILLE DE TP

Tours de Hanoï.

- (a) Compléter les fichiers sousProgrammes.h et sousProgrammes.cpp fournis afin de répondre aux questions 1 à 6 de l'exercice n°2 de la feuille de TD n°4. Ne pas modifier le fichier main.cpp (excepté pour éventuellement ajuster les #include).

Expressions entières en notation postfixée.

- (b) Compléter les fichiers sousProgrammes.h et sousProgrammes.cpp fournis afin de répondre aux questions 1 à 7 de l'exercice n°1 la feuille de TD n°4. Ne pas modifier le fichier main.cpp (excepté pour éventuellement ajuster les #include).
- (c) À l'aide de votre éditeur de texte préféré, produire un jeu d'essai que vous jugerez suffisamment significatif pour tester le sous-programme demandé. Il se présentera sous forme tabulaire, par exemple :

Jeu d'essai et exécution des tests :

Expression testée	Résultat attendu	ok	Résultat obtenu - version 1	Analyse problème rencontré → Modification à apporter
af*				
ac+b-				
...				

Exécuter les tests et noter les résultats obtenus avec, par exemple, les valeurs suivantes pour les variables a..z :

Valeurs des variables :

a	b	c	d	e	f	g	h	i	...	z
10	-2	3	4	5	-1	0	0	0	...	0

Étapes du test

Avant de tester votre programme,

1. Remplissez la colonne 'Expression testée', pour lister un nombre significatif d'expressions à tester.
2. Complétez la colonne 'Résultat attendu' pour chacune des expressions (lignes) à tester
3. Lancez le programme et testez-le pour chacune des expressions identifiées. Pour chaque expression, notez la valeur trouvée (colonne 'Résultat obtenu - version 1'), si cette valeur correspond ou pas à la valeur attendue ('colonne ok'). Si oui, vous passez à l'expression suivante. Sinon, vous notez dans la colonne 'Remarque', si vous en avez une idée, ce qui peut avoir causé l'erreur et une idée de modification.

Une fois les toutes expressions testées, vous corrigez le code selon es indications / directives marquées dans la colonne 'Remarques'.

Vous recommencez les tests en produisant un nouveau tableau et en le remplissant selon les modalités indiquées.

Testez à nouveau toutes les expressions, pour vous assurer que les corrections apportées au programme n'ont pas altérer son comportement sur des cas qui étaient corrects...

Cette opération sera répétées jusqu'à ce que tous les tests se terminent avec OK.

Vous demandez à votre enseignant ce que vous ne comprenez pas.