

Corrigé type de l'examen de rattrapage: Algorithmique et structures de données 2

Exercice 1 (5.5 pts):

1) La procédure **calcul** (2.5 points)

0.25	Procedure calcul (Var T:Tab;Var nbP,nbImp:integer);
0.25	Var i:integer;
	Begin
0.5	nbP ← 0;nbImp ← 0;
0.5	For i ← 0 to n-1 Do
0.5	If T[i] mod 2 = 0 then nbP ← nbP+1
0.5	Else nbImp ← nbImp+1;
	End;

2) Qu'affiche chacune de ces procédures? (3 points)

- La procédure **aff1** affiche: 5, 4, 3, 2, 1
- La procédure **aff2**: erreur , récursivité infinie
- La procédure **aff3** affiche: 3, 4, 5

Exercice 2 (7 pts):

1) Fonction **moyenne** (3.5 pts)

0.25	Function moyenne(L:List) :real;
0.5	Var p:list;s,nb:integer;moy:real;
	Begin
0.5	s ← 0;nb ← 0;
0.25	p ← L;
0.5	While p≠Nil do
0.25	Begin
0.25	s ← s+p^.val;
0.25	nb ← nb+1;
0.25	p ← p^.suiv;
	End;
0.25	moy ← s/nb;
0.25	moyenne ← moy;
	End;

2) Procédure **binariser** (3.5 pts)

0.25	Procedure binariser(Var L:List);
0.5	Var p:list;moy:real;
	Begin
0.5	moy $\leftarrow$ moyenne(L);
0.25	p $\leftarrow$ L;
0.5	While p $\neq$ Nil do
0.25	Begin
0.5	If p^.val < moy then p^.val $\leftarrow$ 0
0.5	Else p^.val $\leftarrow$ 1;
0.25	p $\leftarrow$ p^.next;
	End;
	End;

Exercice 3 (7.5 pts):

1) La procédure **éclater**: (3 pts)

0.25	Procedure eclater(Var F,F1,F2:Queue);
0.25	Var x:integer;
	Begin
0.25	InitializeQueue(F1);
0.25	InitializeQueue(F2);
0.5	While isQueueEmpty(F) $\neq$ False Do
0.25	Begin
0.25	Dequeue(x,F);
0.5	If x mod 2=0 then Enqueue(x,F1)
0.5	Else Enqueue(x,F2);
	End;
	End;

2) Ecrire la procédure **réorganiser**: (4.5 pts)

0.25	Procédure réorganiser(Var F:Queue);
0.75	Var x:integer;F1:Queue;P:Stack;
	Begin
0.25	InitializeQueue(F1);
0.25	InitializeStack(P);
0.5	While isEmpty(F)=False do
0.25	Begin
0.25	Dequeue(x,F);
0.5	If x>0 then Enqueue(x,F1)
0.5	Else Push(x,P);
	End;
0.5	While isEmpty(F1)=False do Begin Dequeue(x,F1); Enqueue(x,F) End;
0.5	While isEmpty(P)=False do Begin Pop(x,P); Enqueue(x,F) End;
	End;