

## EXERCICE 1 : ( 6 points)

On se donne le programme Pascal suivant :

```
program exercice1; uses wincrt;
var .....;
procedure afficher (.....);
var .....;
function OK (.....) : boolean;
  var .....;
  .....;
begin
  i:=2; V:=true;
  while V and (i<e) do
    if e mod i=0 then V:=false else i:=i+1;
    .....;
end;
begin
  for i:=x to y do
    if OK( i ) then write( i );
end;
begin
  repeat readln(a,b); until (a in [1..100]) and (b in [1..a]);
  afficher(a,b);
end.
```

### Travail demandé :

Compléter les pointillés par les déclarations adéquates :

- 1) Encercler, dans la liste suivante, les valeurs pour lesquelles la fonction **OK** renvoie la valeur vrai :

5      1.5      25      13      21      2      9      19      29      27

- 2) Que fait ce programme ?

.....

## Exercice 2 (3 pts)

La fonction 'recherche' donnée ci-après teste si une valeur  $x$  existe ou pas dans un intervalle  $bi..bs$  et ce en utilisant la méthode de la recherche dichotomique.

```
0) Def FN recherche(x:entier):booléen
1) v←faux
   bi←1
   bs←1000
2) répéter
   m← (bi+bs) div 2
   si m=x alors v←vrai
   sinon si x<m alors bs←m-1
   sinon bi←m+1
   jusqu'à (v=vrai) ou (bi>bs)
3) recherche←v
4) fin recherche
```

Un jeu consiste à deviner l'entier  $x$  qui sera généré au hasard par l'ordinateur ( $1 \leq x \leq 1000$ ). Le joueur propose la valeur  $m$ , si cette valeur est égale à  $x$ , le programme lui affiche « BRAVO ! VOUS AVEZ GAGNE » sinon il lui demande de réessayer en choisissant une autre valeur de  $m$  dans le nouvel intervalle  $bi..bs$  calculé et affiché au fur et à mesure du jeu.

Le nombre d'essai ne doit pas dépasser 5. Dans le cas contraire, le programme affiche « DESOLE ! CA SERA POUR UNE AUTRE FOIS ».

### **Travail demandé:**

Modifier la fonction 'recherche' en une procédure qui permet de simuler le jeu ci-dessus (la solution doit être donnée sous forme d'un algorithme).

### **Problème :**

On se propose de remplir un tableau **TC** avec **N** chaînes de caractères ( $3 \leq N \leq N_{max}$ ) avec  $N_{max} = 20$ , puis de remplir un tableau **TN** à partir de **TC** tel que **TN [i]** contient la somme des chiffres plus le nombre des lettres de **TC[i]**, et enfin d'afficher les éléments de **TN** sans redondance.

**NB :** Les chaînes de caractères contiennent uniquement des lettres alphabétiques et/ou des chiffres.

### **Exemple :**

Pour  $n = 5$  et le tableau **TC** suivant :

	1	2	3	4	5
TC	K23CA3	LBN20	SADT250	KIB1B	119HJU

Pour  $i = 1$ , La somme des chiffres :  $2+3+3 = 8$ , Le nombre des lettres = 3 alors  $TN[1]=8+3= 11$

Le tableau TN formé

<b>TN</b>	1	2	3	4	5
	11	5	11	5	14

Le programme affichera : **11 5 14**

**Questions :**

- 1) Analyser le problème en le décomposant en modules
- 2) Analyser chaque module