

## Partie théorique – Corrigé

### Question 1

- a) voir "Algorithmes obligatoires"
- b) La liste doit être triée.
- c) 

```
rechDichoR(lb, 'mv', 0, 7)
= rechDichoR(lb, 'mv', 4, 7)
= rechDichoR(lb, 'mv', 5, 7)
= rechDichoR(lb, 'mv', 5, 5)
= 5
```

### Question 2

- a)+b) voir "Algorithmes obligatoires"
- c) 

```
function horner_rec(var a:poly;x:extended;i:integer):extended;
//La fonction doit être appelée avec i=0
begin
  if i=a.d+1 then result:=0
  else result:=a.c[i]+horner_rec(a,x,i+1)*x
end;
```
- d) \_\_\_\_\_

### Question 3

- a) Après l'appel `p_rec(lb, 'ab+c+def-gh+ij', '+')` la liste `lb` contient:

ab
c
def-gh
ij

Après l'appel `p_rec(lb, 'ab+c+def-gh+ij', '-')` la liste `lb` contient:

ab+c+def
gh+ij

- b) La procédure `p_rec` extrait d'une chaîne de caractères `t` les sous-chaînes séparées par le séparateur `s` et les ajoute à la liste `lb`.
- c) 

```
procedure p_iter(lb: TListBox; texte, sep: string);
var p: integer;
begin
  p := pos(sep, texte);
  while p > 0 do
  begin
    lb.Items.Append(copy(texte,1,p-1));
    delete(texte, 1, p);
    p := pos(sep, texte);
  end;
  lb.items.append(texte);
end;
```