

امتحان مادة برمجة 1  
المراجع غير مسموحة

جامعة دمشق  
كلية الهندسة المعلوماتية  
حل مسائل برمجة ١

### السؤال الأول:

بدأ موظف عمله في الساعة  $x$  و الدقيقة  $y$  ، و أنهاء في الساعة  $r$  و الدقيقة  $s$   
اكتب تابع بلغة باسكال و سمه Duration تمرر إليه المتحولات السابقة و يرد الفترة الزمنية التي قضاها الموظف في عمله بالدقائق.

**الحل:**

الكود

```
program test;
type
hour=0..23;
m=0..59;
function Duration(x,r:hour;y,s:m):integer;
begin
if(x<r)then
Duration:=(r*60)+(s-y)-(x*60)
else
Duration:=((24-x)*60)+(s-y)+(r*60);
end;

var x,y,r,s:integer;
begin
readln(x,y,r,s);
writeln(Duration(x,r,y,s));
readln;
```

### السؤال الثاني:

اكتب إجرائية بلغة باسكال و سمها Fraction تمرر إليه متحول حقيقي  $n$  و يرد قيمة كسرية (بسط و مقام) على نحو **مختزل**. مثال من أجل  $n=3.5$  يرد الكسر المختزل  $7/2$ .

**الحل:**

الكود

```
program test;
procedure Fraction(n:real);
var n1,m:real; i,t:integer;
begin
m:=n;
t:=trunc(n);
n1:=n1-t;
i:=1;
while(n1<>0)do
begin
i:=i+1;
n:=n*i;
t:=trunc(n);
n1:=n-t;
if(n1<>0)then
n:=m;
```

```

end;
writeln(n:6:2,'/',i);
end;
var n:real;
begin
readln(n);
fraction(n);
readln;
end.

```

### طريقة ثانية:

```

program test;
procedure Fraction(n:real);
var n1:real; i:integer;
begin
n1:=n;
i:=1;
while(frac(n)<>0)do
begin
i:=i+1;
n:=n*i;
if(frac(n)<>0)then
n:=n1;
end;
writeln(n:6:2,'/',i);
end;
var n:real;
begin
readln(n);
fraction(n);
readln;
end.

```

### السؤال الثالث:

اكتب تابع بلغة باسكال وسمه SumDigit تمرر إليه عدد صحيح موجب n ويرد مجموع خانات العدد n.  
مثال: من أجل  $n=6789$ ، على التابع أن يرد القيمة 30.

### الحل:

#### الكود

```

program test;
function SumDigit(n:integer):integer;
var res:integer;
begin
res:=0;
while(n<>0)do
begin
res:=res+n mod 10;
n:=n div 10;
end;
SumDigit:=res;
end;
var n:integer;

```

```

begin
writeln('enter the n: ');
readln(n);
while(n<0)do
begin
writeln('enter the n>=0: ');
readln(n);
end;
writeln(SumDigit(n));
readln;
end.

```

**السؤال الرابع :**

- اكتب برنامجاً بلغة باسكال يقوم بقراءة عدد صحيح number من لوحة الملامس ثم أكتب التتابع التالية:
- التابع oddeven يرد القيمة المنطقية في حال كون العدد زوجي.
  - التابع العودي (التراجعي) Sumnumber يقوم بإيجاد مجموع كل الأعداد الصحيحة بين العدد 1 و العدد number.
  - التابع Number\_Digits يرد عدد خانات العدد number (مثال من أجل number=150 فإن التابع يرد 3).
  - التابع PrimeFactors يقوم بإيجاد و طباعة كافة العوامل الأولية للعدد number (مثال: من أجل number=150 فإن التابع يطبع العوامل الأولية على الشكل 2\*3\*5\*5).

**الحل:**

## الكود

```

program test;
function oddeven(number:integer):boolean;
var ok:boolean;
begin
ok:=false;
if(number mod 2=0)then
ok:=true
else
ok:=false;
oddeven:=ok;
end;
function sumnumber(number:integer):integer;
begin
if(number=1)then
sumnumber:=1
else
sumnumber:=number+sumnumber(number-1);
end;

function Number_Digit(number:integer):integer;
var res:integer;
begin
res:=0;
while(number<>0)do
begin
res:=res+1;
number:=number div 10;
end;
Number_Digit:=res

```

```

end;
procedure primeFactors(number:integer);
var i:integer;
begin
for i:=2 to number do
begin
while(number mod i=0)do
begin
write(i);
if(number<>i)then
write('*');
number:=number div i;
end;
end;
end;
var number:integer;

begin
writeln('enter the number: ');
readln(number);
writeln(oddeven(number));
writeln(Sumnumber(number));
writeln(Number_Digit(number));
primeFactors(number);

readln;
end.

```

### السؤال الخامس:

- اكتب برنامجاً بلغة باسكال تعرف فيه نسقاً اسمه vec عدد عناصره الصحيحة n . ثم اكتب الإجراءات و التتابع التالية:
- الإجرائية ReadVec تقوم بتعبئة كافة عناصر النسق من لوحة الملامس، و حفظ الموجب منها.
  - الإجرائية printmat تقوم بطباعة كافة عناصر النسق و التي قيمتها أكبر من القيمة الوسطية لعناصر النسق.
  - التابع FindSec يرد قيمة ثاني أكبر عنصر في النسق (**دون فرز النسق**).
  - الإجرائية MaxFreq ترد كخرج العدد الأكثر تواتراً (تكراراً) في النسق و عدد مرات تكراره.
  - التابع التراجعي (العودي) RecSum يرد كخرج مجموع عناصر الشعاع ذات الدليل الفردي.
  - ثم عرف النسق vec1 من النمط vec و قم باستدعاء الإجراءات على الترتيب التالي:
    - قراءة النسق vec1.
    - استدعاء التابع FindSec من أجل النسق vec1.
    - استدعاء التابع Finds من أجل النسق vec1.
    - استدعاء الإجرائية recsum من أجل النسق vec1.
    - استدعاء الإجرائية printmat من أجل النسق vec1.

الحل:

## الكود

```

program test;
const max_n=50;
type
vec=array[1..max_n] of integer;
procedure ReadVec(var mat:vec;n:integer);
var i:integer;
begin
for i:=1 to n do
begin
write('mat[' ,i,']');
readln(mat[i]);
while(mat[i]<=0)do
begin
write('mat[' ,i,']');
readln(mat[i]);
end;
end;
end;
function average_vec(mat:vec;n:integer):real;
var i:integer;temp:real;
begin
temp:=0;
for i:=1 to n do
temp:=temp+mat[i];
temp:=temp/n;
average_vec:=temp;
end;
procedure printmat(mat:vec;n:integer);
var i:integer; temp:real;
begin
temp:=average_vec(mat,n);
for i:=1 to n do
begin
if(mat[i]>temp)then
write(mat[i],',');
end;
writeln;
end;
function FindSec(mat:vec;n:integer):integer;
var i,max,s:integer;
begin
i:=2;
max:=mat[i];
for i:=2 to n do
if(mat[i]>max)then
max:=mat[i];
s:=0;
for i:=1 to n do
if(mat[i]<max)and (mat[i]>s)then

```

```

s:=mat[i];
if(s=0)then s:=max;
FindSec:=s;
end;
function RecSum(mat:vec;n:integer):integer;
begin
if(n=1)then
RecSum:=mat[n]
else
if(n mod 2=1)then
recsum:=mat[n]+recsum(mat,n-1)
else
recsum:=recsum(mat,n-1);
end;

function max_mat(mat:vec;n:integer;var index:integer):integer;
var i,max:integer;
begin
max:=mat[1];
for i:=2 to n do
if(mat[i]>max)then
begin
max:=mat[i];
index:=i;
end;
max_mat:=max

end;
procedure MaxFreq(mat:vec; n:integer);
var freq:vec; i,index,res:integer;
begin
index:=1;
for i:=1 to n do
freq[i]:=0;
for i:=1 to n do
freq[mat[i]]:=freq[mat[i]]+1;
{for i:=1 to n do
writeln(i,' ',freq[i]);}
res:=max_mat(freq,n,index);
writeln('max freq: ',index,' ',res);
end;
var n:integer; vec1:vec;
begin
readln(n);
ReadVec(vec1,n);
writeln('tow max is : ',FindSec(vec1,n));
maxfreq(vec1,n);
writeln('total index odd: ',recsum(vec1,n));
writeln('print in vector >average all element in vector: ');
printmat(vec1,n); readln; end.

```

(وما توفيقى الا بالله)