

algol,n<

begin

comment

GC84BA2

Time: 9583.05s

No buffer GIER:

Time classic: 9439.92

Time turbo: 8811.91 6.7pct

Buffer GIER:

Time classic: 9583.25

Time turbo: 8912.39 7.0pct

```
;
integer alen,alen1,cipher len;
integer initial shift,shift,plugchar1,plugchar2;
integer i,c1,c2,c3,c4,c5,c6;
boolean found;
real procedure clock count;
code clock count;
1, 37;
  z1      , grf p-1  ; RF:=clock count; stack[p-1]:=RF;
e;
select(32);
clock count;
alen:=read integer;
alen1:=alen-1;
cipher len:=read integer;
begin
  integer array alphabet,rotor,delta1,delta2,reflector[0:alen1];
  integer array reverse alphabet[0:63];
  integer array cipher,clear[1:cipher len];
  procedure read alphabet;
  begin
    integer i,c;
    i:=0;
again:
  c:=lyn;
  if c=60 v c=58 then goto again;
  if c=64 then goto exit;
  alphabet[i]:=c;
  reverse alphabet[c]:=i;
  i:=i+1;
  goto again;
exit:
  end read alphabet;
  procedure read rotor;
  begin
    integer i,j,c,c2;
    i:=0;
again:
  c:=lyn;
  if c=60 v c=58 then goto again;
  if c=64 then goto exit;
  c2:=reverse alphabet[c];
  rotor[i]:=c2;
  i:=i+1;
  goto again;
exit:
```

```

    for j:=0 step 1 until alen1 do
    begin
        i:=rotor[j];
        delta1[j]:=(i-j+alen)mod alen;
        delta2[i]:=(j-i+alen)mod alen
    end delta
end read rotor;
procedure read reflector;
begin
    integer i,c,c2;
    i:=0;
again:
    c:=lyn;
    if c=60 v c=58 then goto again;
    if c=64 then goto exit;
    c2:=reverse alphabet[c];
    reflector[i]:=c2;
    i:=i+1;
    goto again;
exit:
    end read reflector;
    procedure read cipher;
    begin
        integer i,c,c2;
        i:=0;
again:
        c:=lyn;
        if c=60 v c=58 then goto again;
        if c=64 then goto exit;
        c2:=reverse alphabet[c];
        i:=i+1;
        cipher[i]:=c2;
        goto again;
exit:
    end read cipher;
    integer procedure plugboard(c1);
    value c1;
    integer c1;
    plugboard := if c1=plugchar1 then plugchar2 else
        if c1=plugchar2 then plugchar1 else c1;
    integer procedure replace delta(c1,delta);
    value c1;
    integer c1;
    integer array delta;
    replace delta:=(c1+delta[(c1-shift+1000*alen)mod alen])mod alen;
    read alphabet;
    read rotor;
    read reflector;
    read cipher;
    for initial shift:=0 step 1 until 3 do
    for plugchar1:=0 step 1 until alen1 do
    for plugchar2:=0 step 1 until alen1 do
    begin
        shift:=initial shift;
        for i:=1 step 1 until cipher len do
        begin
            c1:=cipher[i];
            c2:=plugboard(c1);
            shift:=shift+1;
            c3:=replace delta(c2,delta1);
            c4:=reflector[c3];
            c5:=replace delta(c4,delta2);
            c6:=plugboard(c5);
            clear[i]:=c6

```

```

end cipher len;
found:=false;
for i:=1 step 1 until cipher len-2 do
begin
  comment
    Look for FEM:

    012345678901234567890123456789
    ABCDEFGHIJKLMNOPQRSTUVWXYZÆØÅ
    ;
    if clear[i]=5 ^
      clear[i+1]=4 ^
      clear[i+2]=12 then found:=true
end;
if found then
begin
  writecr;
  write({ddd},initial shift,plugchar1,plugchar2);
  writetext({<  });
  writechar(60);
  for i:=1 step 1 until cipher len do
  writechar(alphabet[clear[i]]);
  writechar(58)
end
end for plugchar2 plugchar1 initial shift
end inner loop;
writecr;
writetext({<Time:  });
write({ddddddd.dd},clock count)
end;
run<
28
76
ABCDEFGHIJKLMNOPQRSTUVWXYZÆØÅ
DBPEMGZALRNKØYTJXEOÅFUHCQVSI
OHVFMDBÅNSTEJAUZØKLPCEQYRI
ÅLNPSGÅJDRØMYUHÅBXUOUKVQGLTBÅJVXZISFTØZFHAIELDFLÅKHÅDHIYZZÅXZISFTØVPPVVIQGD

```