179

Wit 3

Computer Programme GaussLine

By S.E.A. Jansen.

TU Delft

Delft University of Technology

Faculty of Civil Engineering
Hydraulic and Geotechnical Engineering Division
Hydraulic Engineering Group

Contents

Introduction

This report on the GaussLine computer programme can be seen as an annex to the report "Discussion of the one-line calculations for the nourishments on Sylt (1972, 1978)". This discussion therefore will be referred to several times. In fact formulas often will be referrenced by their "original" number from that report accompanied by the letter "D" (from Discussion). Usually they are repeated here as well. The formulas that are used in this report only will be denoted by a letter "G" (from GaussLine).

The GaussLine computer programme has been developed in order to include the sinusoidal relation between the angle of wave incidence and the sand transport capacity in a one-line computation.

In the first Chapter of this report the theoretical background of the programme is explained. Chapter 2 describes its validation and in Chapter 3 the results of some computation-experiments are shown. Finally in the last chapter full programme documentation can be found.

1 Theoretical background

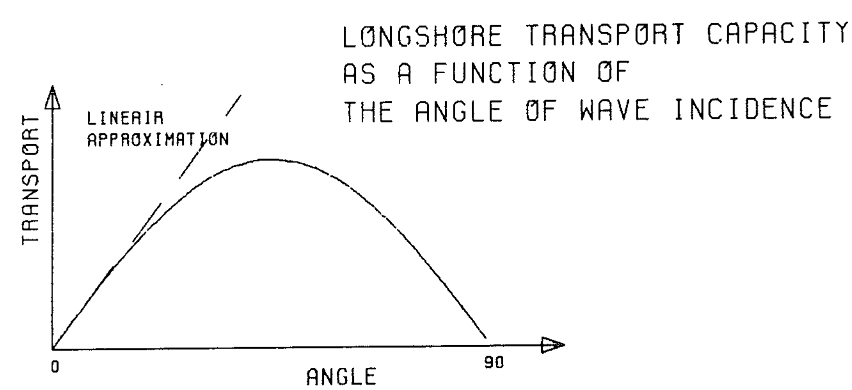
In this chapter the theoretical background of the GaussLine computer programme will be explained.

Purpose

The GaussLine computer programme has been developed in order to evaluate the influence of a very important simplification of the one-line model.
One of the basics of the one-line model consists in the assumed linear relationship between changes in coastline direction and changes in sand transport capacity.
This is expressed by formula D(1.07), which can be found in Section D1.1 (Par. assumptions).

$$\frac{\partial S}{\partial \beta} = -s \qquad D(1.07)$$

A further discussion of this simplification has been given in Section D1.2 (Par. equation of motion), where especially Figure D1.06 is of importance.



LONGSHORE TRANSPORT CAPACITY FIGURE 1.06

From this figure it can be concluded that the simplification of D(1.07) is valid only when small changes in coastline direction are considered. In that section it is explained also that besides the orientation of the coastline the angle of wave incidence plays an important role too.
For a description of the exact details please see the sections mentioned.

In order to apply successfully the analytical methods in one-line theory formula D(1.07) has to be accepted.
However when a numerical approach is used it is possible to take into consideration the sinusoidal shape of Figure D1.06.

To compare the outcome of such a computation with the coastline found by means of an analytical solution has been the main reason for developping this computer programme.

Restrictions

The GaussLine computer programme is, as its name already suggests, only suitable for those situations, in which the initial shape of the coastline can be represented by a Gaussian curve.
Such a curve is given by Formula D(1.08) (Section D1.1, Par. analytical solution).

$$y = CA * \exp[-CB * x^2] \qquad D(1.08)$$

The behaviour in time of such a coastline can be analysed.
As one of the possibilities a Gaussian curved coastline may be developped from a stockpile-type beachfill.

Approach

The basic equation of the one-line theory, equation D(1.02) has been used as a starting point.

$$\frac{\partial S}{\partial x} + h * \frac{\partial y}{\partial t} = 0 \qquad D(1.02)$$

This equation of continuity contains the factors "x", "t", "y", "h" and "S". Hereby "x" and "t" are independent factors characterizing place and time. The shoreline position "y" is the resulting factor, which gets computed. The value of the profile height "h" follows from the initial shape (as provided by the user of the programme).
For the determination of "S", the longshore transport capacity, formula D(1.23) has been used with ϕ0r (=ßr) instead of ß (see page D18).

$$S(\beta r) = -s * 0.5 * \sin[2*\beta r] \qquad D(1.23)$$

As can be noticed from Section D1.2, between D(1.20) and D(1.21), to achieve Formula D(1.23) the influence of changes in "cbr", the wave speed at the breakerline, has been neglected.
Although strictly not correct its effect on the sinusoidal shape of Figure D1.06 can be considered a minor one compared to the linear assumption, which in particular is investigated here.
Since it should complicate the analysis considerably, when the influence of "cbr" should be taken in account, the approach of Formula D(1.23) has been chosen.

Numerical treatment

The main purpose has been to generate numerically results, which can be compared to those of the analytical computation.
Therefore in solving equation D(1.02) numerically the most simple numerical method has been used known as, Forward Time Central Space (FTCS). This policy is also applied by using central differences for the estimation of D(1.23).

So the set equations to solve numerically yields,

$$y[i,n+1] = y[i,n] - \frac{dt}{2*h*dx} (S[i-1,n] - S[i+1,n]) \quad (G.01)$$

$$S[i,n] = -0.5 * s * \sin[2*\phi0r[i,n]] \quad (G.02)$$

$$\phi0r = \arctan[\frac{y[i+1,n] - y[i-1,n]}{2 * dx}] - \phi0 \quad (G.03)$$

Where y[i,n] denotes y(i*dx,n*dt). For "S" and "ϕ0r" the similar notation has been used.
As mentioned before (Par. restrictions) as an initial condition a Gaussian curve according to formula D(1.08) is applied as only option. Instead of at $\pm$infinite the boundary conditions of y=0 are imposed at x=$\pm$4*xcf(te), where xcf(te) denotes the position of the points of contraflexure at the end of the computation period. In this way throughout the whole computation the side-boundaries of the model are located sufficiently "far away" so that the computation is not influenced by loss of sand through the side boundaries.

Of course FTCS is a rather unsophisticated method with strict stability demands. As mentioned before this has not been the primary concern when developping this programme. However the programme has been structured in such a way that another numerical method can be implemented without substantial problems.

2 Validation

In this chapter a description will be given of the validation of the programme.

Data used for testing

Data-Table 1 shows two sets of test-data. With these two sets several test-cases can be constructed.

Data Table 1		set(A)	set(B)
Top of the fill	[m]	100	300
Width of the fill	[m]	300	100
Volume of the fill	[m^2]	1E6	1E6
maximum coastline angle	[deg]	11	61
coastal constant	[m^2/yr/rad]	5e4	5e5

The essential difference between set(A) and set(B) is the magnitude of the gradients in longshore direction. As shown in Data-Table 1 the maximum angle of the coastline with respect to the x-axis yields only 11.4 degrees (0.199 rad) for set(A). But for set(B) this value amounts to 61.2 degrees (1.068 rad), which is a considerably larger value.

Method of validation

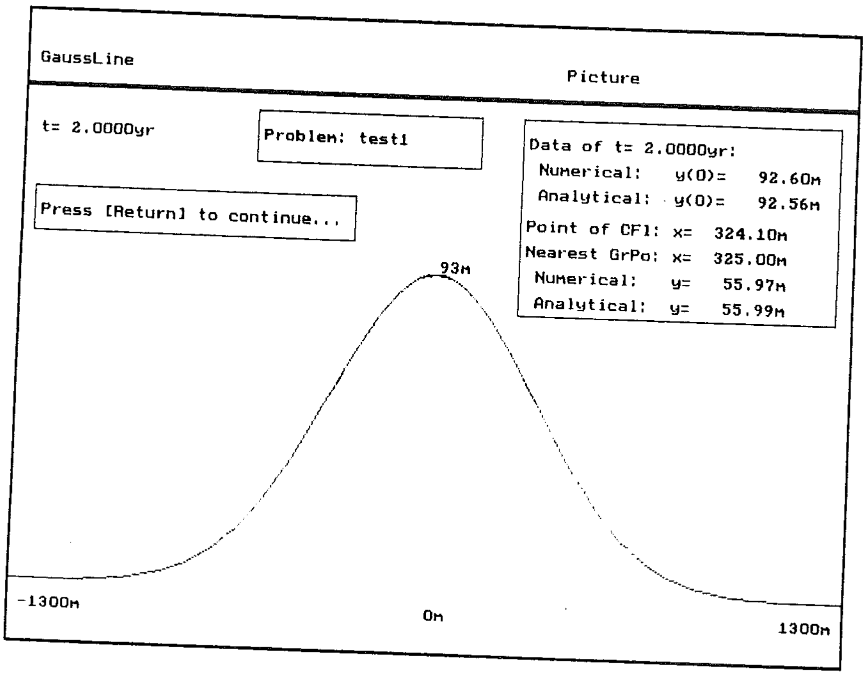
Both the sets of Data-Table 1 have been used to test the programme. The programme can be validated by offering it as an input a situation corresponding to the assumptions of the analytical method.
In that respect the assumed linear relation between the angle of wave incidence and the sand transport capacity has to be considered an essential test parameter.
In such a situation the numerical and analytical solution should yield the same coastline. Therefore the accuracy of the numerical computation can be judged.

Test-cases

Test1 and test2:
Characteristics:
* set(A), thus small geometry angles
* linear relation
* ϕ0=0 degrees, thus small relative angles

To build up some confidence in the programme at first a situation has been offerred, where no difficulties at all are present.
The characteristics show a tame situation, which corresponds to the assumptions of the one-line model at all points. Therefore the coastline computed numerically and the one obtained by the analytical method have to be identical (or expressed in a more correct way: the difference between those two solutions can be brought within every requested margin of inaccuracy by reduction of the time step and the grid distance).

A computation over a period of 2 years (test1) and also over a period of 10 years (test2) yield results, which are quite satisfying as can be seen form the programme output.



--- GaussLine ---

**** Problem: test1 ****

--- GaussLine ----------------------------------- Problem Data ----

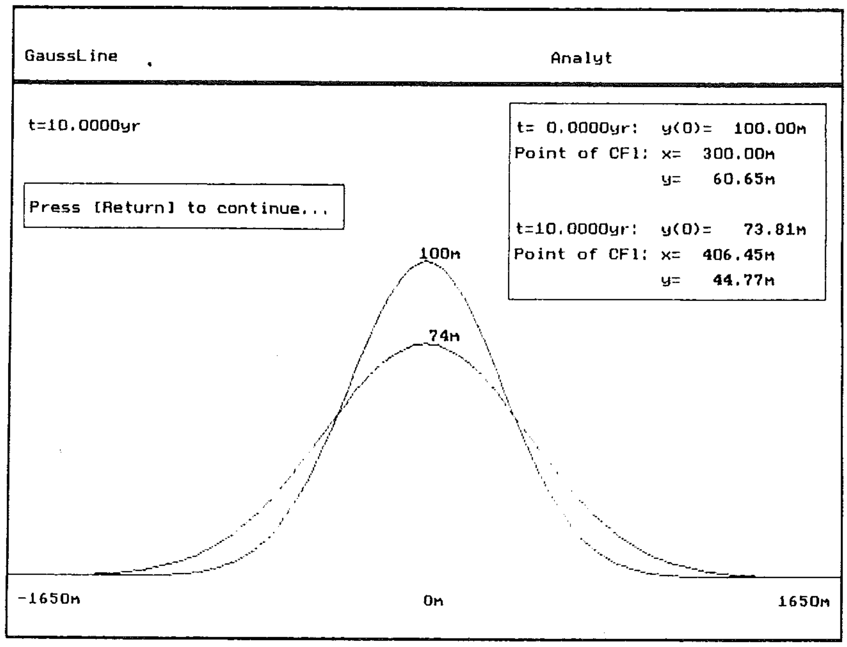
Period after placement of the fill	[yr]	:	2.00
Volume of the fill	[m3]	:	1000000
Profile height	[m]	:	13.30
Coastal constant	[m3/yr/rad]	:	50000.00
Fictive initial period	[yr]	:	11.97
Angle of wave incidence	[rad]	:	0.0000

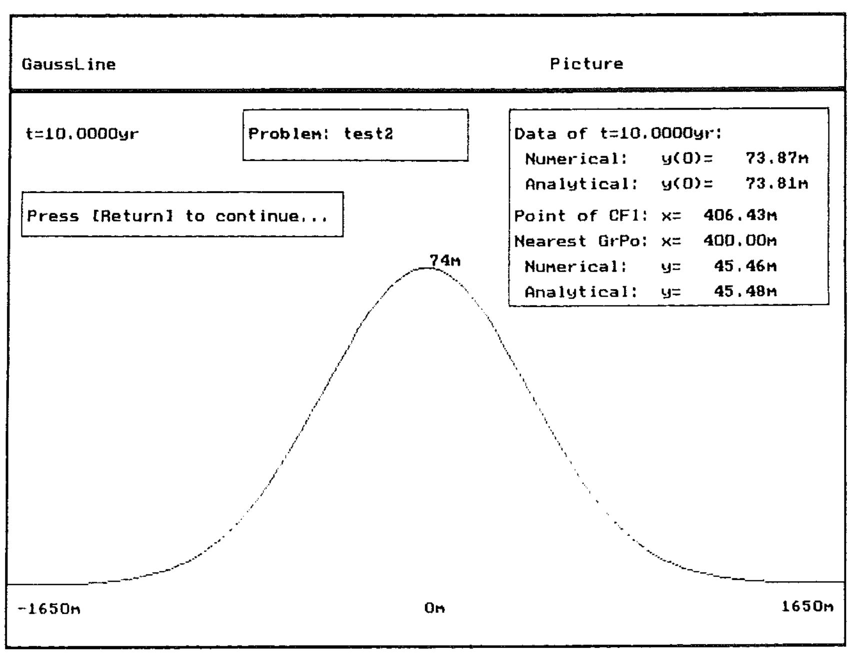
--- GaussLine ---------------- Data for Numerical Computation ----

Time step	[yr]	:	0.0050
Grid distance	[m]	:	25.00
Number of points	[-]	:	52

--- GaussLine ---------------------- Results from Computation ----

x-coordinate x [m]	shoreline position y (analytical) [m]	y (numerical) [m]
-100.00	88.26	88.29
-75.00	90.12	90.15
-50.00	91.47	91.50
-25.00	92.29	92.33
0.00	92.56	92.60
25.00	92.29	92.33
50.00	91.47	91.50
75.00	90.12	90.15
100.00	88.26	88.29

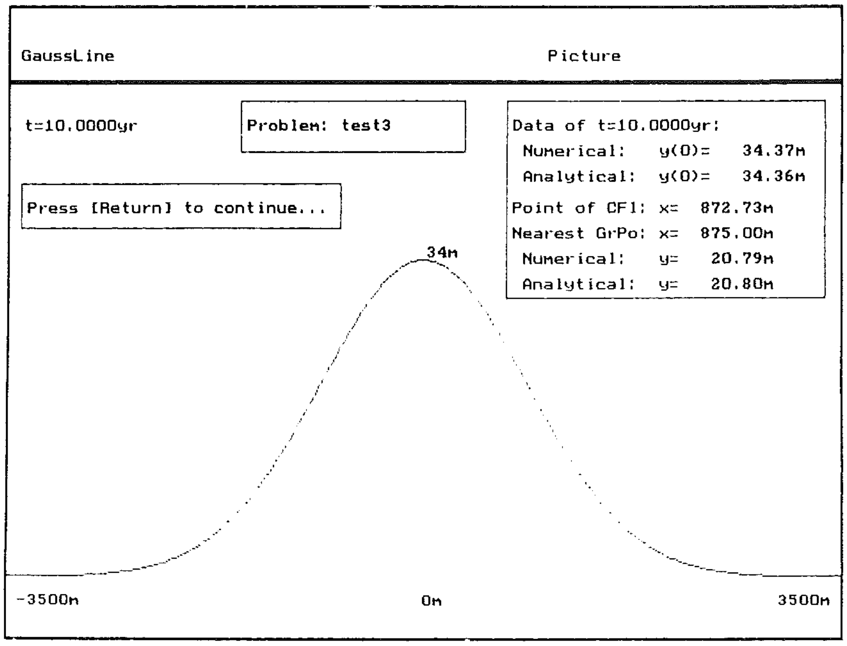

GaussLine
Analyt
t=10.0000yr
t= 0.0000yr: y(0)= 100.00m
Point of CF1: x= 300.00m
y= 60.65m
t=10.0000yr: y(0)= 73.81m
Point of CF1: x= 406.45m
y= 44.77m
Press [Return] to continue...
100m
74m
-1650m
0m
1650m


GaussLine
Picture
t=10.0000yr
Problem: test2
Data of t=10.0000yr:
Numerical: y(0)= 73.87m
Analytical: y(0)= 73.81m
Point of CF1: x= 406.43m
Nearest GrPo: x= 400.00m
Numerical: y= 45.46m
Analytical: y= 45.48m
Press [Return] to continue...
74m
-1650m
0m
1650m

```
--- GaussLine ------------------------------------------------------
                **** Problem: test2 ****

--- GaussLine ---------------------------------- Problem Data ----
Period after placement of the fill  [yr]       :      10.00
Volume of the fill                  [m3]       :    1000000
Profile height                      [m]        :      13.30
Coastal constant                    [m3/yr/rad]:   50000.00
Fictive initial period              [yr]       :      11.97
Angle of wave incidence             [rad]      :      0.0000

--- GaussLine --------------- Data for Numerical Computation ----
Time step                           [yr]       :      0.0100
Grid distance                       [m]        :      25.00
Number of points                    [-]        :       66

--- GaussLine --------------------- Results from Computation ----
 x-coordinate                 shoreline position
    x [m]         y (analytical) [m]        y (numerical) [m]
   -100.00             71.61                      71.66
    -75.00             72.56                      72.62
    -50.00             73.25                      73.31
    -25.00             73.67                      73.73
      0.00             73.81                      73.87
     25.00             73.67                      73.73
     50.00             73.25                      73.31
     75.00             72.56                      72.62
    100.00             71.61                      71.66
```

Test3 and test4:
Characteristics:
* set(B), thus wide geometry angles
* linear relation
* $\phi 0$= 0degrees, thus wide relative angles (test3)
* $\phi 0$= 30degrees, thus wide relative angles (test4)

Since the gradients in transport capacity are of importance for the deformation of the coastline, using a linear relation has to yield the same result as that obtained by the analytical solution.
Thus the wide angles of the geometry of the fill and also the substantial angle of wave incidence (test4) may not cause a difference of numerical and analytical solution.
Also in these two test-cases the programme output is quite satisfactory.



--- GaussLine --

**** Problem: test3 ****

--- GaussLine ---------------------------------- Problem Data ----

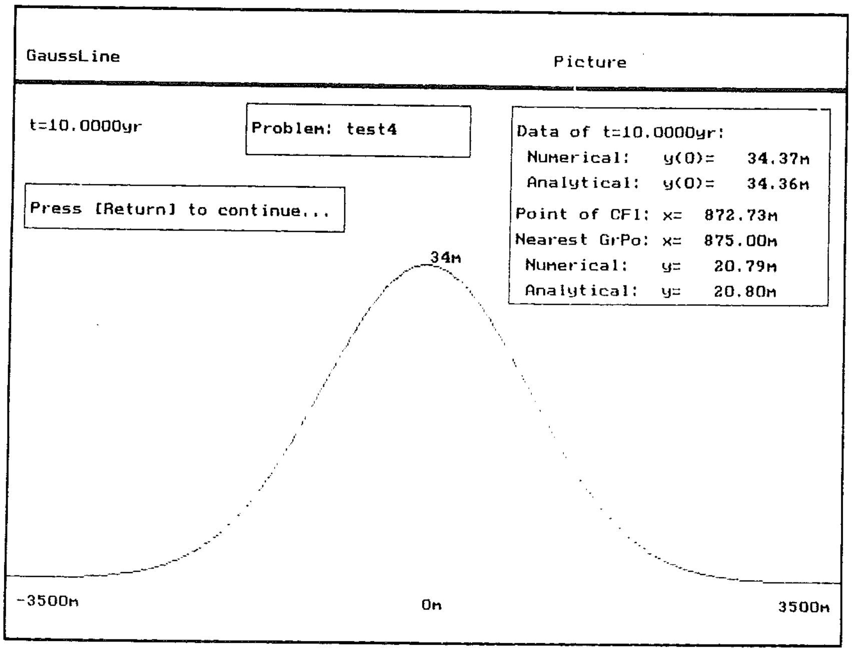
Period after placement of the fill	[yr]	:	10.00
Volume of the fill	[m3]	:	1000000
Profile height	[m]	:	13.30
Coastal constant	[m3/yr/rad]	:	500000.00
Fictive initial period	[yr]	:	0.13
Angle of wave incidence	[rad]	:	0.0000

--- GaussLine --------------- Data for Numerical Computation ----

Time step	[yr]	:	0.0042
Grid distance	[m]	:	25.00
Number of points	[-]	:	140

--- GaussLine --------------------- Results from Computation ----

x-coordinate x [m]	shoreline position y (analytical) [m]	y (numerical) [m]
-100.00	34.14	34.15
-75.00	34.24	34.24
-50.00	34.31	34.31
-25.00	34.35	34.36
0.00	34.36	34.37
25.00	34.35	34.36
50.00	34.31	34.31
75.00	34.24	34.24
100.00	34.14	34.15



--- GaussLine --

**** Problem: test4 ****

--- GaussLine ------------------------------------ Problem Data ----

Period after placement of the fill	[yr]	:	10.00
Volume of the fill	[m3]	:	1000000
Profile height	[m]	:	13.30
Coastal constant	[m3/yr/rad]	:	500000.00
Fictive initial period	[yr]	:	0.13
Angle of wave incidence	[rad]	:	0.5236

--- GaussLine ---------------- Data for Numerical Computation ----

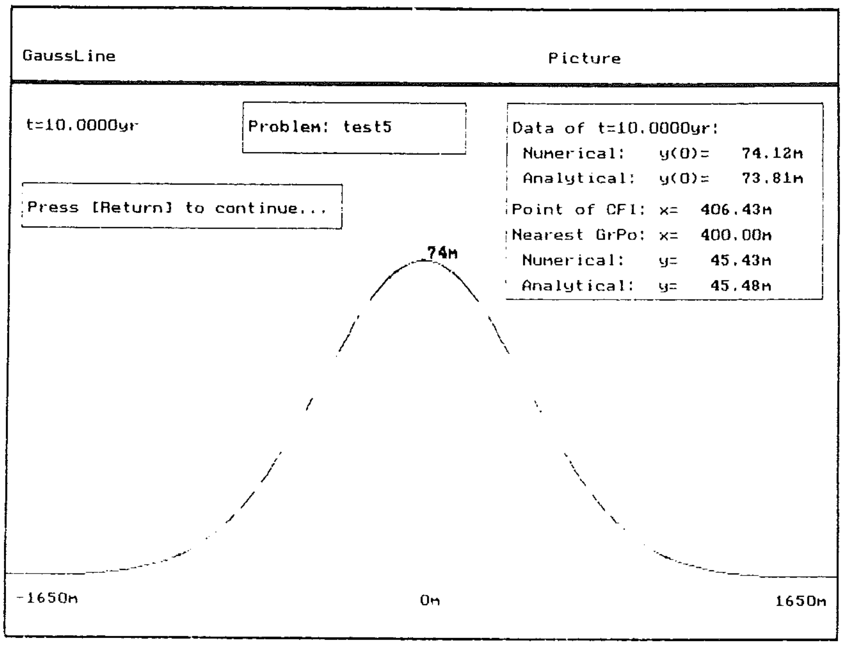
Time step	[yr]	:	0.0042
Grid distance	[m]	:	25.00
Number of points	[-]	:	140

--- GaussLine ---------------------- Results from Computation ----

x-coordinate x [m]	shoreline position y (analytical) [m]	y (numerical) [m]
-100.00	34.14	34.15
-75.00	34.24	34.24
-50.00	34.31	34.31
-25.00	34.35	34.36
0.00	34.36	34.37
25.00	34.35	34.36
50.00	34.31	34.31
75.00	34.24	34.24
100.00	34.14	34.15

Test5:
Characteristics:
* set(A), thus small geometry angles
* sinusoidal relation
* $\phi 0=0$ degrees, thus small relative angles

Although a sinusoidal relation has been used in this test, the coastline computed numerically and analytically have to yield the same result.
Since the relative angles are small (less than 12 degrees in this case) the linear approximation of the sinusoidal curve is quite accurate. For more information on this point refer to Section D1.2 (Par. equation of motion).
From the programme output it can be seen that the programme also gets through this test case correctly.



--- GaussLine ---

**** Problem: test5 ****

--- GaussLine ---------------------------------- Problem Data ----

Period after placement of the fill	[yr]	:	10.00
Volume of the fill	[m3]	:	1000000
Profile height	[m]	:	13.30
Coastal constant	[m3/yr/rad]	:	50000.00
Fictive initial period	[yr]	:	11.97
Angle of wave incidence	[rad]	:	0.0000

--- GaussLine ---------------- Data for Numerical Computation ----

Time step	[yr]	:	0.0100
Grid distance	[m]	:	25.00
Number of points	[-]	:	66

--- GaussLine --------------------- Results from Computation ----

x-coordinate x [m]	shoreline position y (analytical) [m]	y (numerical) [m]
-100.00	71.61	71.91
-75.00	72.56	72.87
-50.00	73.25	73.56
-25.00	73.67	73.98
0.00	73.81	74.12
25.00	73.67	73.98
50.00	73.25	73.56
75.00	72.56	72.87
100.00	71.61	71.91

Conclusion

From the previous five te·t-cases it may be concluded that the method used to compute numerically the position of the coastline works out rather well. The approximation of the analytical solution is quite satisfactory. An inaccuracy of less than 0.5m can easily be achieved. For coastal engineering purposes this should be a sufficient value.
From a users point of view the following remark has to be made. The time step necessary to achieve this accuracy is very small, O(0.001yr). Thus a computation over a period of for instance 10yr may take some time, O(1hr).

3 Experimental use of the programme

After that it has been concluded that the programme produces correct results for those situations that can be computed analytically, it also has been used for a few situations, where the analytical method does not apply anymore. In this chapter several of those problem-cases will be discussed.

Approach

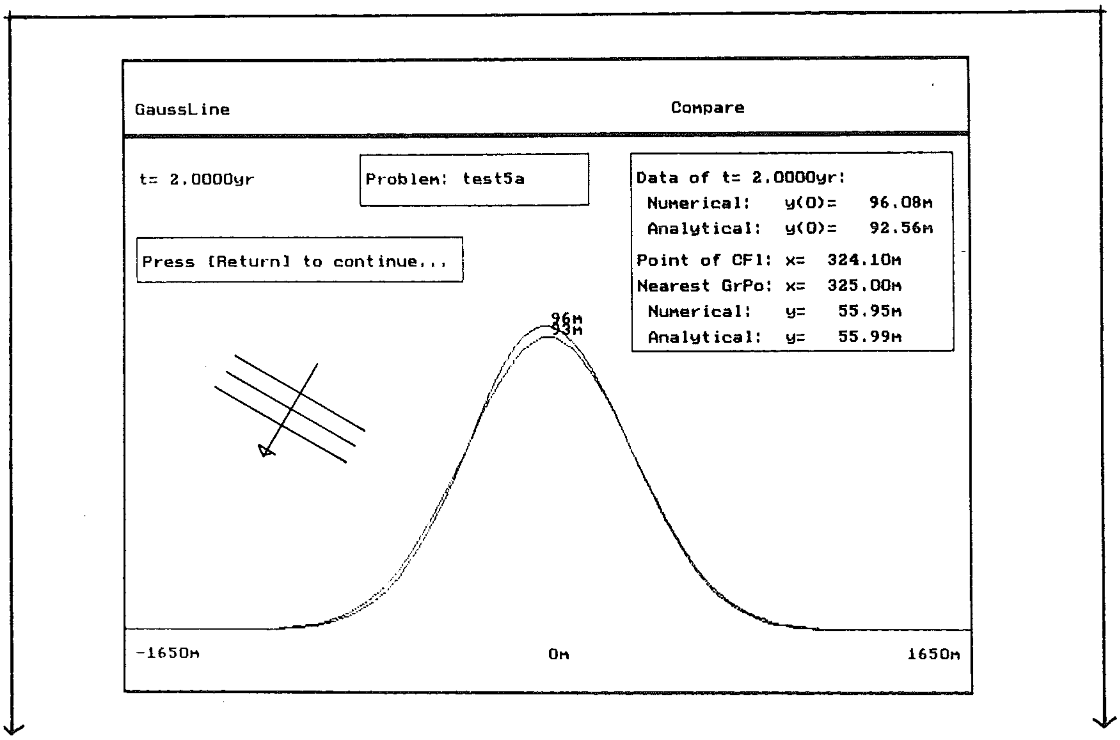
At first the same geometry data, set(A) and set(B) from Data Table 1, have been used to compare the numerical with the analytical solution. Later on forced by the outcome of especially the computations using set(B) a third set, set(C), has been created and has been used in a test problem. Also the situation at Sylt for both the 1972 and the 1978 nourishment has been used as a test-problem.

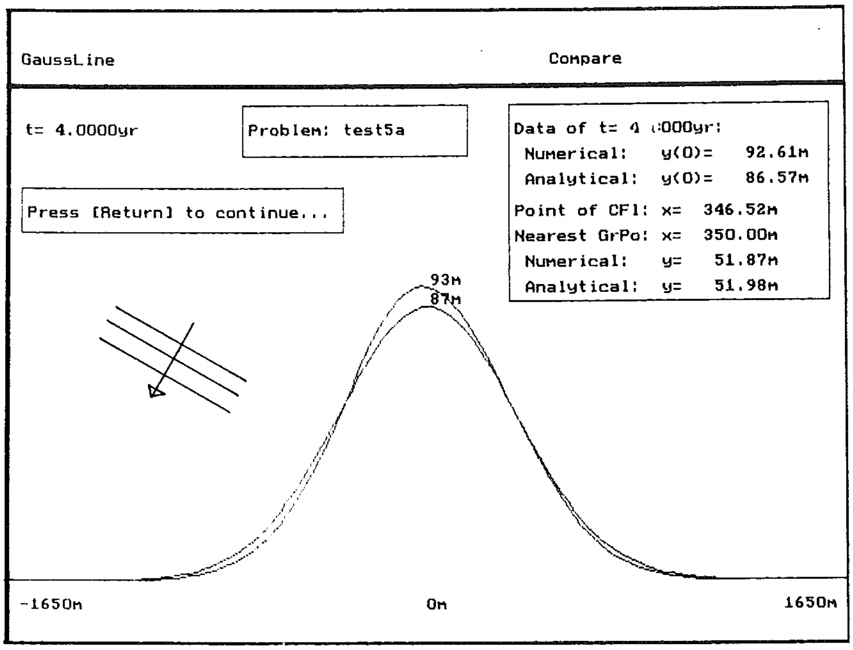
Test-problems

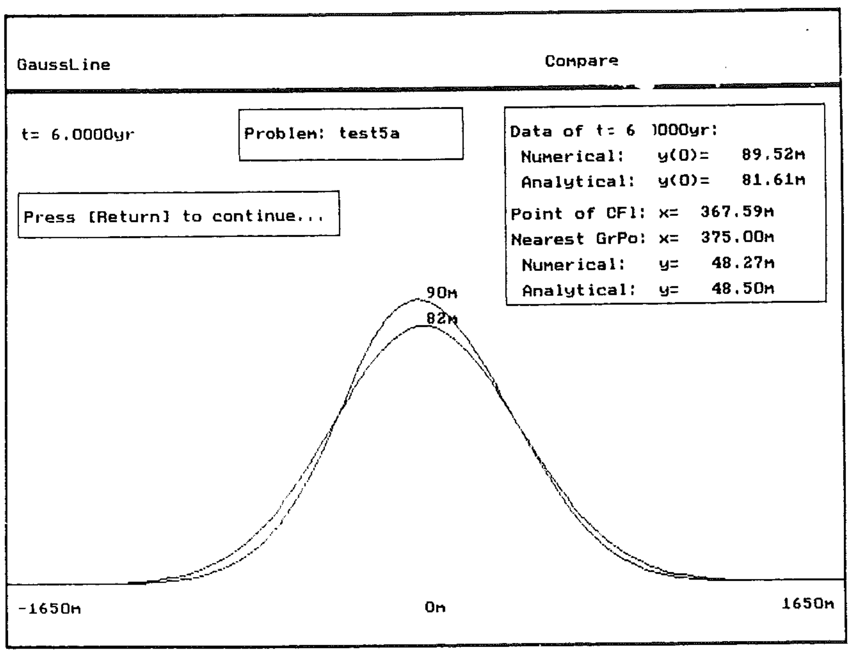
Test5a:
Characteristics:
* set(A), thus small geometry angles
* sinusoidal relation
* ϕ0=-30 degrees, thus moderate relative angles

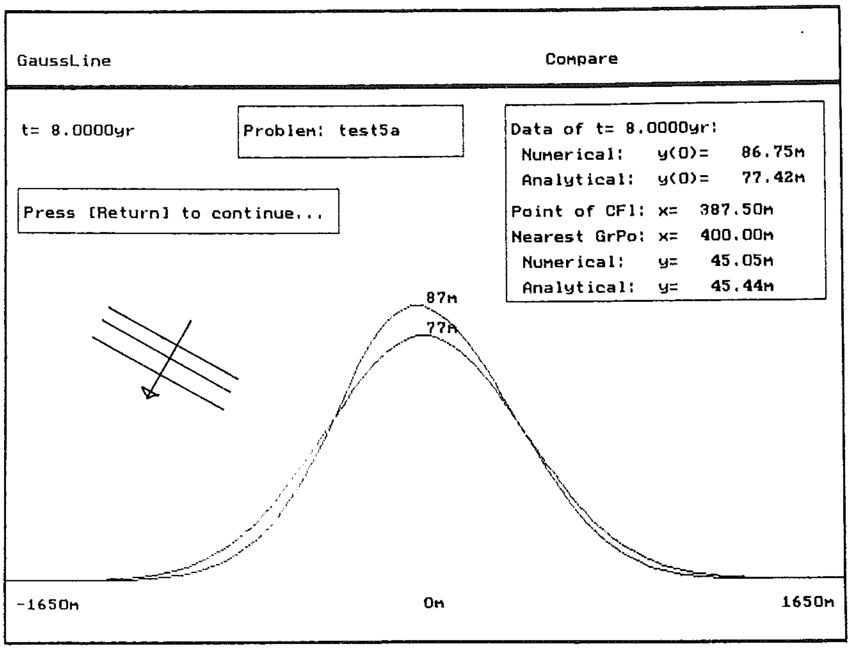
As can be seen from the characteristics the range of relative angle can be denoted by "moderate". In this situation the relative angle varies from -18 degrees to 42 degrees. It is important to notice that the top of the sinusoidal curve (at 45 degrees) is not passed in this case.

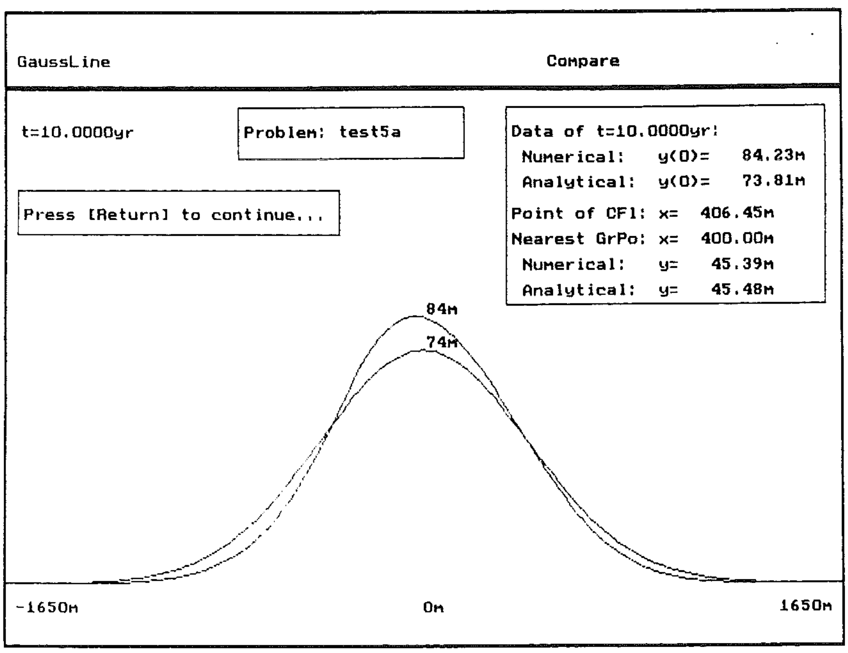
Since the wave-crests approach obliquely from the right an asymmetrical deformation of the Gaussian curve to the left is expected. The obtained output is given below. Five succesive pictures show the behaviour of the coastline in time.




GaussLine
Compare
t= 4.0000yr
Problem: test5a
Press [Return] to continue...
Data of t= 4 :000yr:
Numerical: y(0)= 92.61m
Analytical: y(0)= 86.57m
Point of CFl: x= 346.52m
Nearest GrPo: x= 350.00m
Numerical: y= 51.87m
Analytical: y= 51.98m
93m
87m
-1650m
0m
1650m


GaussLine
Compare
t= 6.0000yr
Problem: test5a
Press [Return] to continue...
Data of t= 6 1000yr:
Numerical: y(0)= 89.52m
Analytical: y(0)= 81.61m
Point of CFl: x= 367.59m
Nearest GrPo: x= 375.00m
Numerical: y= 48.27m
Analytical: y= 48.50m
90m
82m
-1650m
0m
1650m


GaussLine
Compare
t= 8.0000yr
Problem: test5a
Press [Return] to continue...
Data of t= 8.0000yr:
Numerical: y(0)= 86.75m
Analytical: y(0)= 77.42m
Point of CFl: x= 387.50m
Nearest GrPo: x= 400.00m
Numerical: y= 45.05m
Analytical: y= 45.44m
87m
77m
-1650m
0m
1650m


GaussLine
Compare
t=10.0000yr
Problem: test5a
Press [Return] to continue...
Data of t=10.0000yr:
Numerical: y(0)= 84.23m
Analytical: y(0)= 73.81m
Point of CFl: x= 406.45m
Nearest GrPo: x= 400.00m
Numerical: y= 45.39m
Analytical: y= 45.48m
84m
74m
-1650m
0m
1650m

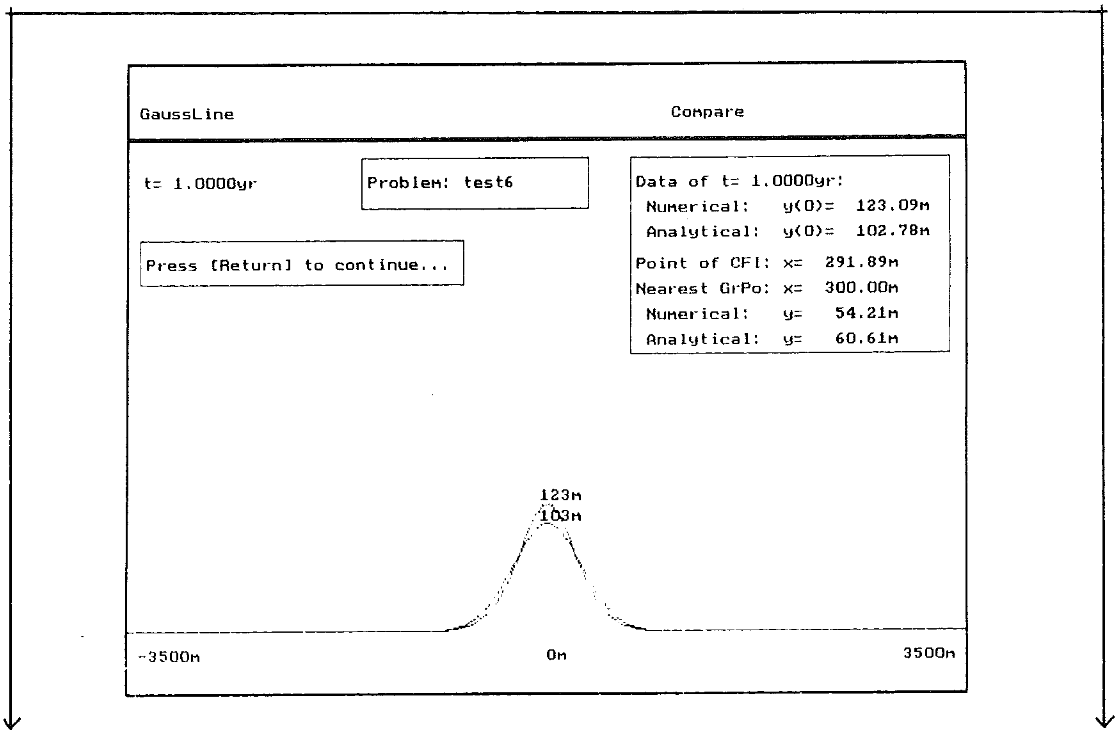
```
--- GaussLine ---------------------------------------------------
              **** Problem: test5a ****


--- GaussLine ---------------------------------- Problem Data ----
Period after placement of the fill  [yr]          :        10.00
Volume of the fill                  [m3]          :      1000000
Profile height                      [m]           :        13.30
Coastal constant                    [m3/yr/rad]:        50000.00
Fictive initial period              [yr]          :        11.97
Angle of wave incidence             [rad]         :      -0.5236


--- GaussLine --------------- Data for Numerical Computation ----
Time step                           [yr]          :       0.0100
Grid distance                       [m]           :        25.00
Number of points                    [-]           :        66


--- GaussLine --------------------- Results from Computation ----
 x-coordinate                   shoreline position
    x [m]          y (analytical) [m]         y (numerical) [m]
   -100.00              71.61                       82.83
    -75.00              72.56                       83.81
    -50.00              73.25                       84.36
    -25.00              73.67                       84.49
      0.00              73.81                       84.23
     25.00              73.67                       83.60
     50.00              73.25                       82.60
     75.00              72.56                       81.28
    100.00              71.61                       79.64
```

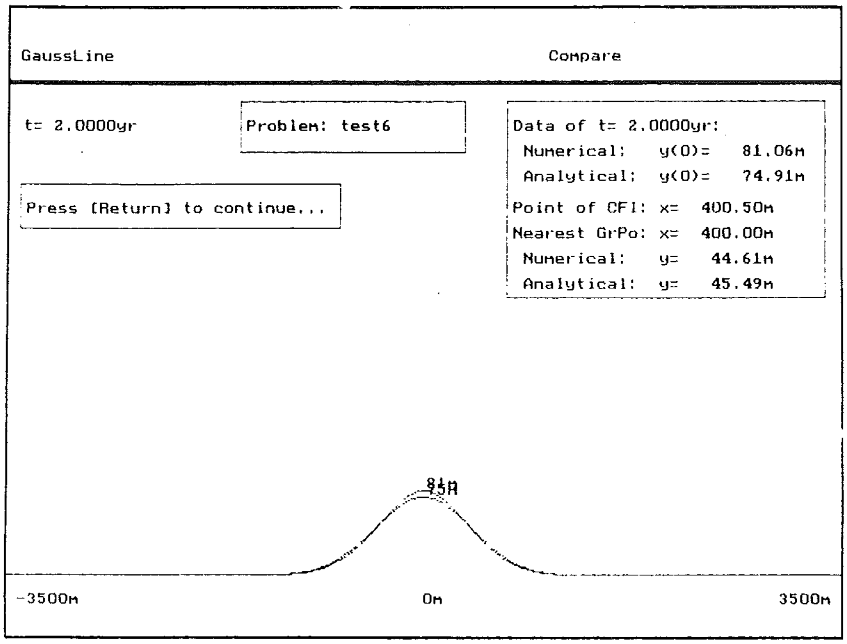
Indeed the output shows an asymmetrical deformation of the Gaussian curve to the left. This can be explained by the fact that the waves incidence obliquely (from the right). Thereby the relative angles change and the situation looses its symmetry.
Another significant effect can be noticed. The numerically computed coastline protrudes more in seaward direction than the analytical solution. This can be understood when is realized that by using a sinusoidal relation the gradients in transport capacity are smaller than in case of using a linear relation. As can be seen from Figure D1.06 the linear approximation yields a maximum gradient. Thus the regression of the Gaussian curve will be smaller in case of a sinusoidal relation.

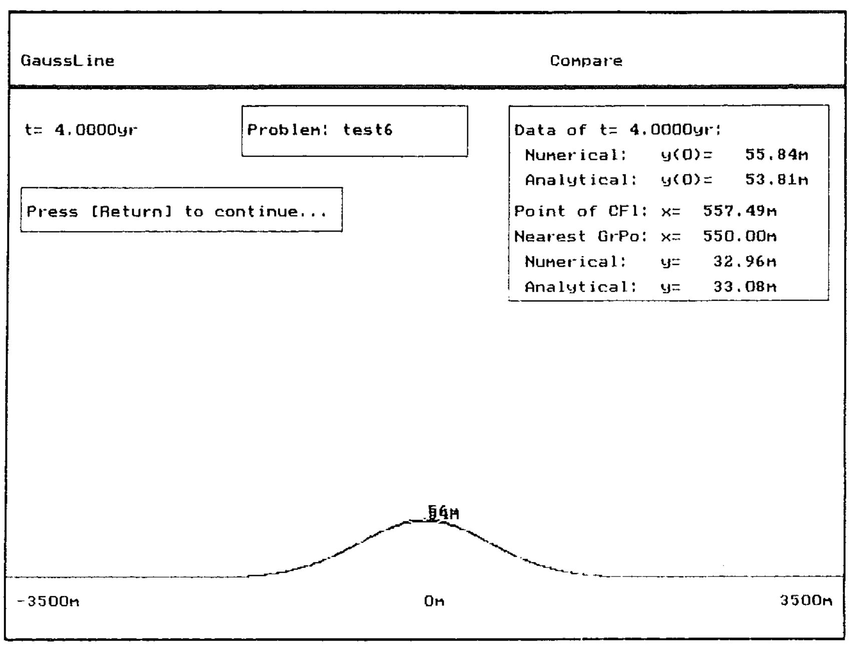
Test6, test6a and test6b:
Characteristics:
* set(B), thus wide geometry angles
* sinusoidal relation
* $\phi 0=0$ degrees, thus wide relative angles

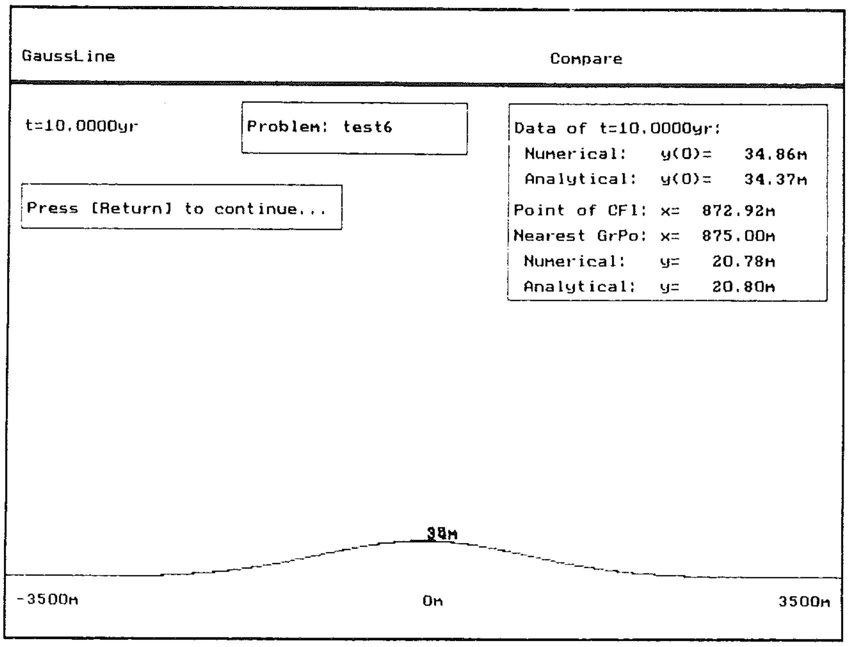
In this initial situation relative angles of 60 degrees occur. Thus the top of the sinusoidal curve is passed. At this point the gradients in transport capacity change of sign and that causes strange behaviour as will become obvious later on.
For now this situation is offerred to the programme just as all previous situations. The angle of wave incidence $\phi=0$ degrees gives a symmetrical problem.

The output of the programme is given below. Again several pictures are provided to study the behaviour of the coastline in time (test6). Because the first two years of the computation period show some strange phenomena this period is analysed in detail (test6a, test6b).




GaussLine
Compare
t= 2.0000yr
Problem: test6
Press [Return] to continue...
Data of t= 2.0000yr:
Numerical: y(0)= 81.06m
Analytical: y(0)= 74.91m
Point of CFl: x= 400.50m
Nearest GrPo: x= 400.00m
Numerical: y= 44.61m
Analytical: y= 45.49m
-3500m
0m
3500m


GaussLine
Compare
t= 4.0000yr
Problem: test6
Press [Return] to continue...
Data of t= 4.0000yr:
Numerical: y(0)= 55.84m
Analytical: y(0)= 53.81m
Point of CFl: x= 557.49m
Nearest GrPo: x= 550.00m
Numerical: y= 32.96m
Analytical: y= 33.08m
-3500m
0m
3500m



--- GaussLine ---

**** Problem: test6 ****

--- GaussLine ------------------------------------ Problem Data ----

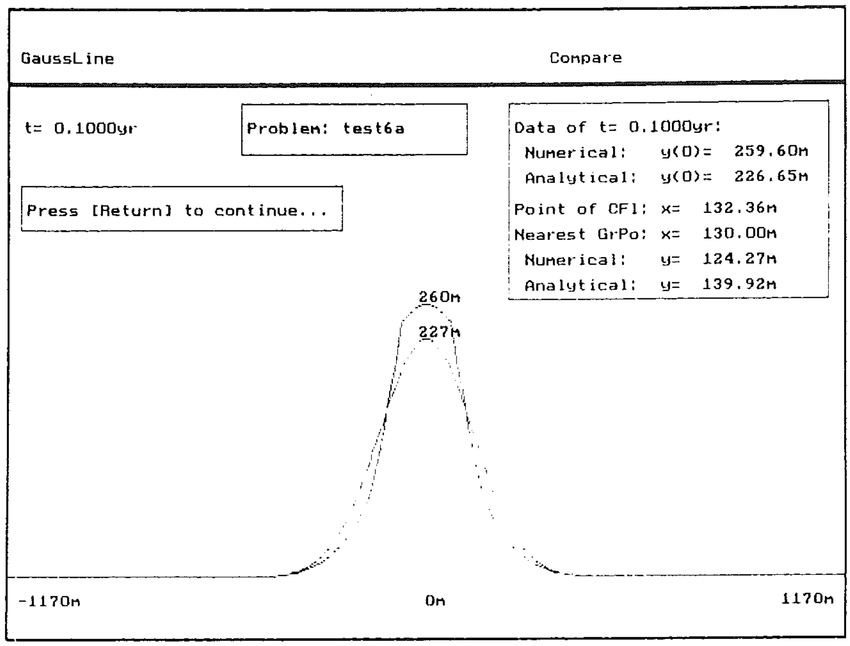
Period after placement of the fill	[yr]	:	10.00
Volume of the fill	[m3]	:	1000000
Profile height	[m]	:	13.30
Coastal constant	[m3/yr/rad]	:	500000.00
Fictive initial period	[yr]	:	0.13
Angle of wave incidence	[rad]	:	0.0000

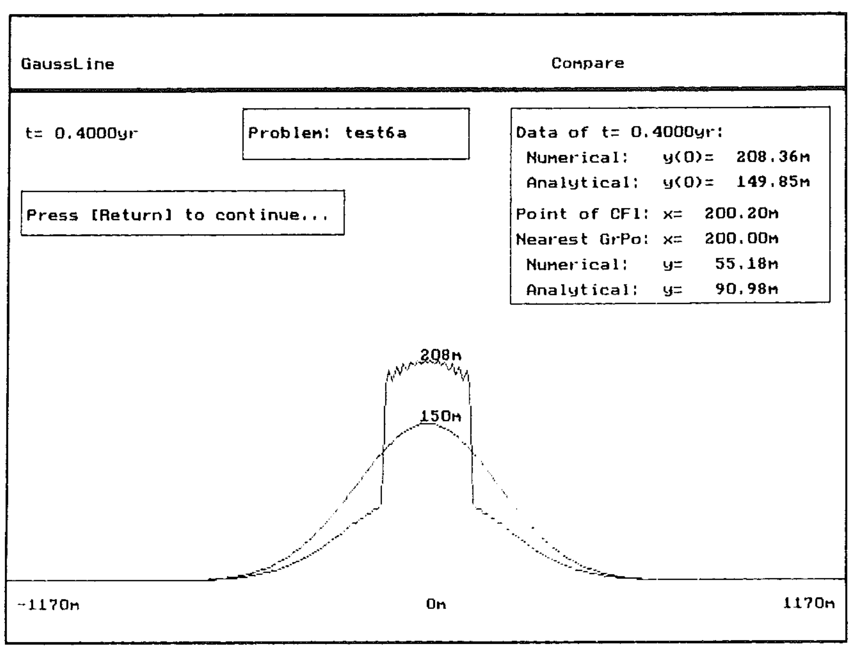
--- GaussLine ---------------- Data for Numerical Computation ----

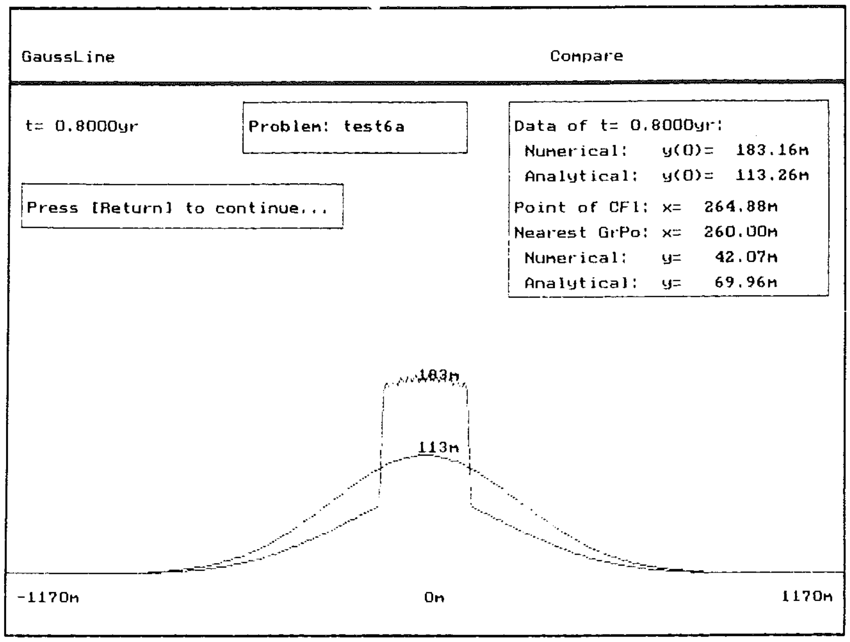
Time step	[yr]	:	0.0050
Grid distance	[m]	:	25.00
Number of points	[-]	:	140

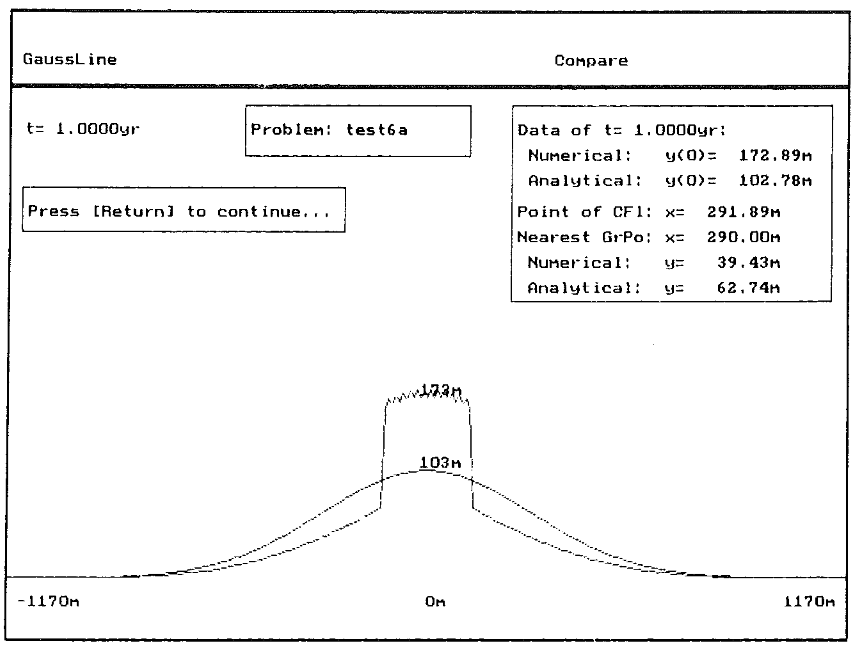
--- GaussLine ---------------------- Results from Computation ----

x-coordinate	shoreline position	
x [m]	y (analytical) [m]	y (numerical) [m]
-100.00	34.14	34.62
-75.00	34.24	34.68
-50.00	34.31	34.80
-25.00	34.35	34.79
0.00	34.37	34.86
25.00	34.35	34.79
50.00	34.31	34.80
75.00	34.24	34.68
100.00	34.14	34.62

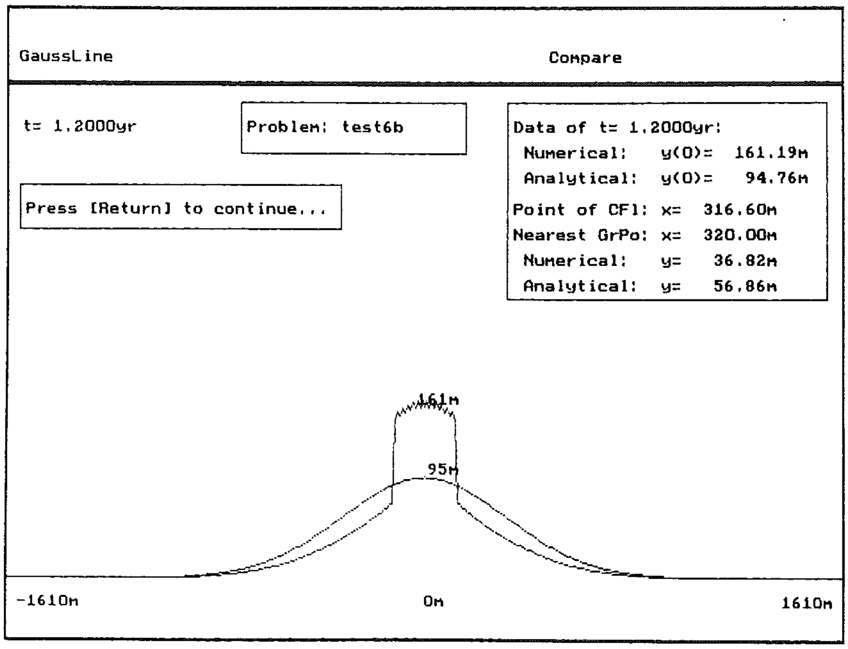

GaussLine
Compare
t= 0.1000yr
Problem: test6a
Press [Return] to continue...
Data of t= 0.1000yr:
Numerical: y(0)= 259.60m
Analytical: y(0)= 226.65m
Point of CFl: x= 132.36m
Nearest GrPo: x= 130.00m
Numerical: y= 124.27m
Analytical: y= 139.92m
260m
227m
-1170m
0m
1170m

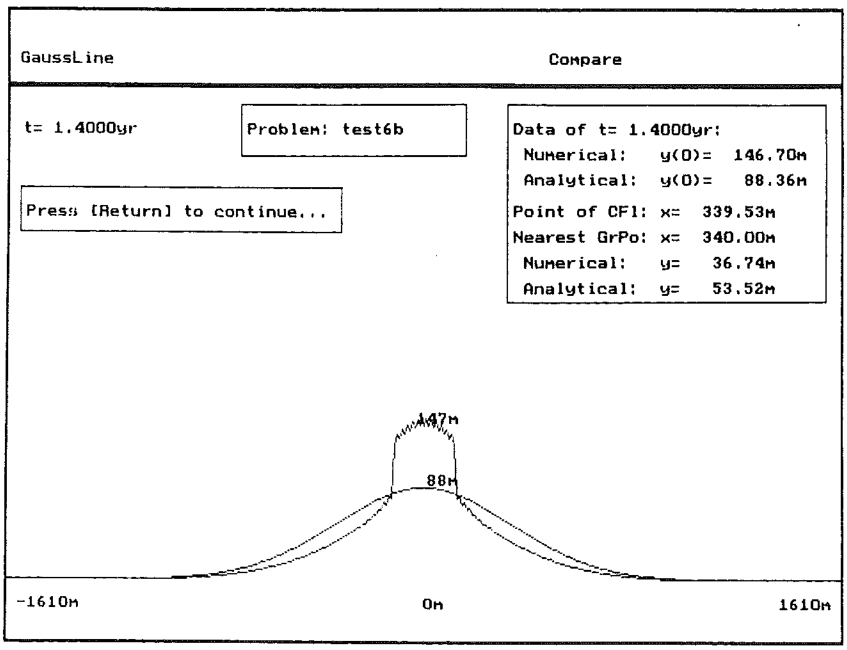

GaussLine
Compare
t= 0.4000yr
Problem: test6a
Press [Return] to continue...
Data of t= 0.4000yr:
Numerical: y(0)= 208.36m
Analytical: y(0)= 149.85m
Point of CFl: x= 200.20m
Nearest GrPo: x= 200.00m
Numerical: y= 55.18m
Analytical: y= 90.98m
208m
150m
-1170m
0m
1170m

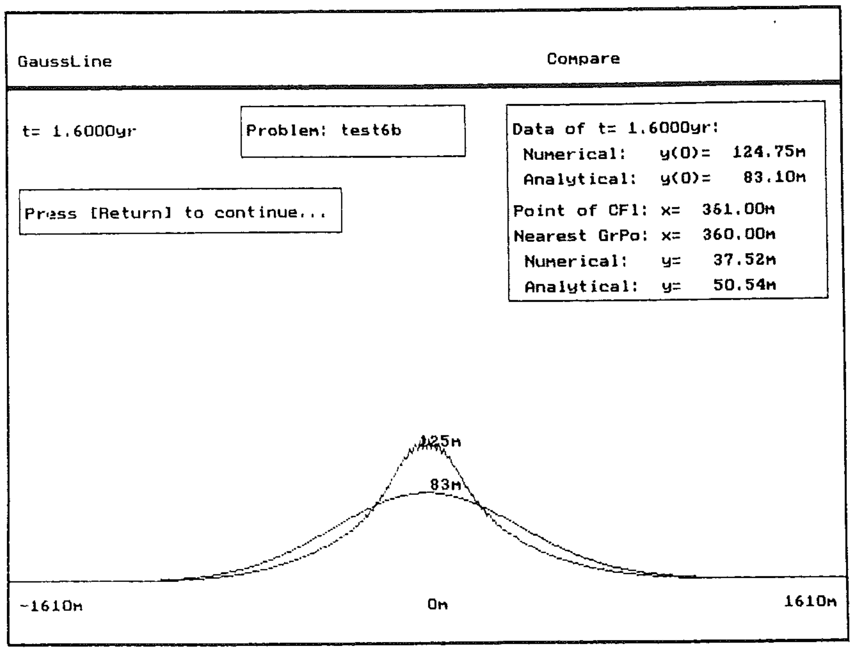

GaussLine
Compare
t= 0.8000yr
Problem: test6a
Data of t= 0.8000yr:
Numerical: y(0)= 183.16m
Analytical: y(0)= 113.26m
Press [Return] to continue...
Point of CFl: x= 264.88m
Nearest GrPo: x= 260.00m
Numerical: y= 42.07m
Analytical: y= 69.96m
183m
113m
-1170m
0m
1170m

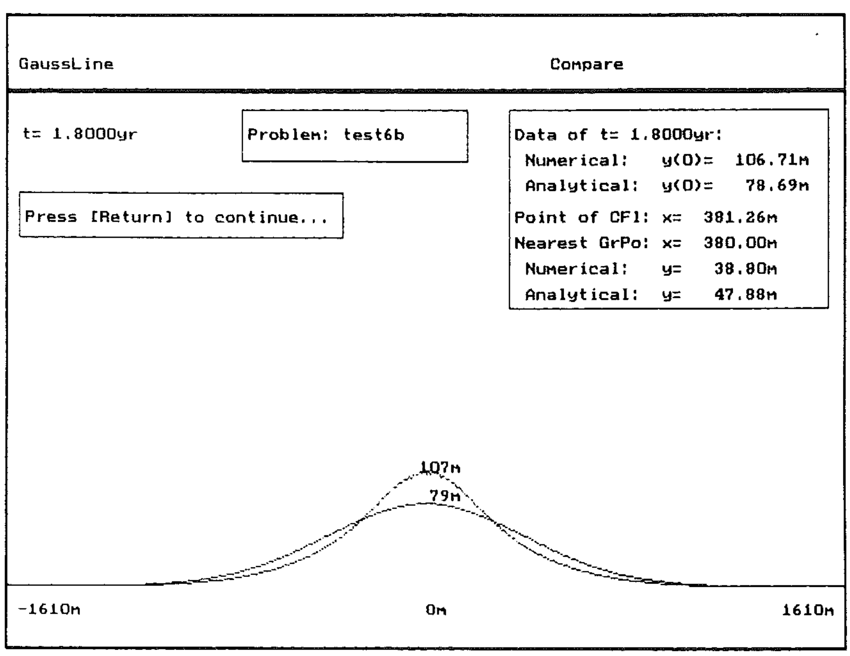

GaussLine
Compare
t= 1.0000yr
Problem: test6a
Data of t= 1.0000yr:
Numerical: y(0)= 172.89m
Analytical: y(0)= 102.78m
Press [Return] to continue...
Point of CFl: x= 291.89m
Nearest GrPo: x= 290.00m
Numerical: y= 39.43m
Analytical: y= 62.74m
173m
103m
-1170m
0m
1170m

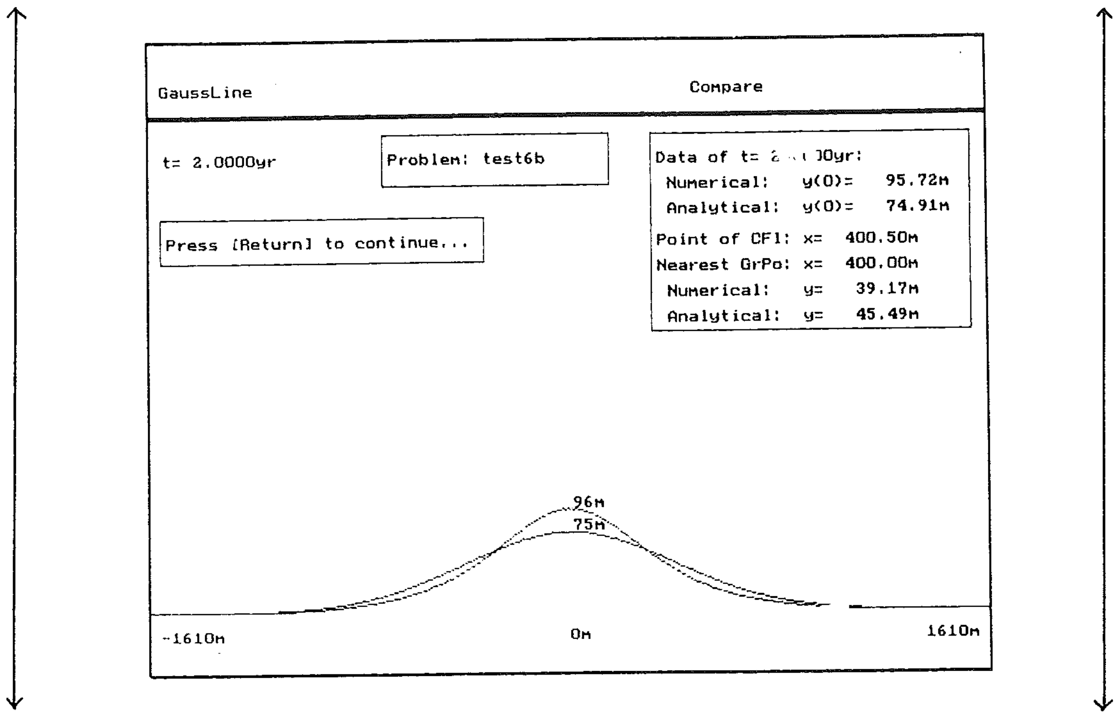
```
--- GaussLine -----------------------------------------------------
                **** Problem: test6a ****


--- GaussLine --------------------------------- Problem Data ----
Period after placement of the fill  [yr]        :        1.00
Volume of the fill                  [m3]        :    1000000
Profile height                      [m]         :       13.30
Coastal constant                    [m3/yr/rad]:    500000.00
Fictive initial period              [yr]        :        0.13
Angle of wave incidence             [rad]       :        0.0000


--- GaussLine --------------- Data for Numerical Computation ----
Time step                           [yr]        :        0.0010
Grid distance                       [m]         :       10.00
Number of points                    [-]         :      117


--- GaussLine --------------------- Results from Computation ----
 x-coordinate                  shoreline position
    x [m]          y (analytical) [m]          y (numerical) [m]
   -100.00              96.92                       165.49
    -90.00              98.01                       173.69
    -80.00              98.99                       168.20
    -70.00              99.87                       176.14
    -60.00             100.63                       170.27
    -50.00             101.28                       177.95
    -40.00             101.82                       171.73
    -30.00             102.24                       179.14
    -20.00             102.54                       172.60
    -10.00             102.72                       179.74
      0.00             102.78                       172.89
     10.00             102.72                       179.74
     20.00             102.54                       172.60
     30.00             102.24                       179.14
     40.00             101.82                       171.73
     50.00             101.28                       177.95
     60.00             100.63                       170.27
     70.00              99.87                       176.14
     80.00              98.99                       168.20
     90.00              98.01                       173.69
    100.00              96.92                       165.49
```


GaussLine
Compare
t= 1.2000yr
Problem: test6b
Press [Return] to continue...
Data of t= 1.2000yr:
Numerical: y(0)= 161.19m
Analytical: y(0)= 94.76m
Point of CFl: x= 316.60m
Nearest GrPo: x= 320.00m
Numerical: y= 36.82m
Analytical: y= 56.86m
161m
95m
-1610m
0m
1610m


GaussLine
Compare
t= 1.4000yr
Problem: test6b
Press [Return] to continue...
Data of t= 1.4000yr:
Numerical: y(0)= 146.70m
Analytical: y(0)= 88.36m
Point of CFl: x= 339.53m
Nearest GrPo: x= 340.00m
Numerical: y= 36.74m
Analytical: y= 53.52m
147m
88m
-1610m
0m
1610m


GaussLine
Compare
t= 1.6000yr
Problem: test6b
Data of t= 1.6000yr:
Numerical: y(0)= 124.75m
Analytical: y(0)= 83.10m
Point of CFl: x= 361.00m
Nearest GrPo: x= 360.00m
Numerical: y= 37.52m
Analytical: y= 50.54m
Press [Return] to continue...
125m
83m
-1610m
0m
1610m


GaussLine
Compare
t= 1.8000yr
Problem: test6b
Data of t= 1.8000yr:
Numerical: y(0)= 106.71m
Analytical: y(0)= 78.69m
Point of CFl: x= 381.26m
Nearest GrPo: x= 380.00m
Numerical: y= 38.80m
Analytical: y= 47.88m
Press [Return] to continue...
107m
79m
-1610m
0m
1610m


GaussLine
Compare
t= 2.0000yr
Problem: test6b
Press [Return] to continue...
Data of t=
Numerical: y(0)= 95.72m
Analytical: y(0)= 74.91m
Point of CF1: x= 400.50m
Nearest GrPo: x= 400.00m
Numerical: y= 39.17m
Analytical: y= 45.49m
96m
75m
-1610m
0m
1610m

--- GaussLine --

**** Problem: test6b ****

--- GaussLine ----------------------------------- Problem Data ----

Period after placement of the fill	[yr]	:	2.00
Volume of the fill	[m3]	:	1000000
Profile height	[m]	:	13.30
Coastal constant	[m3/yr/rad]	:	500000.00
Fictive initial period	[yr]	:	0.13
Angle of wave incidence	[rad]	:	0.0000

--- GaussLine --------------- Data for Numerical Computation ----

Time step	[yr]	:	0.0010
Grid distance	[m]	:	10.00
Number of points	[-]	:	161

--- GaussLine --------------------- Results from Computation ----

x-coordinate	shoreline position	
x [m]	y (analytical) [m]	y (numerical) [m]
-100.00	72.61	89.93
-90.00	73.04	92.36
-80.00	73.43	91.97
-70.00	73.77	94.39
-60.00	74.07	93.59
-50.00	74.33	95.94
-40.00	74.53	94.77
-30.00	74.70	96.99
-20.00	74.81	95.48
-10.00	74.88	97.52
0.00	74.91	95.72
10.00	74.88	97.52
20.00	74.81	95.48
30.00	74.70	96.99
40.00	74.53	94.77
50.00	74.33	95.94
60.00	74.07	93.59
70.00	73.77	94.39
80.00	73.43	91.97
90.00	73.04	92.36
100.00	72.61	89.93

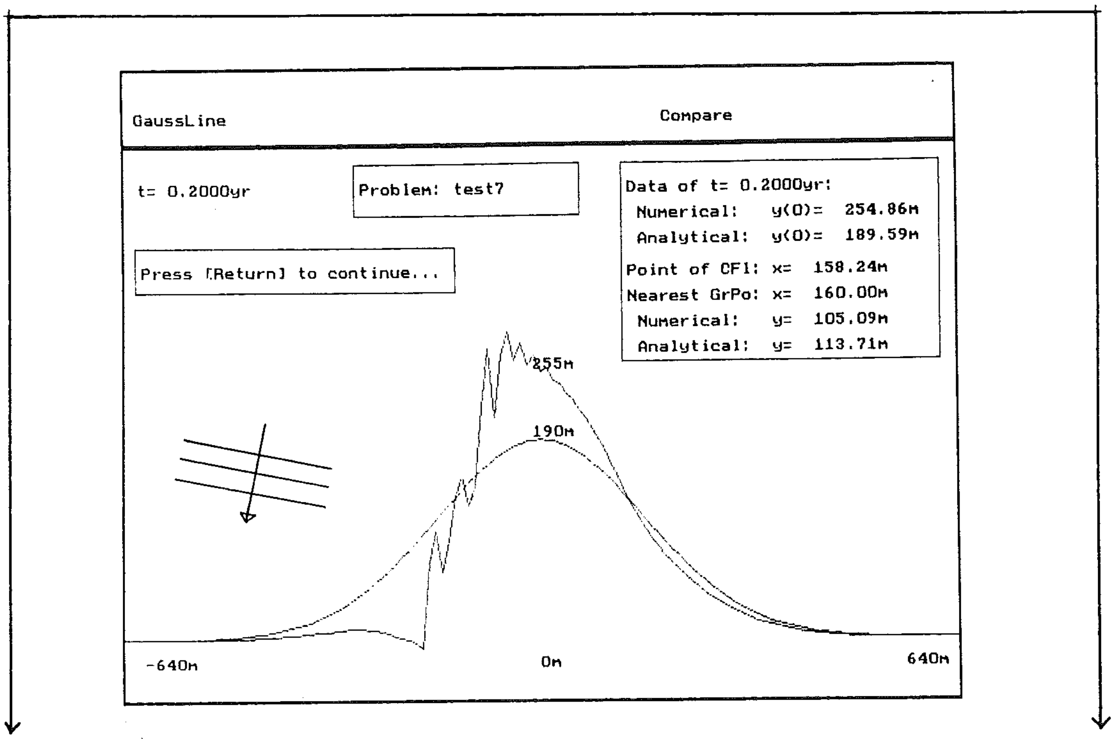
Considering superficially the pictures of test6 one could get the impression that the coastlines look reasonably well. But oscillations that can be noticed on the picture of test6 (t=1.0yr) point to instability of the numerical method.
Therefore test6a has been carried out, hoping to obtain a better impression of the behaviour of the coastline in the first year. From the programme output for test6a a behaviour can be observed that can

be explained. By passing the top of the sinusoidal curve the gradient of the transport capacity changes of sign and also does the transport direction. This explains the block-type shape that can be found from test6a.
However it is not possible to get rid of the oscillations. This can be explained by the same effect of passing the top of the sinusoidal curve. As mentioned before the gradient in transport capacity with respect to the relative angle changes of sign, which using definition (D1.07) corresponds to a change of sign of the coastal constant "s".
The diffusion equation (D1.01) shows that in such a case the diffusion coefficient, the ratio "s/h", becomes negative.
From mathematics it is known that (for positive time direction) the analytical solution of the diffusion equation is not stable in case of a negative diffusion coefficient. Then also numerically it is useless trying to solve it.

Interesting though is the behaviour of the coastline in the period from t=1.0yr and t=2.0yr as shown by test6b. The oscillations remain, but the shoreline seems to recover its original shape of a Gaussian curve. This is probably more a computational effect than that it represents the reality of coasline deformation adequately.

Test7:
Characteristics:
* set(B), thus wide geometry angles
* sinusoidal relation
* $\phi 0$=-10 degrees, thus wide relative angles

This test was originally planned but in fact has lost its value after the experiences of test6, test6a and test6b. Nothing but an instable computation can be expected. This indeed occurs as can be seen from the programme output.



```
--- GaussLine -----------------------------------------------------
                **** Problem: test7 ****


--- GaussLine ---------------------------------- Problem Data ----
Period after placement of the fill  [yr]        :       0.20
Volume of the fill                  [m3]        :    1000000
Profile height                      [m]         :      13.30
Coastal constant                    [m3/yr/rad]:   500000.00
Fictive initial period              [yr]        :       0.13
Angle of wave incidence             [rad]       :    -0.3491


--- GaussLine --------------- Data for Numerical Computation ----
Time step                           [yr]        :     0.0010
Grid distance                       [m]         :      10.00
Number of points                    [-]         :      64


--- GaussLine --------------------- Results from Computation ----
x-coordinate                shoreline position
   x [m]          y (analytical) [m]         y (numerical) [m]
  -100.00             155.27                      145.13
   -90.00             161.27                      221.71
   -80.00             166.84                      276.06
   -70.00             171.91                      211.47
   -60.00             176.44                      269.31
   -50.00             180.35                      292.35
   -40.00             183.62                      265.27
   -30.00             186.21                      281.67
   -20.00             188.08                      261.11
   -10.00             189.21                      270.05
     0.00             189.59                      254.86
    10.00             189.21                      257.33
    20.00             188.08                      245.72
    30.00             186.21                      243.25
    40.00             183.62                      233.47
    50.00             180.35                      227.39
    60.00             176.44                      217.98
    70.00             171.91                      209.20
    80.00             166.84                      199.03
    90.00             161.27                      188.00
   100.00             155.27                      176.25
```

New data

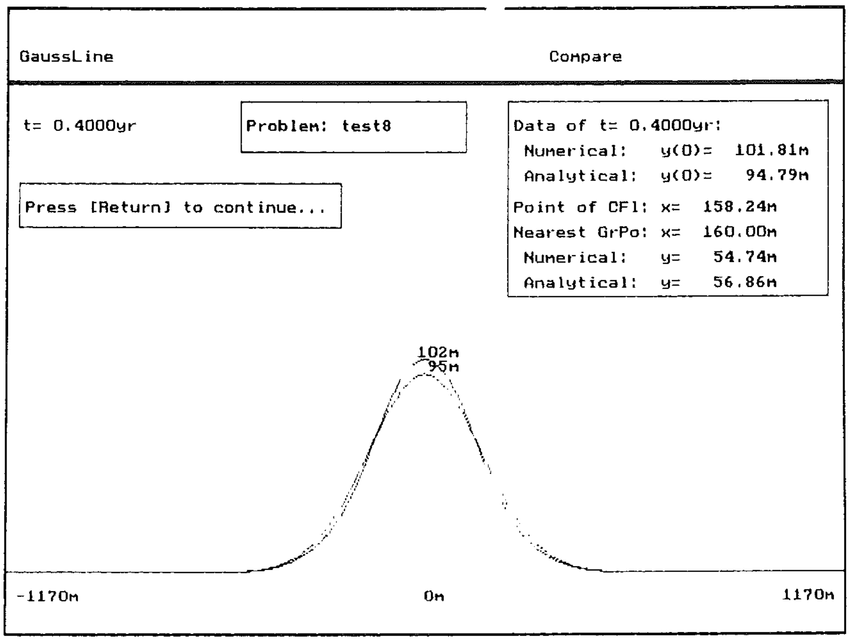
The outcome of especially test6 has led to a revaluation of the problem-cases. It has been decided to add an extra test to investigate the behaviour of a coastline like test6, but only with a maximum relative angle just smaller than 45 degrees. Therefore set(C) has been chosen. The test-cases for the situation of Sylt (under obliquely incidencing waves) have been maintained. However the angle of wave incidence has been chosen smaller so that the relative angles do not pass the 45 degrees limit.
The data is given in Data Table 2,

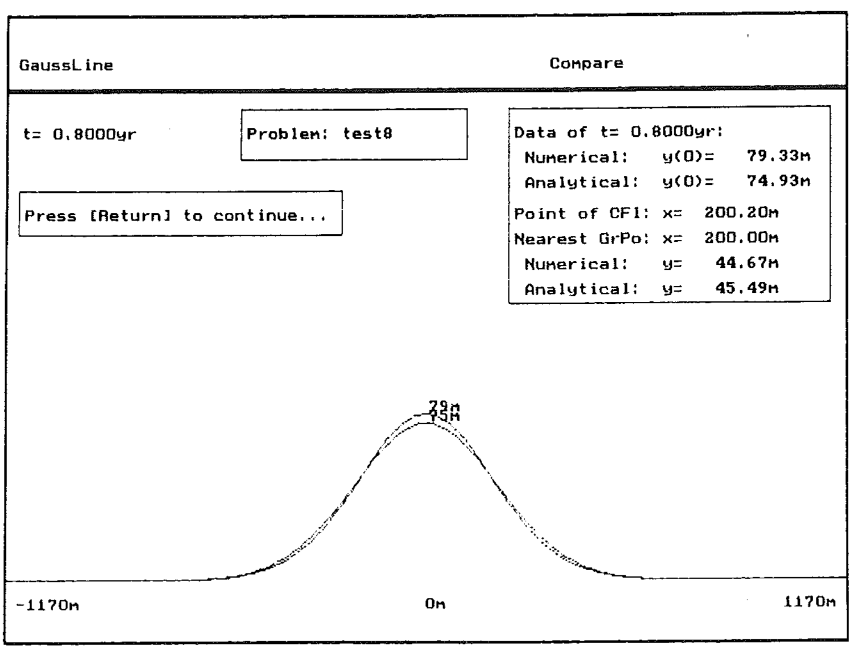
Data Table 2		set(C)	Sylt72	Sylt78
Top of the fill	[m]	150	222	159
Width of the fill	[m]	100	216	289
Volume of the fill	[m^2]	1E6	7.2E5	9.8E5
maximum coastline angle	[deg]	42	31	18
coastal constant	[m^2/yr/rad]	5e4	5e5	5e5

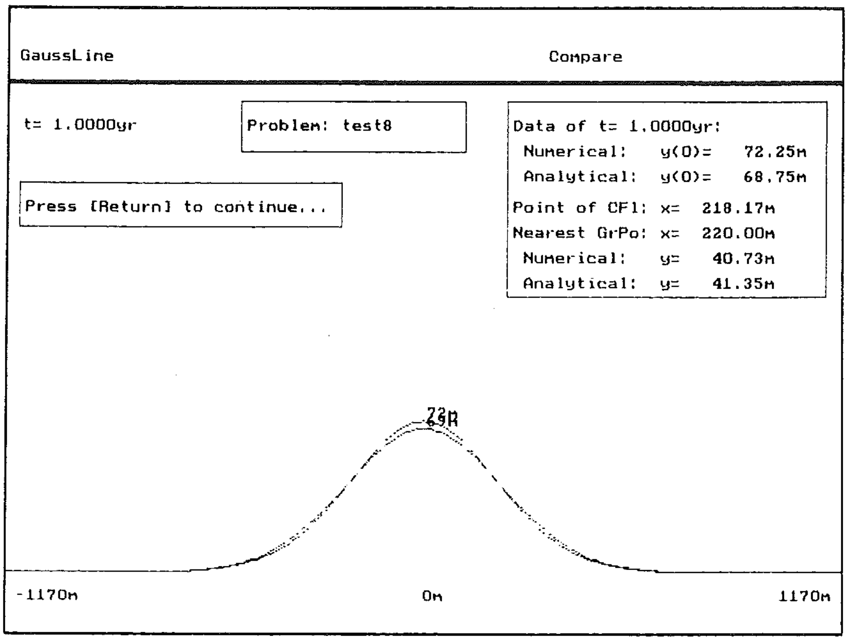
Test-problems, continued

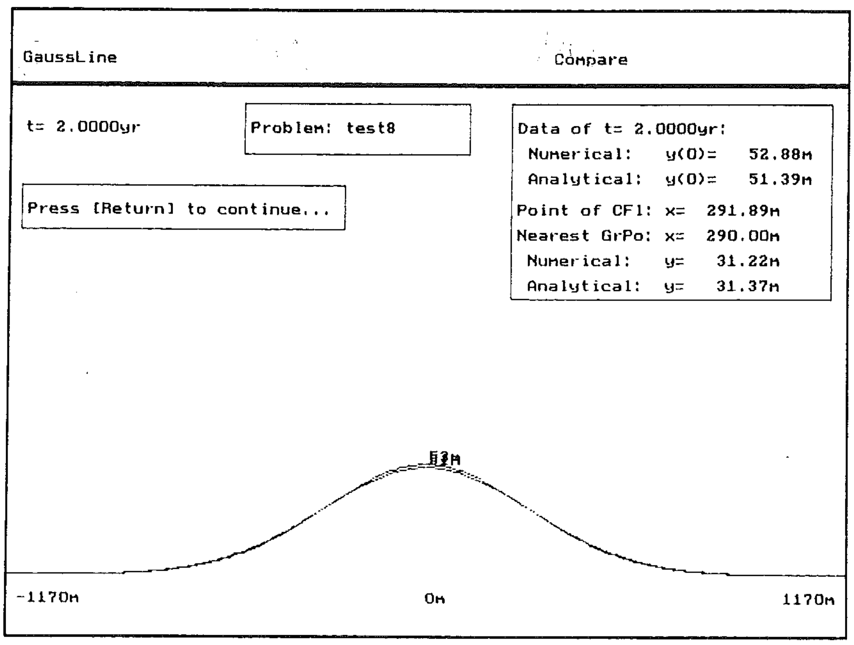
Test8:
Characteristics:
* set(C), thus moderate geometry angles
* sinusoidal relation
* ϕ0=0 degrees, thus moderate relative angles

As mentioned this test has to yield a stable computation, since the maximum coastline angle doe not pass the 45 degrees limit and ϕ0=0 degrees.
The behaviour of this coastline in time is given by the programme output.


GaussLine
Compare
t= 0.4000yr
Problem: test8
Press [Return] to continue...
Data of t= 0.4000yr:
Numerical: y(0)= 101.81m
Analytical: y(0)= 94.79m
Point of CFl: x= 158.24m
Nearest GrPo: x= 160.00m
Numerical: y= 54.74m
Analytical: y= 56.86m
102m
95m
-1170m
0m
1170m


GaussLine
Compare
t= 0.8000yr
Problem: test8
Press [Return] to continue...
Data of t= 0.8000yr:
Numerical: y(0)= 79.33m
Analytical: y(0)= 74.93m
Point of CFl: x= 200.20m
Nearest GrPo: x= 200.00m
Numerical: y= 44.67m
Analytical: y= 45.49m
-1170m
0m
1170m


GaussLine
Compare
t= 1.0000yr
Problem: test8
Press [Return] to continue...
Data of t= 1.0000yr:
Numerical: y(0)= 72.25m
Analytical: y(0)= 68.75m
Point of CFl: x= 218.17m
Nearest GrPo: x= 220.00m
Numerical: y= 40.73m
Analytical: y= 41.35m
-1170m
0m
1170m

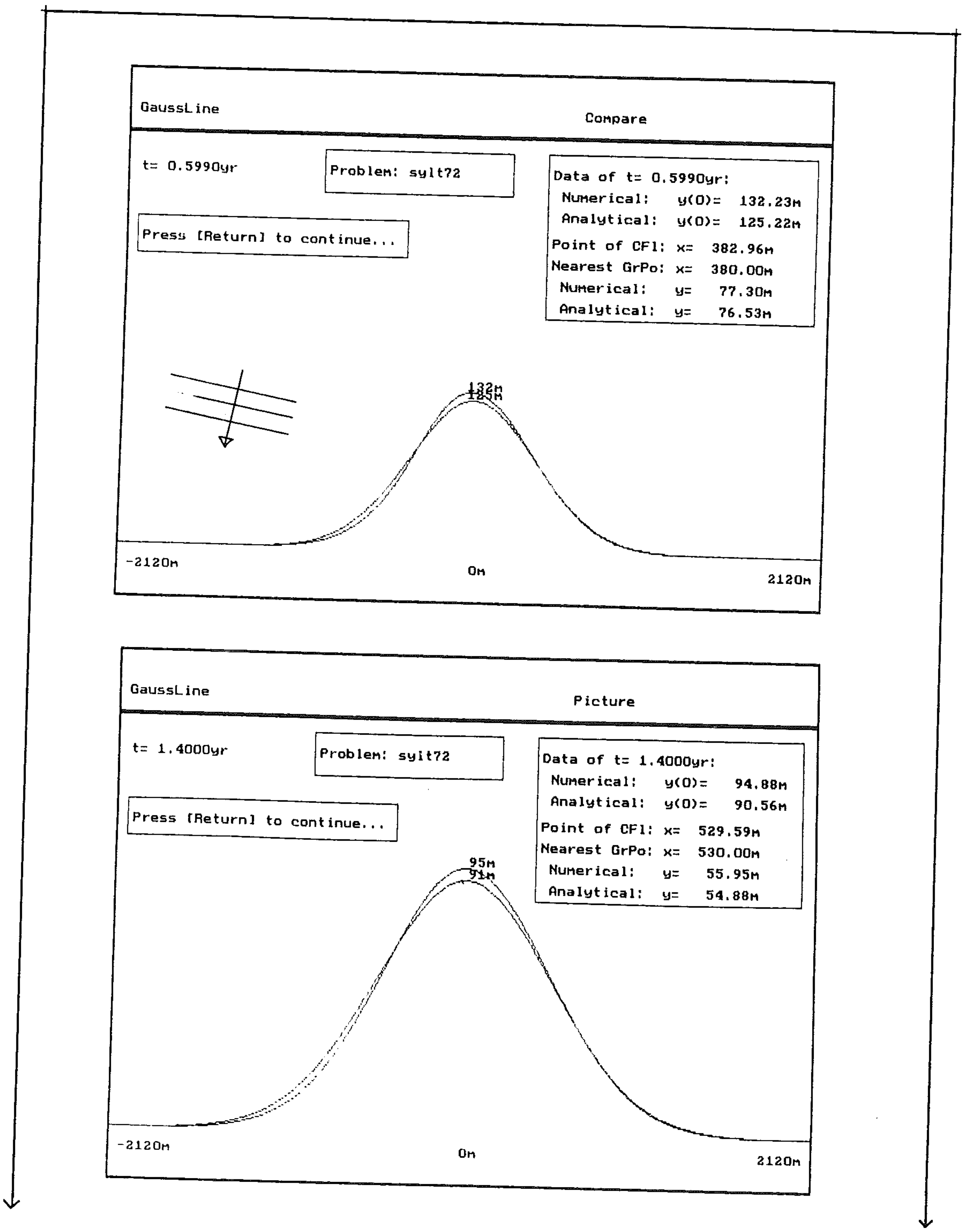

GaussLine
Compare
t= 2.0000yr
Problem: test8
Press [Return] to continue...
Data of t= 2.0000yr:
Numerical: y(0)= 52.88m
Analytical: y(0)= 51.39m
Point of CFl: x= 291.89m
Nearest GrPo: x= 290.00m
Numerical: y= 31.22m
Analytical: y= 31.37m
-1170m
0m
1170m

```
--- GaussLine ---------------------------------------------------
               **** Problem: test8 ****


--- GaussLine ----------------------------------- Problem Data ----
Period after placement of the fill  [yr]        :          2.00
Volume of the fill                  [m3]        :      1000000
Profile height                      [m]         :         26.60
Coastal constant                    [m3/yr/rad]:      500000.00
Fictive initial period              [yr]        :          0.27
Angle of wave incidence             [rad]       :          0.0000


--- GaussLine --------------- Data for Numerical Computation ----
Time step                           [yr]        :          0.0010
Grid distance                       [m]         :         10.00
Number of points                    [-]         :        117


--- GaussLine --------------------- Results from Computation ----
x-coordinate                  shoreline position
   x [m]          y (analytical) [m]          y (numerical) [m]
  -100.00              48.46                       49.69
   -90.00              49.00                       50.28
   -80.00              49.50                       50.81
   -70.00              49.93                       51.29
   -60.00              50.32                       51.71
   -50.00              50.64                       52.06
   -40.00              50.91                       52.35
   -30.00              51.12                       52.58
   -20.00              51.27                       52.75
   -10.00              51.36                       52.84
     0.00              51.39                       52.88
    10.00              51.36                       52.84
    20.00              51.27                       52.75
    30.00              51.12                       52.58
    40.00              50.91                       52.35
    50.00              50.64                       52.06
    60.00              50.32                       51.71
    70.00              49.93                       51.29
    80.00              49.50                       50.81
    90.00              49.00                       50.28
   100.00              48.46                       49.69
```

As can be seen from the pictures this test-problem behaves rather well. No stability problems occur. The numerically computed coastline protrudes more seaward as already explained in test5a.
Another effect can be noticed also. When a long computation period is chosen the numerical and analytical solution tend to come together again. This can be explained by the decreasing coastline angles in time. When the curve is sufficiently flattened it will follow the analytical solution again. This is only valid for symmetrical problems.

Sylt72:
Characteristics:
* Sylt72, thus moderate geometry angles
* sinusoidal relation
* ϕ0=-10 degrees, thus moderate relative angles

Originally a value of ϕ0=39 degrees should be used in this test but the model forces a smaller angle of wave incidence.

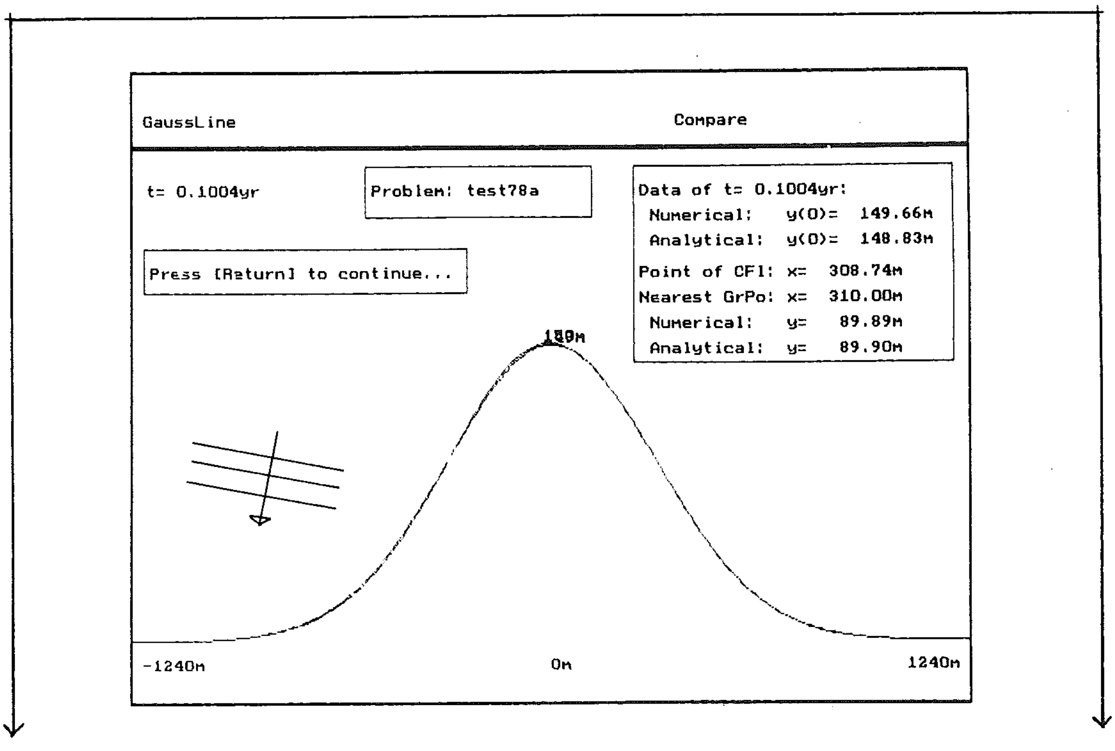


```
--- GaussLine ---------------------------------------------------
               **** Problem: sylt72 ****


--- GaussLine ---------------------------------- Problem Data ----
Period after placement of the fill  [yr]         :        1.40
Volume of the fill                  [m3]         :      720000
Profile height                      [m]          :        5.99
Coastal constant                    [m3/yr/rad]:      500000.00
Fictive initial period              [yr]         :        0.28
Angle of wave incidence             [rad]        :       -0.1745


--- GaussLine --------------- Data for Numerical Computation ----
Time step                           [yr]         :        0.0003
Grid distance                       [m]          :       10.00
Number of points                    [-]          :       212


--- GaussLine --------------------- Results from Computation ----
 x-coordinate                  shoreline position
    x [m]          y (analytical) [m]          y (numerical) [m]
   -100.00              88.95                         93.05
    -90.00              89.26                         93.40
    -80.00              89.53                         93.71
    -70.00              89.77                         93.99
    -60.00              89.98                         94.23
    -50.00              90.15                         94.43
    -40.00              90.30                         94.59
    -30.00              90.41                         94.72
    -20.00              90.49                         94.81
    -10.00              90.54                         94.86
      0.00              90.56                         94.88
     10.00              90.54                         94.86
     20.00              90.49                         94.80
     30.00              90.41                         94.71
     40.00              90.30                         94.58
     50.00              90.15                         94.41
     60.00              89.98                         94.21
     70.00              89.77                         93.97
     80.00              89.53                         93.69
     90.00              89.26                         93.39
    100.00              88.95                         93.04
```

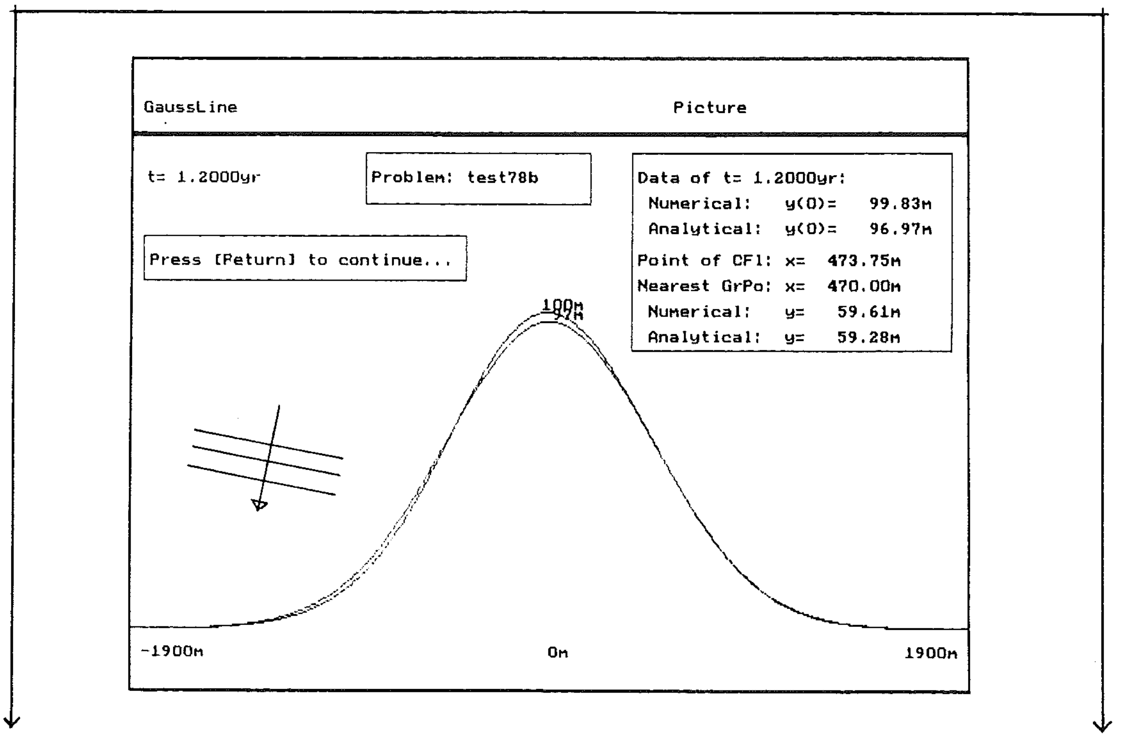
The programme output shows the position of the coastline at t=0.6yr and t=1.4yr, which corresponds to the measured series 72.t2 and 72.t3 (see, Section D2.1). From the quantitative output it can be seen that in this situation differences in coastline position occur of about 5m at t=1.4yr.

Test78a and test78b:
Characteristics:
* Sylt78, thus small geometry angles
* sinusoidal relation
* ϕ0=-10 degrees, thus moderate relative angles

For this test-problem, which better had been named Sylt78, the same remarks are valid as for the Sylt72 problem.



```
--- GaussLine --------------------------------------------------------
                  **** Problem: test78a ****


--- GaussLine ---------------------------------- Problem Data ----
Period after placement of the fill  [yr]        :         0.10
Volume of the fill                  [m3]        :       980000
Profile height                      [m]         :         8.51
Coastal constant                    [m3/yr/rad]:     500000.00
Fictive initial period              [yr]        :         0.71
Angle of wave incidence             [rad]       :      -0.1745


--- GaussLine --------------- Data for Numerical Computation ----
Time step                           [yr]        :       0.0004
Grid distance                       [m]         :        10.00
Number of points                    [-]         :       124


--- GaussLine --------------------- Results from Computation ----
 x-coordinate                     shoreline position
    x [m]          y (analytical) [m]            y (numerical) [m]
   -100.00              141.23                        142.90
    -90.00              142.64                        144.25
    -80.00              143.92                        145.45
    -70.00              145.06                        146.50
    -60.00              146.05                        147.41
    -50.00              146.89                        148.16
    -40.00              147.59                        148.77
    -30.00              148.13                        149.22
    -20.00              148.52                        149.52
    -10.00              148.76                        149.67
      0.00              148.83                        149.66
     10.00              148.76                        149.51
     20.00              148.52                        149.21
     30.00              148.13                        148.75
     40.00              147.59                        148.16
     50.00              146.89                        147.41
     60.00              146.05                        146.53
     70.00              145.06                        145.50
     80.00              143.92                        144.33
     90.00              142.64                        143.03
    100.00              141.23                        141.60
```


GaussLine
Picture
t= 1.2000yr
Problem: test78b
Press [Return] to continue...
Data of t= 1.2000yr:
Numerical: y(0)= 99.83m
Analytical: y(0)= 96.97m
Point of CFl: x= 473.75m
Nearest GrPo: x= 470.00m
Numerical: y= 59.61m
Analytical: y= 59.28m
100m
97m
-1900m
0m
1900m

```
--- GaussLine ---------------------------------------------------------
                **** Problem: test78b ****

--- GaussLine ---------------------------------- Problem Data ----
Period after placement of the fill  [yr]        :        1.20
Volume of the fill                  [m3]        :     980000
Profile height                      [m]         :        8.51
Coastal constant                    [m3/yr/rad]:    500000.00
Fictive initial period              [yr]        :        0.71
Angle of wave incidence             [rad]       :       -0.1745

--- GaussLine ---------------- Data for Numerical Computation ----
Time step                           [yr]        :        0.0004
Grid distance                       [m]         :       10.00
Number of points                    [-]         :      190

--- GaussLine ---------------------- Results from Computation ----
 x-coordinate                     shoreline position
    x [m]          y (analytical) [m]          y (numerical) [m]
   -100.00              94.84                       97.88
    -90.00              95.24                       98.28
    -80.00              95.60                       98.64
    -70.00              95.92                       98.95
    -60.00              96.20                       99.22
    -50.00              96.44                       99.44
    -40.00              96.63                       99.61
    -30.00              96.78                       99.73
    -20.00              96.89                       99.81
    -10.00              96.95                       99.84
      0.00              96.97                       99.83
     10.00              96.95                       99.77
     20.00              96.89                       99.66
     30.00              96.78                       99.51
     40.00              96.63                       99.31
     50.00              96.44                       99.07
     60.00              96.20                       98.78
     70.00              95.92                       98.45
     80.00              95.60                       98.07
     90.00              95.24                       97.65
    100.00              94.84                       97.19
```

Again an asymmetrical deformation can be seen from the output of the programme. Quantitatively the differences amount to about 2m (78.t2) and about 4m (78.t3).

Conclusion

It can be concluded that the expected asymmetrical deformation of a Gaussian curved coastline under obliquely incidencing waves can be represented by this model, using a sinusoidal relation between relative angle and sand transport capacity.

However from the test-problems it becomes clear also that a reliable result can only be obtained, if the relative angle does not exceed the limit of 45 degrees.

A mathematical explanation for this limitation can be found, but its physical meaning is not directly obvious.

4 Programme listing

```
--- GaussLine ------------------------------------------ READ.ME ----

This floppy disk contains the programme facilities of the GaussLine
programme, which has been designed to compute analytically and
numerically the behaviour of a beachfill with the shape of a Gaussian
curve.
For more information on the theoretical background see the programme
documentation "Computer programme GaussLine".

The GaussLine programme is available one two floppy disks. The first
one (this disk) contains the programme facilities. The second one
contains the datafiles of the tests descibed in the documentation.

It is most convenient to insert the programme floppy in drive B: and
and the data floppy in drive A:.
When a hard disk (C:) is available it can be advised to copy all
files of this programme floppy to a subdirectory on the hard disk.
Then the data floppy can be used from drive A: also.

To start the programme (thus from B: or from C:,subdirectory) the
command  GAUSS  can be typed when the Personal Computer has a 8087
mathematical coprocessor. If a coprocessor is not available the
command  GLINE  has to be used.

The programme autodetects with which type of graphical card the PC is
equipped. Only a ATT400 graphical card can not be autodetected and
therefore has to be specified in the file  GSLCFG.DEF. This file can
be found on this programme floppy. There it can also be specified if
another datadrive (instead of A:) has to be used by the programme.

Description of the other files on this programme floppy:
  GLINE.EXE     --   Programme executable  (no 8087 coprocessor)
  GAUSS.EXE     --   Programme executable  (8087 coprocessor)
  GSLNUM.DEF    --   Default file numerical data
  GSLCFG.DEF    --   Default file configuration of PC
  GSLEXP.TXT    --   Text file explanation
  GSLFLG.DAT    --   Data file flagpage
  GRAPH.TPU     --   Turbo Pascal 4.0 graphical unit
  *****.BGI     --   Several graphics drivers

--- GaussLine -------------------------------------------------------
```

```
{--- GaussLine -------------------------------------------- BEGIN ----}
{ This programme computes coastline positions according to the one-   }
{ line schemetization. The outcome of an analytical method using a    }
{ linear relation between the angle of wave incidence and the sand    }
{ transport capacity can be compared with a numerical method using    }
{ a sinusoidal relation.                                              }

{--- GausLine ---------------------------------------------------------}

PROGRAM  GaussLine;

USES    Crt, DOS, Graph, Printer;                    { turbo units    }

CONST   nmax    =  1000;                             { array size     }

TYPE    CLine   =  ARRAY [ -nmax.. nmax ] OF real;   { coastline      }
        FName   =  string[  9];                      { file name      }
        OName   =  string[ 30];                      { option name    }

VAR     {        ****  Problem Variables  ****      }
        dx      : REAL;      { x-grid distance          }
        dt      : REAL;      { time step                }
        h       : REAL;      { profile height           }
        npts    : INTEGER;   { number of grid points    }
        nstep   : INTEGER;   { number of time steps     }
        PbName  : FName;     { problem name             }
        phi0    : REAL;      { angle of wave incidence  }
        s       : REAL;      { coastal constant         }
        t       : REAL;      { time                     }
        Ti      : REAL;      { initial period           }
        te      : REAL;      { computation period       }
        Tf      : REAL;      { top of the fill at t=0   }
        t0      : REAL;      { starting time            }
        Vf      : REAL;      { volume of the fill       }
        Wf      : REAL;      { width of the fill at t=0 }
        x       : REAL;      { x-coordinate             }
        yao     : CLine;     { old shoreline analytical }
        yan     : CLine;     { new shoreline analytical }
        yno     : CLine;     { old shoreline numerical  }
        ynn     : CLine;     { new shoreline numerical  }

VAR     {        ****  Auxilary Variables  ****     }
        DD      : string[2];{ data drive               }
        eWf     : REAL;      { width of the fill at t=te }
        flag    : BOOLEAN;   { flagpage                 }
        iview   : INTEGER;   { interval screen i/o      }
        lin     : BOOLEAN;   { linear or sinusoidal     }
        opt     : CHAR;      { option                   }
        quit    : BOOLEAN;   { quit programme           }
        ynd     : CLine;     { last drawn (num) sh-line }

VAR     {        ****  Graphics Variables  ****     }
        GraphDr: INTEGER;    { graphics driver          }
        GraphMo: INTEGER;    { graphics mode            }
        XScale : REAL;       { x-scale factor           }
        Xsmax  : INTEGER;    { total number of x-pixels }
```

```
       XZero  : INTEGER;  { x-zero axis                 }
       YScale : REAL;     { y-scale factor              }
       Ysmax  : INTEGER;  { total number of y-pixels    }
       YZero  : INTEGER;  { y-zero axis                 }

{      ****  Include Libraries  ****                    }
{$I a:TXTPROC.LIB}       { tools text lay-out           }
{$I a:ANAPROC.LIB}       { tools analytical solution    }
{$I a:NUMPROC.LIB}       { tools numerical solution     }
{$I a:GRFPROC.LIB}       { tools graphics lay-out       }
{$I a:I-OPROC.LIB}       { tools input/output           }
{$I a:INIPROC.LIB}       { initation options            }
{$I a:OPTPROC.LIB}       { programme options            }


BEGIN
   Text_screen( 'normal');
   Config_defaults;
   Initiate_graphics;
   Draw_flag;
   RestoreCrtMode;
   quit:= FALSE;
   REPEAT
     ClrScr;
     SkLn( 3);
     Top_title( 'Main Menu');
     SkLn( 3); Blank( 6);
     invers( ' E '); write( '  Explanation');
     SkLn( 2); Blank( 6);
     invers( ' A '); write( '  Analytical Solution');
     SkLn( 2); Blank( 6);
     invers( ' C '); write( '  Compare Numerical and Analytical Solution'
     SkLn( 2); Blank( 6);
     invers( ' P '); write( '  Picture Result of Computation');
     SkLn( 2); Blank( 6);
     invers( ' Q '); write( '  Quit GaussLine Programme Execution');
     SkLn( 3); Blank( 6);
     Write( '       Select the Option of Your Choice  >>> ');
     opt:= readkey; write( opt);
     CASE opt OF
       'E', 'e'  : Explain;
       'A', 'a'  : Analyt;
       'C', 'c'  : Compare;
       'P', 'p'  : Picture;
       'Q', 'q'  : quit:= true;
     END;
   UNTIL quit;
   CloseGraph;
   ClrScr; SkLn( 2); Top_title( 'End'); SkLn( 1);
   Blank( 3); Writeln( ' End of GaussLine Programme Execution.');
   Blank( 3); Writeln( ' Have a nice day.');
END.

{--- GausLine ---------------------------------------------- END ----}
```

```
{--- TXTPROC.LIB ---------------------------------------- BEGIN ----}
{ This library contains some tools for writing messages to the text }
{ screen.                                                           }

{--- TXTPROC.LIB ------------------------------------ CONTENTS ----}
{ PROC  Text_screen     :  Text screen initiation                   }
{ PROC  invers          :  Reverse video                            }
{ PROC  SkLn            :  New lines                                }
{ PROC  Blank           :  Spaces                                   }
{ PROC  Top_title       :  Title line                               }
{ PROC  Explain_screen  :  Screen for option Explain                }

{--- TXTPROC.LIB ---------------------------------------------------}

TYPE   Str12 = string[ 12];
       Str70 = string[ 70];

PROCEDURE  Text_Screen( Mode : Str12);
  BEGIN
    TextColor( Yellow);
    TextBackground( Black);
    IF mode = 'reverse' THEN
    BEGIN
      TextColor( Black);
      TextBackground( Yellow);
    END;
  END;

PROCEDURE  invers( Message : Str70);
  BEGIN
    Text_Screen( 'reverse');
    write( Message);
    Text_Screen( 'normal');
  END;

PROCEDURE  SkLn( n : INTEGER);
  VAR i : INTEGER;
  BEGIN
    FOR i:= 1 TO n DO writeln;
  END;

PROCEDURE  Blank( n : INTEGER);
  VAR i : INTEGER;
  BEGIN
    FOR i:=1 TO n DO write( ' ');
  END;

PROCEDURE  Top_title( option : str12);
  VAR i : INTEGER;
  BEGIN
    Blank( 2);
    write( '--- ');
    write( 'GaussLine');
    write( ' ');
    For i:=1 TO ( 39- Length( option)) DO write( '-');
    write( ' ');
```

```
    write( option);
    writeln( ' ----');
  END;

PROCEDURE  Explain_screen( i : INTEGER);
  BEGIN
    ClrScr;
    SkLn( 1); Top_title( 'Explain');
    Blank( 57); write( 'PAGE ', i: 2);
    SkLn( 1);
  END;

{--- TXTPROC.LIB ------------------------------------------ END ----}
```

```
{--- ANAPROC.LIB ------------------------------------------ BEGIN ----}
{ This library contains some tools for the computation of the        }
{ analytical solution of the diffusion equation  for the specific    }
{ initial condition of a stock-pile type beachfill.                  }

{--- ANAPROC.LIB --------------------------------------- CONTENTS ----}
{ FUNC  CA     : Time dependent coefficient                          }
{ FUNC  CB     : Time dependent coefficient                          }
{ PROC  Anly   : Construction of the analytical solution             }

{--- ANAPROC.LIB -----------------------------------------------------}

FUNCTION  CA( t: REAL) : REAL;
  BEGIN
    CA:= Vf/ sqrt( 4* Pi* h* s*( t+ Ti));
  END;

FUNCTION  CB( t: REAL) : REAL;
  BEGIN
    CB:= h/( 4* s*( t+ Ti));
  END;

PROCEDURE  Anly( VAR ya: CLine;
                     t : REAL;
                     np: INTEGER);
  VAR  i : INTEGER;
  BEGIN
    x:= 0;
    FOR i:= 0 TO np DO
    BEGIN
      IF x> 4* sqrt( 0.5/ CB( t)) THEN ya[ i]:= 0
        ELSE ya[ i]:= CA( t)* exp( -CB( t)* x* x);
      ya[-i]:= ya[ i];
      x:= x+ dx;
    END;
  END;

{--- ANAPROC.LIB ------------------------------------------ END ----}
```

```
{--- NUMPROC.lIB ------------------------------------------ BEGIN ----}
{ This library contains some tools for the computation of the        }
{ numerical solution of the diffusion equation as it appears in      }
{ shoreline computations (one-line model).                           }

{--- NUMPROC.LIB --------------------------------------- CONTENTS ----}
{ FUNC  dydx             :  Coastline direction                      }
{ FUNC  phi0r            :  Relative angle of wave incidence         }
{ FUNC  Sl_lin           :  Longshore sand transport linear relation }
{ FUNC  Sl_sin           :  Longshore transport sinusoidal relation  }
{ PROC  FTCS_lin         :  Forward time central space using Sl_lin  }
{ PROC  FTCS_sin         :  Forward time central space using Sl_sin  }
{ PROC  Stability_check  :  Check on numerical stability             }

{--- NUMPROC.LIB -----------------------------------------------------}

FUNCTION  dydx( VAR y : CLine;
                    i : INTEGER) : REAL;
  BEGIN
    dydx:= ( y[ i+ 1]- y[ i- 1])/( 2* dx);
  END;

FUNCTION  phi0r( VAR y : CLine;
                     i : INTEGER) : REAL;
  BEGIN
    phi0r := arctan( dydx( y, i))- phi0;
  END;

FUNCTION  Sl_lin( VAR y : CLine;
                      i : INTEGER) : REAL;
  BEGIN
    Sl_lin:= -s* dydx( y, i);
  END;

FUNCTION  Sl_sin( VAR y : CLine;
                      i : INTEGER) : REAL;
  BEGIN
    Sl_sin:= -0.5* s* sin( 2* phi0r( y, i));
  END;

PROCEDURE  FTCS_lin( VAR  yn : CLine;
                     VAR  yo : CLine;
                          np : INTEGER);
  VAR  i : INTEGER;
  BEGIN
    FOR i:= -np TO np DO
      yn[ i]:= yo[ i]- dt/( 2* h* dx)*
               ( Sl_lin( yo, i+ 1)- Sl_lin( yo, i- 1));
  END;

PROCEDURE  FTCS_sin( VAR  yn : CLine;
                     VAR  yo : CLine;
                          np : INTEGER);
  VAR  i : INTEGER;
  BEGIN
    FOR i:= -np TO np DO
```

```
    yn[ i]:= yo[ i]- dt/( 2* h* dx)*
            ( Sl_si1( yo, i+ 1)- Sl_sin( yo, i- 1));
  END;

PROCEDURE  Stability_check( VAR dt     : REAL;
                                method : OName);
  VAR crit  : REAL;
      ch    : CHAR;
  BEGIN
    IF method = 'FTCS' THEN
    BEGIN
      crit:= 0.5* h/ s* sqr( dx);
      IF dt> crit THEN
      BEGIN
        SkLn( 2);
        Top_Title( 'Stability');
        Blank( 2); invers( 'warning:');
        writeln( ' Time Step too big');
        dt:= 0.5* crit;
        Blank( 2); writeln( 'Standard corrective action has been taken.')
        Blank( 2); writeln( 'New time step: ', dt: 8: 5);
        Blank( 2); write( 'Press [Return] to continue... '); ch:= readkey
      END;
    END;
  END;

{--- NUMPROC.LIB ------------------------------------------ END ----}
```

```
{--- GRFPROC.LIB ----------------------------------------- BEGIN ----}
{ This library contains some tools for writing text to and drawing   }
{ pictures on the graphical screen.                                  }

{--- GRFPROC.LIB -------------------------------------- CONTENTS ----}
{ FUNC  GSPX              :  Graphical x-coordinate (pixel number)  }
{ FUNC  GSPY              :  Graphical y-coordinate (pixel number)  }
{ PROC  Ana_display       :  Display data analytical solution       }
{ PROC  Num_display       :  Display data numerical solution        }
{ PROC  Time_display      :  Display running time                   }
{ PROC  Graph_window      :  Draw text window on graphical screen   }
{ PROC  Clear_window      :  Clear window from graphical screen     }
{ PROC  Initiate_graphics :  Initiation of graphical system         }
{ PROC  Draw_boundary     :  Draw GaussLine window                  }
{ PROC  Draw_coastline    :  Draw computed coastline                }
{ PROC  Del_coastline     :  Erase drawn coastline                  }

{--- GRFPROC.LIB ----------------------------------------------------}

FUNCTION  GSPX( frac :REAL) : INTEGER;
  BEGIN
    GSPX:= Round( XsMax* frac);
  END;

FUNCTION  GSPY( frac :REAL) : INTEGER;
  BEGIN
    GSPY:= Round( YsMax* frac);
  END;

PROCEDURE  Ana_display;
  VAR TStr : String[ 7];
      XCf  : REAL;
      XStr : string[ 8];
      YCf  : REAL;
      YStr : string[ 8];
  BEGIN
    Str( yao[ 0]: 8: 2, YStr);
    OutTextXY( 5, GSPY( 0.05), 't= 0.0000yr:  y(0)='+ YStr+ 'm');
    XCf:= sqrt( 0.5/ CB( 0));
    Str( XCf: 8: 2, XStr);
    OutTextXY( 5, GSPY( 0.09), 'Point of CFl: x='+ XStr+ 'm');
    YCf:= CA( 0)* exp( -CB( 0)* XCf* XCf);
    Str( YCf: 8: 2, YStr);
    OutTextXY( 5, GSPY( 0.13), '              y='+ YStr+ 'm');
    Str( yan[ 0]: 8: 2, YStr);
    Str( te: 7: 4, TStr);
    OutTextXY( 5, GSPY( 0.21), 't='+ TStr+ 'yr:  y(0)='+ YStr+ 'm');
    XCf:= sqrt( 0.5/ CB( te));
    Str( XCf: 8: 2, XStr);
    OutTextXY( 5, GSPY( 0.25), 'Point of CFl: x='+ XStr+ 'm');
    YCf:= CA( te)* exp( -CB( te)* XCf* XCf);
    Str( YCf: 8: 2, YStr);
    OutTextXY( 5, GSPY( 0.29), '              y='+ YStr+ 'm');
  END;
```

```
PROCEDURE  Num_display;
  VAR NCf   : INTEGER;
      XCf   : REAL;
      TStr  : string[ 7];
      XcStr : string[ 8];
      YaStr : string[ 8];
      YnStr : string[ 8];
  BEGIN
    Str( t: 7: 4, TStr);
    OutTextXY( 5, GSPY( 0.05), 'Data of t='+ TStr+ 'yr:');
    Str( ynn[ 0]: 8: 2, YnStr);
    OutTextXY( 5, GSPY( 0.09), ' Numerical:   y(0)='+ YnStr+ 'm');
    Str( yan[ 0]: 8: 2, YaStr);
    OutTextXY( 5, GSPY( 0.13), ' Analytical:  y(0)='+ YaStr+ 'm');
    XCf:= sqrt( 0.5/ CB( t));
    Str( XCf: 8: 2, XcStr);
    OutTextXY( 5, GSPY( 0.18), 'Point of CFl: x='+ XcStr+ 'm');
    NCf:= round( XCf/ dx);
    Str( Ncf* dx: 8: 2, XcStr);
    OutTextXY( 5, GSPY( 0.22), 'Nearest GrPo: x='+ XcStr+ 'm');
    Str( ynn[ NCf]: 8: 2, YnStr);
    OutTextXY( 5, GSPY( 0.26), ' Numerical:   y='+ YnStr+ 'm');
    Str( yan[ NCf]: 8: 2, YaStr);
    OutTextXY( 5, GSPY( 0.30), ' Analytical:  y='+ YaStr+ 'm');
  END;

PROCEDURE  Time_display( t : REAL);
  VAR TStr : string[ 7];
  BEGIN
    Str( t: 7: 4, TStr);
    OutTextXY( 2, GSPY( 0.05), 't='+ TStr+ 'yr');
  END;


PROCEDURE  Graph_window( purpose : OName;
                         message : OName);
  VAR ch : CHAR;
  BEGIN
    IF (purpose= 'anadat') OR (purpose= 'numdat')
       OR (purpose= 'picdat') THEN
    BEGIN
      SetViewPort( GSPX( 0.60), GSPY( 0.15), GSPX( 0.98), GSPY( 0.46),
                   ClipOn);
      ClearViewPort;
      SetViewPort( 0, 0, XsMax, YsMax, ClipOn);
      Rectangle( GSPX( 0.60), GSPY( 0.15), GSPX( 0.98), GSPY( 0.46));
      SetViewPort( GSPX( 0.60), GSPY( 0.15), GSPX( 0.98), GSPY( 0.46),
                   ClipOn);
      IF purpose= 'anadat' THEN ana_display ELSE Num_display;
    END;
    IF purpose = 'continue' THEN
    BEGIN
      SetViewPort( 10, GSPY( 0.28), GSPX( 0.4), GSPY( 0.35), ClipOn);
      ClearViewPort;
      SetViewPort( 0, 0, XsMax, YsMax, ClipOn);
      Rectangle( 10, GSPY( 0.28), GSPX( 0.4), GSPY( 0.35));
```

```
      SetViewPort( 10, GSPY( 0.28), GSPX( 0.4), GSPY( 0.35), ClipOn);
      OutTextXY( 5, GSPY( 0.05), 'Press [Return] to continue...');
      ch:= Readkey;
    END;
    IF purpose = 'normal' THEN
      SetViewPort( 0, 0, XsMax, YsMax, ClipOn);
    IF purpose = 'pbname' THEN
    BEGIN
      SetViewPort( GSPX( 0.28), GSPY( 0.15), GSPX( 0.55), GSPY( 0.23),
                   Clipon);
      ClearViewPort;
      SetViewPort( 0, 0, XsMax, YsMax, ClipOn);
      Rectangle( GSPX( 0.28), GSPY( 0.15), GSPX( 0.55), GSPY( 0.23));
      SetViewPort( GSPX( 0.28), GSPY( 0.15), GSPX( 0.55), GSPY( 0.23),
                   Clipon);
      OutTextXY( 5, GSPY( 0.05), 'Problem: '+ PbName);
    END;
    IF purpose = 'time' THEN
    BEGIN
      SetViewPort( 10, GSPY( 0.15), GSPX( 0.25), GSPY( 0.23), ClipOn);
      ClearViewPort;
      SetViewPort( 10, GSPY( 0.15), GSPX( 0.25), GSPY( 0.23), ClipOn);
      IF message = '' THEN  Time_display( t)
      ELSE OutTextXY( 5, GSPY( 0.05), message);
    END;
    IF purpose = 'working' THEN
    BEGIN
      SetViewPort( 10, GSPY( 0.28), GSPX( 0.4), GSPY( 0.35), ClipOn);
      ClearViewPort;
      SetViewPort( 10, GSPY( 0.28), GSPX( 0.4), GSPY( 0.35), ClipOn);
      OutTextXY( 2, GSPY( 0.05), message);
    END;
  END;

PROCEDURE  Clear_window( purpose : OName);
  BEGIN
    IF purpose = 'continue' THEN
      SetViewPort( 10, GSPY( 0.28), GSPX( 0.4), GSPY( 0.35), ClipOn);
    IF purpose = 'normal' THEN
      SetViewPort( 0, 0, XsMax, YsMax, ClipOn);
    IF purpose = 'numdat' THEN
      SetViewPort( GSPX( 0.60), GSPY( 0.15), GSPX( 0.98), GSPY( 0.46),
                   ClipOn);
    IF purpose = 'time' THEN
      SetViewPort( 10, GSPY( 0.15), GSPX( 0.25), GSPY( 0.23), ClipOn);
    IF purpose = 'working' THEN
      SetViewPort( 10, GSPY( 0.28), GSPX( 0.4), GSPY( 0.35), ClipOn);
    ClearViewPort;
  END;

PROCEDURE  Initiate_graphics;
  VAR HiMode : INTEGER;
      LoMode : INTEGER;
      ChSize : Word;
  BEGIN
    InitGraph( GraphDr, GraphMo, '');
```

```
    GetModeRange( GraphDr, LoMode, HiMode);
    SetGraphMode( HiMode);
    Xsmax:= GetMaxX;
    Ysmax:= GetMaxY;
    ChSize:= 1;
    SetTextJustify( LeftText, BottomText);
    SetTextStyle( DefaultFont, HorizDir, ChSize);
    SetLineStyle( SolidLn, 0, NormWidth);
  END;

PROCEDURE  Draw_Boundary( option: OName);
  BEGIN
    SetLineStyle( Solidln, 0, ThickWidth);
    Rectangle( 0,  0, Xsmax, GSPY( 0.12));
    Rectangle( 0, GSPY( 0.12), Xsmax, Ysmax);
    SetLineStyle( SolidLn, 0, NormWidth);
    OutTextXY( 10, GSPY( 0.09), 'GaussLine');
    OutTextXY( GSPX( 0.65), GSPY( 0.09), option);
  END;

PROCEDURE  Draw_coastline( VAR  y: CLine);
  VAR HStr  : string[ 6];
      i     : INTEGER;
      xp    : INTEGER;
      yp    : INTEGER;
  BEGIN
    MoveTo( 0, YZero);
    FOR i:= -npts TO npts DO
    BEGIN
      x:= i* dx;
      xp:=  trunc( x/ XScale)+ XZero;
      yp:= -trunc( y[ i]/ YScale)+ YZero;
      LineTo( xp, yp);
    END;
    IF NOT flag THEN
    BEGIN
      xp:= GSPX( 0.48);
      yp:=-GSPY( 0.00)- trunc( y[ 0]/ YScale) + YZero;
      Str( y[ 0]: 4: 0, HStr);
      OutTextXY( xp, yp, HStr+ 'm');
      HStr:= '0';
      OutTextXY( GSPX( 0.5), GSPY( 0.95), HStr+ 'm');
      Str( -npts* dx: 6: 0, HStr);
      OutTextXY( 2, GSPY( 0.95), HStr+ 'm');
      xp:= GSPX( 0.90);
      yp:= GSPY( 0.95);
      Str( npts* dx: 6: 0, HStr);
      OutTextXY( xp, yp, HStr+ 'm');
    END;
  END;

PROCEDURE  Del_coastline( VAR  y: cline);
  VAR DrawColor : word;
  BEGIN
    DrawColor:= GetColor;
    SetColor( GetBkColor);
```

```
    Draw_coastline( y);
    SetColor( DrawColor);
  END;

{--- GRFPROC.LIB ------------------------------------------- END ----}
```

```
{--- I-OPROC.LIB ---------------------------------------- BEGIN ----}
{ This library contains some tools for I/O to screen and disk.      }

{--- I-OPROC.LIB ------------------------------------- CONTENTS ----}
{ FUNC  Exists           :  Check problem-name                      }
{ PROC  Name_list        :  Display existing problem-names          }
{ PROC  Skip_char        :  Read from fixed position                }
{ PROC  Ask_data         :  Read data from keyboard                 }
{ PROC  Read_data        :  Read data from file on disk             }
{ PROC  Config_defaults  :  Get configuration of PC                 }
{ PROC  Get_analyt       :  Obtain input data for option Analyt     }
{ PROC  Get_compare      :  Obtain input data for option Compare    }
{ PROC  Get_picture      :  Obtain input data for option Picture    }
{ PROC  Get_data         :  Obtain data for a programme option      }
{ PROC  Get_text         :  Obtain text for option Explain          }
{ PROC  Put_data         :  Write results to disk                   }

{--- I-OPROC.LIB ---------------------------------------------------}

VAR  NwPbm : BOOLEAN;
     WkFil : Text;

FUNCTION  Exists( PbName : FName) : BOOLEAN;
  BEGIN
    assign( WkFil, DD+ PbName+ '.GSL');
    {$I-} Reset( WkFil); {$I+}
    IF IOResult= 0 THEN
    BEGIN
      Close( WkFil);
      Exists:= TRUE;
    END
    ELSE Exists:= FALSE;
  END;

PROCEDURE  Name_list;
  VAR  dot   : INTEGER;
       EPbNm : SearchRec;
       Nme   : string[ 9];
  BEGIN
    Blank( 4); writeln( 'Existing problem-names are:');
    FindFirst( DD+ '*.GSL', Archive, EPbNm);
    WHILE DosError= 0 DO
    BEGIN
      dot:= Pos( '.', EPbNm.Name);
      Nme:= Copy( EPbNm.Name, 1, dot-1);
      Blank( 8); writeln( Nme);
      FindNext( EPbNm);
    END;
  END;

PROCEDURE  Skip_char( n : INTEGER);
  VAR c : CHAR;
      i : INTEGER;
  BEGIN
    FOR i:=1 TO n DO read( WkFil, c);
  END;
```

```
PROCEDURE  Ask_data( option : OName);
  VAR hCA : REAL;
      hCB : REAL;
  BEGIN
    Blank( 2);
    writeln( '--- Please supply the following input data ---');
    Blank( 2); write( 'Top of the fill     [m]   >>> '); read( Tf);
    Blank( 2); write( 'Width of the fill   [m]   >>> '); read( Wf);
    Blank( 2); write( 'Volume of the fill  [m3]  >>> '); read( Vf);
    writeln;
    Blank( 2); write( 'Coastal constant  [m3/yr/rad]  >>> '); read( s);
    IF option= 'Compare' THEN
    BEGIN
      Blank( 2);
      write( 'Angle of wave incidence [deg]  >>> '); read(phi0);
    END;
    writeln;
    Blank( 2); write( 'Computation period  [yr]  >>> '); read( te);
    hCB:= 1/( 2* Wf* Wf); hCA:= Tf;
    Ti:= Vf/( 4* hCA* s* sqrt( Pi* hCB)); h:= 4* hCB* s* Ti;
    phi0:= Pi* phi0/ 180;
  END;

PROCEDURE  Read_data( PbName : FName);
  VAR dumx  : REAL;
      i     : INTEGER;
  BEGIN
    Assign( WkFil, DD+ PbName+ '.GSL');
    SkLn( 2); Blank( 4);
    writeln( 'Reading Data from File: ', DD+ PbName+ '.GSL');
    Blank( 4); writeln( 'Please wait.');
    Reset( WkFil);
    FOR i:= 1 TO 5 DO readln( WkFil);
    Skip_char( 49); readln( WkFil, t0);
    Skip_char( 49); readln( WkFil, Vf);
    Skip_char( 49); readln( WkFil,  h);
    Skip_char( 49); readln( WkFil,  s);
    Skip_char( 49); readln( WkFil, Ti);
    Skip_char( 49); readln( WkFil, phi0);
    FOR i:=1 TO 3 DO readln( WkFil);
    Skip_char( 49); readln( WkFil, dt);
    Skip_char( 49); readln( WkFil, dx);
    Skip_char( 49); readln( WkFil, npts);
    FOR i:=1 TO 5 DO readln( WkFil);
    FOR i:= -npts TO npts DO
    BEGIN
      readln( WkFil, dumx, yao[ i], yno[ i]);
    END;
    Close( WkFil);
    FOR i:= npts+ 1 TO nmax DO
    BEGIN
      yao[ i]:= 0; yno[ i]:= 0;
      yao[-i]:= 0; yno[-i]:= 0;
    END;
  END;
```

```
PROCEDURE  Config_Defaults;
  VAR  drive   : String[ 25];
       CfgFil  : Text;
       i       : INTEGER;
       psn     : INTEGER;
       screen  : String[ 25];
  BEGIN
    assign( CfgFil, 'GSLCFG.DEF');
    Reset( CfgFil);
    FOR i:= 1 TO 12 DO readln( CfgFil);
    readln( CfgFil, screen);
    readln( CfgFil, drive);
    close( CfgFil);
    IF Pos( 'ATT400', screen)<> 0 THEN
    BEGIN
      GraphDr:= ATT400; GraphMo:= ATT400Hi
    END
    ELSE  GraphDr:= Detect;
    psn:= Pos( ':', drive);
    IF psn= 0 THEN  DD:='' ELSE  DD:=Copy( drive, psn-1, psn);
 END;

PROCEDURE  Get_analyt;
  VAR ch : CHAR;
      OK : BOOLEAN;
  BEGIN
    OK:= FALSE;
    REPEAT
      ClrScr;
      SkLn( 2), Top_title( 'Analyt'); SkLn( 2);
      Ask_data( 'Analyt');
      SkLn( 2), Blank( 2);
      write( '---- All data correct "[Y/N] : '); ch:=readkey;
      writeln( ch);
      IF ch IN [ 'Y', 'y'] THEN OK:= TRUE;
    UNTIL OK;
  END;

PROCEDURE  Get_compare;
  VAR  ch   : CHAR;
       i    : INTEGER;
       OK   : BOOLEAN;
       tn   : REAL;
  BEGIN
    ClrScr;
    SkLn( 2); Top_title( 'Problem-name'); SkLn( 2);
    Blank( 4);
    writeln( 'To restart a previous problem you will have to provide');
    Blank( 4);
    writeln( 'an already existing problem-name.');
    Blank( 4);
    writeln( 'Or you can define a new problem by providing a new');
    Blank( 4);
    writeln( '(not existing) problem name.');
    SkLn( 2); Blank( 4);
    write( 'To restart a previous problem type: '); invers( ' E ');
```

```
SkLn( 2); Blank( 4);
write( 'Or to define a new problem type:    '); invers(' N ');
SkLn( 2); Blank( 4);
write( 'Please specify your choice   >>> ');
REPEAT ch:=readkey UNTIL ch IN [ 'E', 'e', 'N', 'n']; writeln( ch);
IF ch IN  [ 'N', 'n'] THEN
REPEAT
  NwPbm:= TRUE; OK:= FALSE;
  SkLn( 2);
  Blank( 4); writeln( '---- New Problem ----');
  Blank( 4); write( 'Please supply Problem-name: '); readln( PbName);
  IF Exists( PbName) THEN
  BEGIN
    SkLn( 1); Blank( 4); write( '---- ');
    invers( ' This problem-name already exists !!! ');
    SkLn( 1);
    Name_list;
  END;
  IF PbName= '' THEN
  BEGIN
    SkLn( 1); Blank( 4); write( '---- ');
    invers( ' Please supply a problem-name !!! ');
  END;
  IF NOT Exists( PbName) AND  (PbName <> '') THEN
  BEGIN
    ClrScr; SkLn(2);
    Blank( 2); writeln( 'New Problem-name: ', PbName);
    SkLn( 1);
    t0:=0;
    REPEAT
      Ask_data( 'Compare');
      SkLn( 2); Blank( 2);
      write( '---- All data correct  [Y/N] : ');
      REPEAT ch:=readkey UNTIL ch IN [ 'Y', 'y', 'N', 'n'];
      writeln (ch);
      IF ch IN [ 'Y', 'y'] THEN  OK:= TRUE;
    UNTIL OK;
  END;
UNTIL OK
ELSE
REPEAT
  NwPbm:= FALSE; OK:= FALSE; SkLn( 2);
  Blank( 4); writeln( '---- Existing Problem ----');
  Blank( 4); write( 'Please supply Problem-name: '); readln( PbName);
  IF NOT Exists( PbName) THEN
  BEGIN
    SkLn( 1);
    Blank( 4); write( '---- ');
    invers( ' This problem-name does not exist !!! '); SkLn( 1);
    Name_list;
  END
  ELSE
  BEGIN
    Read_data( PbName);
    SkLn( 2); Blank( 4);
    writeln( 'Now for this new comparison you have to provide');
```

```
        Blank( 4);
        writeln( 'a new (not existing) problem-name'); SkLn( 2);
        REPEAT
          Blank( 4); writeln( '---- New Problem ----');
          Blank( 4); write( 'Please supply Problem-name: '); readln( PbNa
          IF Exists( PbName) THEN
          BEGIN
            SkLn( 1); Blank( 4);
            write( '---- ');
            invers( ' This problem-name already exists !!! ');
            SkLn( 1);
            Name_list;
          END;
        UNTIL NOT Exists( PbName);
        writeln; Blank( 4);
        writeln( 'And also a new'); Blank( 4);
        write( 'computation period  [yr]  >>> '); read( tn);
        te:= t0+ tn;
        OK:= TRUE;
      END;
    UNTIL OK;
    ClrScr; SkLn( 2); Top_title( 'Problem-type'); SkLn( 2);
    Blank( 4);
    writeln( 'You can compare the analytical solution to the numerical');
    Blank( 4);
    writeln( 'solution in two different ways.');
    Blank( 4);
    writeln( 'You can use a LINEAR relation between the angle of wave');
    Blank( 4);
    writeln( 'incidence and the longshore transport capacity.');
    Blank( 4);
    writeln( 'You can also use a SINUSOIDAL relationship.');
    SkLn( 2); Blank( 4);
    write( 'For the linear relation type:     '); invers( ' L ');
    SkLn( 2); Blank( 4);
    write( 'For the sinusoidal relation type: '); invers( ' S ');
    SkLn( 2); Blank( 4);
    write( 'Please specify your choice   >>> ');
    REPEAT ch:=readkey UNTIL ch IN [ 'L', 'l', 'S', 's']; writeln( ch);
    IF ch IN  [ 'L', 'l'] THEN lin:=TRUE ELSE lin:= FALSE;
  END;

PROCEDURE  Get_picture;
  VAR OK : BOOLEAN;
  BEGIN
    ClrScr;
    SkLn( 2); Top_title( 'Problem-name'); SkLn( 2);
    Blank( 4);
    writeln( 'To get a picture of the results of a problem computed');
    Blank( 4);
    writeln( 'earlier you will have to provide an already existing');
    Blank( 4);
    writeln( 'problem-name.');
    Blank( 4); SkLn( 2);
    OK:= FALSE;
    REPEAT
```

```
      Blank( 4); writeln( '---- Existing Problem ----');
      Blank( 4); write( 'Please supply Problem-name: '); readln( PbName);
      IF NOT Exists( PbName) THEN
      BEGIN
        SkLn( 1);
        Blank( 4); write( '---- ');
        invers( ' This problem-name does not exist !!! '); SkLn( 1);
        Name_list;
      END
      ELSE
      BEGIN
        Read_data( PbName);
        OK:= TRUE;
      END;
    UNTIL OK;
  END;

PROCEDURE  Get_data( option : OName);
  BEGIN
    IF option= 'Analyt'  THEN Get_analyt;
    IF option= 'Compare' THEN Get_compare;
    IF option= 'Picture' THEN Get_picture;
  END;

PROCEDURE  Get_text( option : OName);
  VAR ch     : CHAR;
      ExpFil : Text;
      i      : INTEGER;
      npag   : INTEGER;
      NxtPg  : BOOLEAN;
      TxtLn  : String[ 70];
  BEGIN
    Assign( ExpFil, 'GSLEXP.TXT');
    Reset( ExpFil);
    FOR i:=1 TO 5 DO readln( ExpFil);
    readln( ExpFil, npag); readln( ExpFil);
    FOR i:=1 TO npag DO
    BEGIN
      ClrScr;
      Explain_screen( i);
      NxtPg:= FALSE;
      REPEAT
        readln( ExpFil, TxtLn);
        IF TxtLn[ 1]= '#' THEN NxtPg:= TRUE
        ELSE
        BEGIN
          Blank( 2); writeln( TxtLn);
        END;
      UNTIL NxtPg;
      GotoXY( 40, 24); write( 'Press [Return] to continue... ');
      ch:=readkey;
    END;
    Close( ExpFil);
  END;
```

```
PROCEDURE  Put_data( PbName : FName);
  VAR i       : INTEGER;
      OutFil : Text;
  BEGIN
    Assign( OutFil, DD+ PbName+ '.GSL');
    Rewrite( OutFil);
    Writeln( OutFil,
     '--- GaussLine ----------------------------------------------------'
    Writeln( OutFil,
     '                ****  Problem: ', PbName, ' ****');
    Writeln( OutFil); Writeln( OutFil);
    Writeln( OutFil,
     '--- GaussLine ----------------------------------- Problem Data ----'
    Writeln( OutFil,
     'Period after placement of the fill  [yr]       : ',  t: 12: 2);
    Writeln( OutFil,
     'Volume of the fill                  [m3]       : ', Vf: 10: 0);
    Writeln( OutFil,
     'Profile height                      [m]        : ',  h: 12: 2);
    Writeln( OutFil,
     'Coastal constant                    [m3/yr/rad]: ',  s: 12: 2);
    Writeln( OutFil,
     'Fictive initial period              [yr]       : ', Ti: 12: 2);
    Writeln( OutFil,
     'Angle of wave incidence             [rad]      : ', phi0: 14: 4);
    Writeln( OutFil); Writeln( OutFil);
    Writeln( OutFil,
     '--- GaussLine ---------------- Data for Numerical Computation ----'
    Writeln( OutFil,
     'Time step                           [yr]       : ', dt: 14: 4);
    Writeln( OutFil,
     'Grid distance                       [m]        : ', dx: 12: 2);
    Writeln( OutFil,
     'Number of points                    [-]        : ', npts: 10);
    Writeln( OutFil); Writeln( OutFil);
    Writeln( OutFil,
     '--- GaussLine ---------------------- Results from Computation ----'
    Writeln( OutFil,
     ' x-coordinate                     shoreline position             '
    Writeln( OutFil,
     '    x [m]           y (analytical) [m]          y (numerical) [m] '
    FOR i:= -npts TO +npts DO
    BEGIN
      x:= i* dx;
      Writeln( OutFil,  ' ': 3,  x: 8: 2,
                        ' ': 13, yan[ i]: 8: 2,
                        ' ': 21, ynn[ i]: 8: 2);
    END;
    Writeln( OutFil,
     '--- GaussLine ----------------------------------- End of data ----'
    Writeln( OutFil);
    Close( OutFil);
  END;

{--- I-OPROC.LIB ------------------------------------------ END ----}
```

```
{--- INIPROC.LIB -------------------------------------- BEGIN ----}
{ This library contains some tools for initiation of the GaussLine  }
{ programme and the GaussLine programme options.                    }

{--- INIPROC.LIB ------------------------------------ CONTENTS ----}
{ PROC  Create_flag            :  Generation flagpage               }
{ PROC  Draw_flag              :  Display flagpage                  }
{ PROC  Ini_anal               :  Initial conditions analyt option  }
{ PROC  Ini_comp               :  Initial conditions compare option }
{ PROC  Ini_pict               :  Initial conditions picture option }
{ PROC  Initiate_calculation   :  Initiation of a programme option  }

{--- INIPROC.LIB -------------------------------------------------}

PROCEDURE  Create_flag;
  VAR  FlgFil : TEXT;
       i      : INTEGER;
       j      : INTEGER;
       tim    : ARRAY[ 1..4] OF REAL;
       yfl    : CLine;
  BEGIN
    npts:= 400;
    dx:= 10;
    Vf:= 1e6;
    h  := 25;
    s  := 5e5;
    Ti:= 0.3;
    tim[ 1]:= 0; tim[ 2]:= 0.1; tim[ 3]:= 0.5; tim[ 4]:= 2;
    assign( FlgFIl, 'GSLFLG.DAT');
    Rewrite( FlgFil);
    FOR i:=1 TO 4 DO
    BEGIN
      Anly( yfl, tim[ i], npts);
      FOR j:= -npts TO npts DO
      BEGIN
        IF (j MOD 10)= 0 THEN  writeln( FlgFil);
        write( FlgFil, '  ', yfl[ j]: 4: 0);
      END;
    END;
    Close( FlgFil);
  END;

PROCEDURE  Draw_flag;
  VAR FlgFil : Text;
      i      : INTEGER;
      j      : INTEGER;
      yfl    : CLine;
  BEGIN
    flag:= TRUE;
    npts:= 400;
    dx:=10;
    XScale:= 2000/ XsMax;
    YScale:= 300/ YsMax;
    XZero:= GSPX( 0.5);
    YZero:= GSPY( 0.9);
    Draw_boundary( 'Release of August 15, 1988');
```

```
    Graph_window( 'time', 'Version 1.0');
    Graph_window( 'normal', '');
    OutTextXY( GSPX( 0.65), GSPY( 0.30), 'Deformation of');
    OutTextXY( GSPX( 0.65), GSPY( 0.35), 'a nourishment ');
    OutTextXY( GSPX( 0.65), GSPY( 0.40), 'with the shape of');
    OutTextXY( GSPX( 0.65), GSPY( 0.45), 'a Gaussian curve');
    assign( FlgFIl, 'GSLFLG.DAT');
    Reset( FlgFil);
    FOR i:=1 TO 4 DO
    BEGIN
      FOR j:= -npts TO npts DO
      BEGIN
        read( FlgFil, yfl[ j]);
      END;
      Draw_coastline( yfl);
    END;
    Close( FlgFil);
    Flag:= FALSE;
    Graph_window( 'continue', '');
  END;

PROCEDURE  Ini_anal;
  VAR DefFil : TEXT;
      i      : INTEGER;
  BEGIN
    Assign( DefFil, 'GSLNUM.DEF');
    Reset( DefFil);
    FOR i:=1 TO 12 DO readln( DefFil);
    readln( DefFil, dx);
    close( DefFil);
    eWf:= 4* sqrt( 0.5/ CB( te));
    npts:= trunc( eWf/ dx)+ 1;
    Xscale:= eWf/( 0.5* Xsmax);
    Yscale:= CA( 0)/( 0.5* Ysmax);
    XZero := GSPX( 0.5);
    YZero := GSPY( 0.9);
  END;

PROCEDURE  Ini_comp;
  CONST eps     = 1e-6;
  VAR   ch      : CHAR;
        DefFil  : Text;
        i       : INTEGER;
  BEGIN
    Assign( DefFil, 'GSLNUM.DEF');
    Reset( DefFil);
    FOR i:=1 TO 12 DO readln( DefFil);
    IF NwPbm THEN
    BEGIN
      readln( DefFil, dx);
      readln( DefFil, dt);
    END
    ELSE
    BEGIN
      readln( DefFil); readln( DefFil);
    END;
```

```
    readln( DefFil, iview);
    close( DefFil);
    Stability_check( dt, 'FTCS');
    nstep:= trunc( (te- t0)/ dt -eps)+ 1;
    IF NOT NwPbm THEN
    BEGIN
      IF yno[ 0]> yao[ 0] THEN
        Tf:= yno[ 0] ELSE Tf:= yao[ 0];
    END;
    eWf:= 4* sqrt( 0.5/ CB( te));
    npts:= trunc( eWf/ dx)+ 1;
    IF npts > nmax-2  THEN
    BEGIN
      SkLn( 2);
      Top_Title( 'init_comp');
      Blank( 2); invers( ' warning:'); writeln( ' Array too small.');
      npts:= nmax- 2;
      Blank( 2); writeln( 'Standard corrective action has been taken.');
      Blank( 2); write(' Press [return] to continue... '); ch:=readkey;
    END;
    IF NwPbm THEN
    BEGIN
      Anly( yao, t0, npts);
      yno:= yao;
    END;
    FOR i.=1 TO 2 DO
    BEGIN
      yno[ npts+ i]:=0;
      yno[-npts- i]:=0;
      ynn[ npts+ i]:=0;
      ynn[-npts- i]:=0;
    END;
    ynd:= yno;
    ynn:= yno;
    Xscale:= eWf/( 0.5* Xsmax);
    Yscale:= Tf/( 0.5* Ysmax);
    XZero := GSPX( 0.5);
    YZero := GSPY( 0.9);
  END;

PROCEDURE  Ini_pict;
  VAR YMax : REAL;
  BEGIN
    eWf:= 4* sqrt( 0.5/ CB( t0));
    IF yno[ 0]> yao[ 0] THEN YMax:= yno[ 0]
      ELSE YMax:= yao[ 0];
    Xscale:= eWf/( 0.5* Xsmax);
    Yscale:= YMax/( 0.5* Ysmax);
    XZero := GSPX( 0.5);
    YZero := GSPY( 0.9);
    ynn:= yno;
    yan:= yao;
  END;

PROCEDURE  Initiate_calculation( option : OName);
  BEGIN
```

```
    SkLn( 2);
    Top_Title( 'initiating');
    IF option= 'Analyt'  THEN Ini_anal;
    IF option= 'Compare' THEN Ini_comp;
    IF option= 'Picture' THEN Ini_pict;
  END;

{--- INIPROC.LIB ------------------------------------------ END ----}
```

```
{--- OPTPROC.LIB ---------------------------------------- BEGIN -----}
{ This library contains the GaussLine programme options.             }

{--- OPTPROC.LIB -------------------------------------- CONTENTS ----}
{ PROC  Explain   :  Help Screens                                     }
{ PROC  Analyt    :  Analytical solution                              }
{ PROC  Compare   :  Comparison of analytical and numerical solution }
{ PROC  Picture   :  Show result of specific comparison               }

{--- OPTPROC.LIB ----------------------------------------------------}

PROCEDURE  Explain;
  BEGIN
    Get_text( 'Explain');
  END;

PROCEDURE  Analyt;
  VAR ch     : CHAR;
      ready : BOOLEAN;
  BEGIN
    ClrScr;
    Get_data( 'Analyt');
    ready:= FALSE;
    REPEAT
      Initiate_graphics;
      Initiate_calculation( 'Analyt');
      ClearViewPort;
      Draw_boundary( 'Analyt');
      Anly( yao, 0, npts);
      Draw_coastline( yao);
      Anly( yan, te, npts);
      Draw_coastline( yan);
      t:= te; Graph_window( 'time', '');
      Graph_window( 'anadat', '');
      Graph_window( 'continue', '');
      RestoreCrtMode;
      SkLn( 2);
      Top_title( 'Analyt');
      SkLn( 1);
      Blank( 4); write( 'Another value for "t" [Y/N] : ');
      REPEAT ch:=readkey UNTIL ch IN [ 'Y', 'y', 'N', 'n']; write( ch);
      IF ch IN [ 'Y', 'y'] THEN
      BEGIN
        SkLn( 1); Blank( 4);
        write( 'Please supply computation period  [yr]  >>> ');
        read( te);
      END
      ELSE ready:= TRUE;
    UNTIL ready;
  END;

PROCEDURE  Compare;
  VAR i     : INTEGER;
  BEGIN
    ClrScr;
    Get_data( 'Compare');
```

```
    Initiate_calculation( 'Compare');
    Initiate_graphics;
    Draw_boundary( 'Compare');
    Draw_coastline( yno);
    Draw_coastline( yao);
    Graph_window( 'pbname', '');
    FOR i:=1 TO nstep DO
    BEGIN
      t:= t0+ i* dt;
      Graph_window( 'time', '');
      Graph_window( 'working','computing...');
      IF lin THEN FTCS_lin( ynn, yno, npts)
        ELSE FTCS_sin( ynn, yno, npts);
      IF ( i MOD iview)= 0 THEN
      BEGIN
        Anly( yan, t, npts);
        Clear_window( 'working');
        Clear_window( 'numdat');
        Graph_window( 'normal', '');
        Del_coastline( yao);
        Del_coastline( ynd);
        Draw_coastline( yan);
        Draw_coastline( ynn);
        Graph_window( 'pbname', '');
        Graph_window( 'numdat', '');
        Graph_window( 'continue', '');
        yao:= yan;
        ynd:= ynn;
      END;
      yno:= ynn;
    END;
    Graph_window( 'time', '');
    Graph_window( 'normal', '');
    Anly( yan, t, npts);
    Del_coastline( yao);
    Del_coastline( ynd);
    Draw_coastline( yan);
    Draw_coastline( ynn);
    Graph_window( 'pbname', '');
    Graph_window( 'numdat', '');
    Graph_window( 'working', 'Writing data to disk '+ DD);
    Put_data( PbName);
    Graph_window( 'continue', '');
    RestoreCrtMode;
  END;

PROCEDURE  Picture;
  BEGIN;
    Get_data( 'Picture');
    Initiate_calculation( 'Picture');
    Initiate_graphics;
    Draw_boundary( 'Picture');
    Draw_coastline( yno);
    Draw_coastline( yao);
    t:= t0; Graph_window( 'time', '');
    Graph_window( 'pbname', '');
```

```
    Graph_window( 'picdat', '');
    Graph_window( 'continue', '');
    RestoreCrtMode;
  END;

{--- OPTPROC.LIB ------------------------------------------- END ----}
```

```
--- GaussLine ----------------------------------- Default Values -----1
-                                                                     2
-  This file contains default values for the numerical estimation     3
-  of the shoreline ( dx, dt)                                         4
-  and a default value for the output to the screen ( iview).         5
-                                                                     6
-  ** When desired YOU MAY MODIFY these defaults YOURSELF **          7
-                                                                     8
-  Therefore you can overwrite the values at the left side of the     9
-  text below (please notice the plus (+) sign).                     10
-  However make sure NOT to alter these first TWELVE (12) lines.     11
----------------------------+----------------------------------------12
    10                      + (dx)     x-grid distance     [m]
     0.001                  + (dt)     time step           [yr]
   200                      + (iview)  interval graphical output
----------------------------+-----------------------------------------
  The last interval denotes the NUMBER OF TIME STEPS between two
  successive sets of graphical output to the screen.
======================================================================
```

```
--- GaussLine ---------------------------------- Default Values -----1
-                                                                    2
-  This file contains default values for the configuration of the    3
-  hardware of your personal computer.                               4
-                                                                    5
-  ** When desired YOU MAY MODIFY these defaults YOURSELF **         6
-      possibilities for Graph-Card :  ATT400  or  AUTODETECT        7
-                    for Data-Drive :  any valid drive specification 8
-  Therefore you can overwrite the values at the left side of the    9
-  text below from the first (!) colum of the line.                 10
-  However make sure NOT to alter these first TWELVE (12) lines.    11
-----------------------------+---------------------------------------12
ATT400                       + Graph-Card   (ATT400 or AUTODETECT)
A:                           + Data-Drive   (any valid drive)
-----------------------------+---------------------------------------
  The ATT400 graphics card can not be autodetected and therefore has
  to be specified. It is used for instance in the Olivetti PC.
  The data drive field may be left empty. In that case the current
  drive will be used.
======================================================================
```

```
--- GaussLine --------------------------------- Explanation file ----
-                                                                   -
- This file contains the text used by the Explain option on the main -
- menu of the GaussLine programme.                                  -
---------------------------------------------------------------------
6
#  1
GAUSSLINE PROGRAMME

This programme has been designed to evaluate the influence of the
SINUSOIDAL relation between LONGSHORE SAND TRANSPORT CAPACITY and
the ANGLE OF WAVE INCIDENCE (relative to the coastline direction).

In ANALYTICAL models a LINEAR relation is assumed. Using a sinus-
oidal relation demands a numerical treatment of the equations.

In order to satisfy above objective the GaussLine programme
enables you to
  *  obtain explanation on the programme options    (this option)
  *  study the analytical solution                     (choose A)
  *  compare the analytical to the numerical solution  (choose C)
  *  picture the results of a previous comparison      (choose P)

The application of the programme is RESTRICTED to the situation of
a coastline with an initial shape of a Gaussian curve.
#  2
ANALYTICAL SOLUTION                                   (Option A)

This option enables you to study the behaviour in time of a beach-
fill with the initial shape of a Gaussian curve.
The analytical solution is used to compute the shoreline position.

First you will have to provide the geometry of the fill and then
you can give several time intervals to study its behaviour.

The programme will ask you for the following data
  *  position of the top of the fill      [m]
  *  width of the fill                    [m]
     (i.e distance from the y-axis to a point of contraflexure)
  *  volume of the fill                   [m3]
  *  coastal constant                     [m3/yr/rad]
  *  (several) time interval(s)           [yr]

The shoreline position for each time interval is shown on the
screen together with some characteristic values.
#  3
COMPARE ANALYTICAL AND NUMERICAL SOLUTION         (Option C)

This option enables you to compare the differences in shoreline
position computed by a numerical method to those computed by the
analytical solution.

In the case of a numerical method either a sinusoidal or a linear
relation between angle of wave incidence and transport of sand
can be used.
As a matter of course using the linear relation has to yield the
```

same results as using the analytical solution.

For comparison you can define a new problem or you can continue an already existing problem.

4

NEW PROBLEM

First you will have to provide a new problem-name.
Then the programme will ask you for

* position of the top of the fill [m]
* width of the fill [m]
  (i.e distance from the y-axis to a point of contraflexure)
* volume of the fill [m3]
* coastal constant [m3/yr/rad]
* angle of wave incidence [deg]
* computation period [yr]

EXISTING PROBLEM

First you will have to provide an existing problem-name.
The results of this comparision will be used as an initial condition for your continued comparison.

The data will be read from the file 'PbName'.GSL.
After reading you will have to supply a new problem-name and a new computation period for this continued comparison.

5

Default values for the time step (dt), the grid distance (dx) and the interval of successive output to the screen (iview) are stored in the file GSLNUM.DEF. You can modify these defaults yourself by editing the file and changing the values. You have to do this before you start the GaussLine programme.

Also the default destination for reading data from and writing data to disk ('PbName'.GSL files) can be changed. Therefore you have to modify the file GSLCFG.DEF before you start the GaussLine programme.

Output of the compare option on your PRINTER can be obtained by pressing the SCRPRT (Screen Print) key on your keyboard during execution of the programme. The actual picture on the screen will be printed. Output files ('PbName.GSL') can be printed using the MS-DOS print command.

6

PICTURE RESULT OF COMPUTATION (Option P)

This option enables you to look at the result of a comparison carried out previously.

You will have to provide an existing problem-name.
The GaussLine programme will open the file 'PbName'.GSL and will show the resulting shoreline positions on screen.

If desired you can use the SCRPRT key on your keyboard to send this picture to your printer.

QUIT (Option Q)

This option quits the GaussLine programme execution and returns you to the MS DOS operating system.
END

GSLFLG.DAT -- Datafile for flagpage.

```
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    1    1    1    2    2    3
  3    4    6    7    9   11   13   16   19   23
 28   32   38   44   50   57   64   72   80   88
 96  104  112  119  125  131  136  140  143  145
146  145  143  140  136  131  125  119  112  104
 96   88   80   72   64   57   50   44   38   32
 28   23   19   16   13   11    9    7    6    4
  3    3    2    2    1    1    1    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0
```

```
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1
1	1	1	2	2	3	3	4	5	6
8	9	11	13	15	18	21	24	28	32
36	41	46	51	57	62	68	74	80	86
92	98	103	108	113	117	120	123	125	126
126	126	125	123	120	117	113	108	103	98
92	86	80	74	68	62	57	51	46	41
36	32	28	24	21	18	15	13	11	9
8	6	5	4	3	3	2	2	1	1
1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0									
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1	1	2
2	2	2	3	3	4	4	5	6	6
7	8	9	11	12	13	15	16	18	20
22	24	26	29	31	34	36	39	42	45
48	51	54	57	60	63	66	69	71	74
76	79	81	83	84	86	87	88	89	89
89	89	89	88	87	86	84	83	81	79
76	74	71	69	66	63	60	57	54	51
48	45	42	39	36	34	31	29	26	24
22	20	18	16	15	13	12	11	9	8
7	6	6	5	4	4	3	3	2	2
2	2	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0									
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1
1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	3	3	3	3
4	4	4	5	5	5	6	6	7	7
7	8	8	9	10	10	11	11	12	13

```
14   14   15   16   17   18   18   19   20   21
22   23   24   25   26   27   28   29   30   31
32   33   34   35   36   37   38   39   40   41
42   43   44   45   46   47   47   48   49   49
50   50   51   51   52   52   52   52   52   53
53   53   52   52   52   52   52   51   51   50
50   49   49   48   47   47   46   45   44   43
42   41   40   39   38   37   36   35   34   33
32   31   30   29   28   27   26   25   24   23
22   21   20   19   18   18   17   16   15   14
14   13   12   11   11   10   10    9    8    8
 7    7    7    6    6    5    5    5    4    4
 4    3    3    3    3    2    2    2    2    2
 2    1    1    1    1    1    1    1    1    1
 1    1    1    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0    0    0    0    0    0    0    0    0    0
 0
```