

Zadanie N3

Celem tego zadania było rozwiązać układ równań postaci $(N+1) \times (N+1)$ dla $N=1000$ i $h=0.01$.

$$\begin{cases} y_0 = 1 \\ (D_2 y)_n + y_n = 0 \\ y_{N-1} - 2y_N + y_0 = 0 \end{cases}$$

Gdzie

$$(D_2 y)_n = \frac{y_{n-1} - 2y_n + y_{n+1}}{h^2}$$

Równanie zapisałam w postaci macierzowej. Macierz podaną w poleceniu można było przekształcić mnożąc przez h^2 od drugiego do przedostatniego wiersza, wówczas macierz wygląda następująco:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1, & -2+h^2, & 1, & 0 & \dots & 0 & 0 & 0 \\ 0 & 1, & -2+h^2, & 1, & \dots & 0 & 0 & 0 \\ 0 & 0 & 1, & -2+h^2, & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2+h^2 & 1 & 0 \\ 0 & 0 & 0 & 0 & \dots & 1 & -2+h^2 & 1 \\ 1 & 0 & 0 & 0 & \dots & 0 & 1 & -2 \end{bmatrix} \cdot \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_{N-2} \\ y_{N-1} \\ y_N \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Podana macierz jest macierzą rzadką - obliczanie wszystkich elementów macierzy byłoby nieoptymalne, gdyż złożoność obliczeniowa wyniosłaby wówczas $O(N^3)$. Ponieważ większość elementów to 0, a prawie wszystkie niezerowe elementy znajdują się w pobliżu diagonali (jak w macierzy wstęgowej) najlepiej byłoby użyć algorytmów stworzonych dla takich właśnie macierzy, których złożoność może być nawet $O(N)$. Ponieważ jednak ostatni element pierwszej kolumny macierzy jest niezerowy, nie można użyć algorytmu dla macierzy trójdzielnej. Można jednak zastosować wzór Shermanna Morrisona, dzięki któremu łatwo można przekształcić tę macierz na trójdzielną i dzięki temu obliczyć równania równie szybko.

Metoda polega na skonstruowaniu nowej macierzy, którą łatwiej nam policzyć, poprzez dodanie do starej innej macierzy która powstaje z iloczynu dwóch wektorów - u i v^T .

$$A_1 = A + uv^T$$

Wzór wygląda wówczas następująco:

$$A_1^{-1} = A^{-1} - \frac{A^{-1}uv^T A^{-1}}{1 + v^T A^{-1}u}$$

Ja przyjąłam za $u = [0, 0, \dots, 1]^T$ oraz za $v = [1, 0, \dots, 0]^T$ tak, że skonstruowałam

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1, & -2 + h^2, & 1, & 0 & \dots & 0 & 0 & 0 \\ 0 & 1, & -2 + h^2, & 1, & \dots & 0 & 0 & 0 \\ 0 & 0 & 1, & -2 + h^2, & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2 + h^2 & 1 & 0 \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 + h^2 & 1 \\ 1 & 0 & 0 & 0 & \dots & 0 & 1 & -2 \end{bmatrix} =$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1, & -2 + h^2, & 1, & 0 & \dots & 0 & 0 & 0 \\ 0 & 1, & -2 + h^2, & 1, & \dots & 0 & 0 & 0 \\ 0 & 0 & 1, & -2 + h^2, & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2 + h^2 & 1 & 0 \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 + h^2 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & -2 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

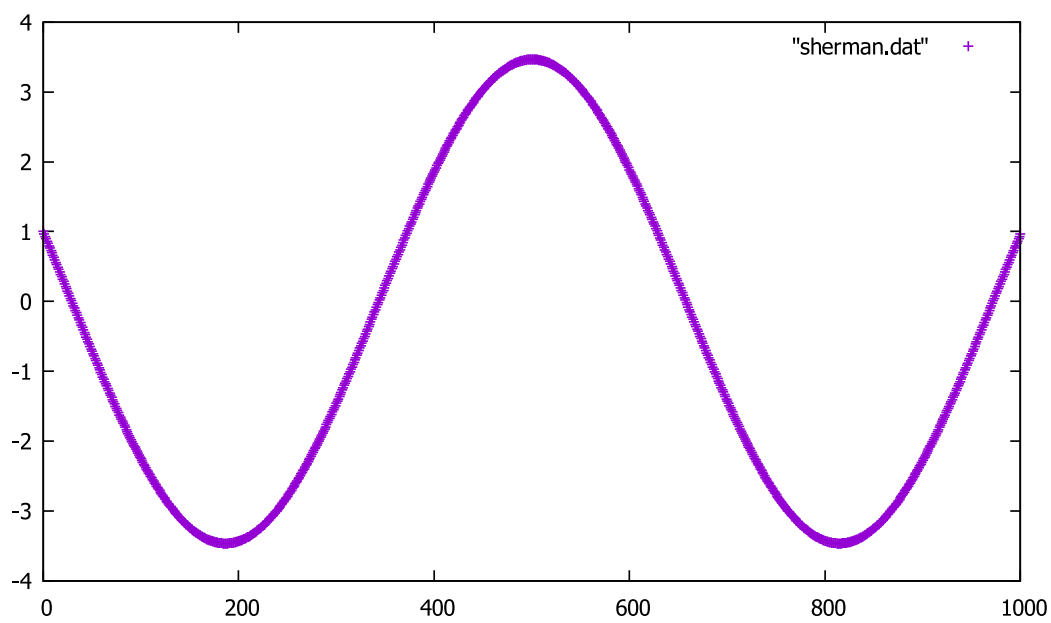
Interesuje nas znalezienie pewnego w $(A_1 w = b \Rightarrow w = A_1^{-1} b)$, które wynosi $(A^{-1} - \frac{A^{-1} u v^T A^{-1}}{1 + v^T A^{-1} u}) b$

Algorytm Shermana-Morrisona prezentuje się następująco:

1. Rozwiązuję równanie $Az = b$ (nową macierz rozwiązuję algorytmem Thomasa, ze starą kolumną wyrazów wolnych)
2. Rozwiązuję równanie $Aq = u$ (tym razem kolumnę wyrazów wolnych zastępuję wektorem u i to również rozwiązuję algorytmem Thomasa)
3. Obliczam $w = z - \frac{v^T z}{1 + v^T q} q$ gdzie w jest moim szukanym wektorem rozwiązań układu, $v^T z$ oraz $v^T q$ są liczbami.

Zadaniem mojego programu jest obliczenie metodą Thomasa wektora z, q oraz obliczenie wektora w według powyższego wzoru. Złożoność tych algorytmów wynosi $O(4N)$ czyli $O(N)$.

Wykres rozwiązań układu równań podanego w poleceniu (nh, y_n) przedstawia się następująco:



Analiza błędów: błędy (oprócz błędów ludzkich) wynikają w tej metodzie głównie z ograniczonych możliwości komputera i liczb zmiennoprzecinkowych. Po podstawieniu do równania $A_1 \cdot x$ otrzymujemy jakieś przybliżenie odpowiedzi b. Różnica pojawia się na 16. miejscu po przecinku więc największy błąd jest rzędu 10^{-16} .

(dodatkowe obliczenia błędu których nie ma w kodzie wyglądały tak)

```
double odp[N+1]; int j=0;
for(int i=0;i<=N;i++){
    odp[i]=down[i]*w[j]+diag[i]*w[j+1]+up[i]*w[j+2];
    printf("%.16f\n",odp[i]);
    j++;
}
```

W zadaniu tym, można było nie tworzyć całych wektorów zmiennych, gdyż większość z nich wynosiła tyle samo, lecz teraz rozwiązanie można rozszerzyć również na inne macierze podobnej postaci, a i tak wciąż pozostaje $O(N)$.

Instrukcje uruchomienia programu:

1. Aby uruchomić program w języku c na linuxie należy przejść do katalogu w którym znajdują się pliki i wpisać w terminalu komendę:
-> make all
2. A następnie
-> make run
3. Aby pozbyć się plików tymczasowych należy użyć komendy:
-> make clean

Poniżej zamieszczony został kod programu oraz wynik jego działań czyli zawartość pliku "sherman.dat"

```

#include <stdlib.h>
#include <stdio.h>

int main(){

FILE *plik;
if((plik=fopen("sherman.dat","w"))==NULL) return -1;
int N = 1000;
double n=1000;
double h=0.0001;
double dzielnik=0;
double dzielnik2=0;
double down[N+1];
down[0]=0; down[N]=1;
double diag[N+1];
diag[0]=1;diag[N]=-2;
double up[N];
up[0]=0;
double b[N+1]; b[0]=1;
b[N]=0;
double z[N+1];
double q[N+1];
double u[N+1]; u[0]=0;
u[N]=1;
double v[N+1]; v[0]=1;
v[N]=0;
for(int i=1;i<=N-1;++i){
    down[i]=1;
    diag[i]=-2+h;
    up[i]=1;
    b[i]=0;
    u[i]=0;
    v[i]=0;
}
double lambda[N],beta[N+1];
double lambda2[N],beta2[N+1];

lambda[0]=up[0]/diag[0];
beta[0]=b[0]/diag[0];

lambda2[0]=up[0]/diag[0];
beta2[0]=u[0]/diag[0];

for(int i=1;i<=N-1;++i){
    dzielnik=diag[i]-(down[i]*lambda[i-1]);
    lambda[i]=(up[i])/dzielnik;

    beta[i]=(b[i]-down[i]*beta[i-1])/dzielnik;

```

```

        dzielnik2=diag[i]-(down[i]*lambda2[i-1]);
        lambda2[i]=(up[i])/dzielnik2;

        beta2[i]=(u[i]-down[i]*beta2[i-1])/dzielnik2;

    }

dzielnik=diag[N]-(down[N]*lambda[N-1]);
beta[N]=(b[N]-down[N]*beta[N-1])/dzielnik;
z[N]=beta[N];
dzielnik2=diag[N]-(down[N]*lambda2[N-1]);
beta2[N]=(u[N]-down[N]*beta2[N-1])/dzielnik2;
q[N]=beta2[N];
// printf("%f\n",beta2[N]);

for(int i=N-1;i>=0;--i){

    z[i]=beta[i]-(lambda[i]*z[i+1]);

    q[i]=beta2[i]-(lambda2[i]*q[i+1]);
    // printf("%f\n",z[i]);
    // printf("%f\n",q[i]);
}
double w[N+1];

double vTz=0;
double vTq=0;

for(int i=0;i<=N;++i){

    vTz+=v[i]*z[i];
    vTq+=v[i]*q[i];

}

for(int i=0;i<=N;i++){

    w[i]=z[i]-(((vTz)/(1+vTq))*q[i]);
    printf("%f\n",w[i]);
    fprintf(plik,"%d %f\n",i,w[i]);
}

fclose(plik);

return 0;
}

```

0 1.000000	51 -0.747514	102 -2.304777	153 -3.275443
1 0.966761	52 -0.781325	103 -2.330552	154 -3.286624
2 0.933425	53 -0.815057	104 -2.356095	155 -3.297477
3 0.899995	54 -0.848707	105 -2.381402	156 -3.308000
4 0.866476	55 -0.882273	106 -2.406471	157 -3.318192
5 0.832870	56 -0.915750	107 -2.431300	158 -3.328052
6 0.799180	57 -0.949136	108 -2.455885	159 -3.337579
7 0.765411	58 -0.982427	109 -2.480225	160 -3.346773
8 0.731565	59 -1.015620	110 -2.504316	161 -3.355632
9 0.697646	60 -1.048711	111 -2.528158	162 -3.364155
10 0.663657	61 -1.081697	112 -2.551746	163 -3.372342
11 0.629602	62 -1.114575	113 -2.575079	164 -3.380192
12 0.595484	63 -1.147342	114 -2.598155	165 -3.387703
13 0.561307	64 -1.179994	115 -2.620971	166 -3.394876
14 0.527073	65 -1.212528	116 -2.643525	167 -3.401710
15 0.492786	66 -1.244941	117 -2.665814	168 -3.408203
16 0.458451	67 -1.277229	118 -2.687837	169 -3.414355
17 0.424069	68 -1.309389	119 -2.709591	170 -3.420166
18 0.389645	69 -1.341419	120 -2.731074	171 -3.425635
19 0.355182	70 -1.373314	121 -2.752284	172 -3.430762
20 0.320684	71 -1.405072	122 -2.773219	173 -3.435545
21 0.286153	72 -1.436690	123 -2.793876	174 -3.439985
22 0.251594	73 -1.468164	124 -2.814255	175 -3.444081
23 0.217010	74 -1.499491	125 -2.834351	176 -3.447832
24 0.182404	75 -1.530668	126 -2.854164	177 -3.451239
25 0.147779	76 -1.561692	127 -2.873692	178 -3.454300
26 0.113140	77 -1.592560	128 -2.892933	179 -3.457016
27 0.078490	78 -1.623268	129 -2.911884	180 -3.459387
28 0.043832	79 -1.653815	130 -2.930544	181 -3.461411
29 0.009169	80 -1.684196	131 -2.948911	182 -3.463090
30 -0.025494	81 -1.714408	132 -2.966983	183 -3.464422
31 -0.060155	82 -1.744449	133 -2.984758	184 -3.465407
32 -0.094810	83 -1.774316	134 -3.002235	185 -3.466046
33 -0.129456	84 -1.804005	135 -3.019411	186 -3.466338
34 -0.164088	85 -1.833514	136 -3.036286	187 -3.466284
35 -0.198704	86 -1.862839	137 -3.052857	188 -3.465883
36 -0.233300	87 -1.891978	138 -3.069123	189 -3.465136
37 -0.267873	88 -1.920928	139 -3.085082	190 -3.464042
38 -0.302419	89 -1.949686	140 -3.100732	191 -3.462601
39 -0.336935	90 -1.978249	141 -3.116072	192 -3.460815
40 -0.371417	91 -2.006614	142 -3.131101	193 -3.458682
41 -0.405862	92 -2.034778	143 -3.145817	194 -3.456204
42 -0.440267	93 -2.062739	144 -3.160218	195 -3.453379
43 -0.474627	94 -2.090494	145 -3.174303	196 -3.450210
44 -0.508940	95 -2.118039	146 -3.188070	197 -3.446695
45 -0.543202	96 -2.145373	147 -3.201519	198 -3.442836
46 -0.577410	97 -2.172493	148 -3.214647	199 -3.438632
47 -0.611560	98 -2.199395	149 -3.227454	200 -3.434085
48 -0.645649	99 -2.226077	150 -3.239939	201 -3.429194
49 -0.679673	100 -2.252536	151 -3.252099	202 -3.423960
50 -0.713629	101 -2.278770	152 -3.263934	203 -3.418384

204 -3.412466	255 -2.680973	306 -1.267136	357 0.469203
205 -3.406207	256 -2.658866	307 -1.234809	358 0.503524
206 -3.399607	257 -2.636494	308 -1.202358	359 0.537794
207 -3.392668	258 -2.613858	309 -1.169787	360 0.572010
208 -3.385389	259 -2.590960	310 -1.137098	361 0.606170
209 -3.377771	260 -2.567803	311 -1.104296	362 0.640268
210 -3.369816	261 -2.544390	312 -1.071384	363 0.674303
211 -3.361523	262 -2.520722	313 -1.038365	364 0.708270
212 -3.352895	263 -2.496802	314 -1.005241	365 0.742167
213 -3.343931	264 -2.472633	315 -0.972017	366 0.775989
214 -3.334633	265 -2.448216	316 -0.938696	367 0.809734
215 -3.325001	266 -2.423555	317 -0.905282	368 0.843397
216 -3.315037	267 -2.398651	318 -0.871776	369 0.876977
217 -3.304741	268 -2.373507	319 -0.838183	370 0.910468
218 -3.294115	269 -2.348125	320 -0.804507	371 0.943869
219 -3.283160	270 -2.322509	321 -0.770750	372 0.977175
220 -3.271876	271 -2.296661	322 -0.736916	373 1.010384
221 -3.260265	272 -2.270583	323 -0.703009	374 1.043491
222 -3.248328	273 -2.244278	324 -0.669031	375 1.076494
223 -3.236066	274 -2.217749	325 -0.634986	376 1.109389
224 -3.223480	275 -2.190998	326 -0.600877	377 1.142174
225 -3.210573	276 -2.164027	327 -0.566709	378 1.174844
226 -3.197344	277 -2.136841	328 -0.532484	379 1.207397
227 -3.183795	278 -2.109440	329 -0.498206	380 1.239829
228 -3.169928	279 -2.081829	330 -0.463877	381 1.272137
229 -3.155744	280 -2.054010	331 -0.429503	382 1.304318
230 -3.141244	281 -2.025985	332 -0.395085	383 1.336368
231 -3.126431	282 -1.997757	333 -0.360628	384 1.368285
232 -3.111304	283 -1.969330	334 -0.326135	385 1.400065
233 -3.095867	284 -1.940706	335 -0.291609	386 1.431705
234 -3.080120	285 -1.911887	336 -0.257055	387 1.463202
235 -3.064065	286 -1.882878	337 -0.222474	388 1.494552
236 -3.047703	287 -1.853680	338 -0.187871	389 1.525753
237 -3.031037	288 -1.824297	339 -0.153250	390 1.556802
238 -3.014068	289 -1.794731	340 -0.118613	391 1.587695
239 -2.996797	290 -1.764986	341 -0.083964	392 1.618428
240 -2.979226	291 -1.735065	342 -0.049307	393 1.649001
241 -2.961358	292 -1.704970	343 -0.014645	394 1.679408
242 -2.943193	293 -1.674704	344 0.020019	395 1.709647
243 -2.924735	294 -1.644271	345 0.054681	396 1.739715
244 -2.905983	295 -1.613674	346 0.089337	397 1.769610
245 -2.886941	296 -1.582915	347 0.123984	398 1.799327
246 -2.867611	297 -1.551998	348 0.158619	399 1.828865
247 -2.847994	298 -1.520926	349 0.193238	400 1.858219
248 -2.828091	299 -1.489701	350 0.227837	401 1.887388
249 -2.807906	300 -1.458328	351 0.262414	402 1.916368
250 -2.787441	301 -1.426809	352 0.296964	403 1.945156
251 -2.766696	302 -1.395147	353 0.331485	404 1.973750
252 -2.745675	303 -1.363346	354 0.365973	405 2.002147
253 -2.724379	304 -1.331408	355 0.400424	406 2.030343
254 -2.702811	305 -1.299337	356 0.434835	407 2.058336

408 2.086124	459 3.172099	510 3.450733	561 2.851111
409 2.113702	460 3.185917	511 3.447273	562 2.831254
410 2.141070	461 3.199416	512 3.443468	563 2.811113
411 2.168223	462 3.212595	513 3.439319	564 2.790692
412 2.195160	463 3.225453	514 3.434826	565 2.769991
413 2.221877	464 3.237988	515 3.429990	566 2.749014
414 2.248371	465 3.250200	516 3.424810	567 2.727761
415 2.274641	466 3.262086	517 3.419288	568 2.706236
416 2.300684	467 3.273647	518 3.413424	569 2.684440
417 2.326496	468 3.284880	519 3.407219	570 2.662376
418 2.352076	469 3.295784	520 3.400673	571 2.640045
419 2.377421	470 3.306359	521 3.393786	572 2.617450
420 2.402528	471 3.316604	522 3.386561	573 2.594594
421 2.427394	472 3.326516	523 3.378997	574 2.571478
422 2.452018	473 3.336096	524 3.371095	575 2.548105
423 2.476397	474 3.345343	525 3.362856	576 2.524478
424 2.500528	475 3.354255	526 3.354280	577 2.500597
425 2.524408	476 3.362831	527 3.345369	578 2.476467
426 2.548037	477 3.371071	528 3.336124	579 2.452089
427 2.571411	478 3.378974	529 3.326545	580 2.427466
428 2.594527	479 3.386539	530 3.316633	581 2.402600
429 2.617384	480 3.393766	531 3.306390	582 2.377494
430 2.639980	481 3.400653	532 3.295816	583 2.352150
431 2.662311	482 3.407200	533 3.284912	584 2.326571
432 2.684376	483 3.413406	534 3.273680	585 2.300759
433 2.706173	484 3.419271	535 3.262121	586 2.274718
434 2.727699	485 3.424794	536 3.250235	587 2.248448
435 2.748952	486 3.429975	537 3.238024	588 2.221954
436 2.769931	487 3.434813	538 3.225490	589 2.195238
437 2.790632	488 3.439307	539 3.212633	590 2.168302
438 2.811054	489 3.443457	540 3.199455	591 2.141149
439 2.831196	490 3.447263	541 3.185956	592 2.113783
440 2.851054	491 3.450724	542 3.172140	593 2.086204
441 2.870627	492 3.453840	543 3.158005	594 2.058418
442 2.889913	493 3.456610	544 3.143556	595 2.030425
443 2.908910	494 3.459035	545 3.128791	596 2.002229
444 2.927616	495 3.461115	546 3.113714	597 1.973833
445 2.946029	496 3.462847	547 3.098326	598 1.945240
446 2.964148	497 3.464234	548 3.082628	599 1.916452
447 2.981970	498 3.465274	549 3.066621	600 1.887473
448 2.999494	499 3.465968	550 3.050308	601 1.858304
449 3.016718	500 3.466315	551 3.033690	602 1.828950
450 3.033641	501 3.466316	552 3.016768	603 1.799414
451 3.050260	502 3.465970	553 2.999545	604 1.769697
452 3.066574	503 3.465277	554 2.982022	605 1.739803
453 3.082581	504 3.464238	555 2.964200	606 1.709735
454 3.098281	505 3.462852	556 2.946082	607 1.679496
455 3.113670	506 3.461120	557 2.927670	608 1.649089
456 3.128748	507 3.459042	558 2.908965	609 1.618518
457 3.143513	508 3.456618	559 2.889968	610 1.587784
458 3.157964	509 3.453848	560 2.870683	611 1.556892

612 1.525844	663 -0.187770	714 -1.853595	765 -3.047655
613 1.494643	664 -0.222373	715 -1.882793	766 -3.064017
614 1.463293	665 -0.256954	716 -1.911803	767 -3.080073
615 1.431797	666 -0.291509	717 -1.940622	768 -3.095821
616 1.400158	667 -0.326035	718 -1.969247	769 -3.111260
617 1.368378	668 -0.360528	719 -1.997675	770 -3.126387
618 1.336462	669 -0.394985	720 -2.025903	771 -3.141202
619 1.304412	670 -0.429403	721 -2.053928	772 -3.155702
620 1.272231	671 -0.463777	722 -2.081748	773 -3.169887
621 1.239923	672 -0.498106	723 -2.109360	774 -3.183755
622 1.207492	673 -0.532384	724 -2.136761	775 -3.197305
623 1.174939	674 -0.566609	725 -2.163949	776 -3.210535
624 1.142269	675 -0.600778	726 -2.190919	777 -3.223443
625 1.109485	676 -0.634887	727 -2.217671	778 -3.236030
626 1.076590	677 -0.668932	728 -2.244201	779 -3.248293
627 1.043587	678 -0.702910	729 -2.270507	780 -3.260231
628 1.010480	679 -0.736818	730 -2.296586	781 -3.271843
629 0.977272	680 -0.770652	731 -2.322435	782 -3.283127
630 0.943966	681 -0.804409	732 -2.348051	783 -3.294084
631 0.910566	682 -0.838086	733 -2.373433	784 -3.304711
632 0.877074	683 -0.871678	734 -2.398578	785 -3.315008
633 0.843495	684 -0.905184	735 -2.423482	786 -3.324973
634 0.809832	685 -0.938599	736 -2.448145	787 -3.334605
635 0.776088	686 -0.971921	737 -2.472562	788 -3.343904
636 0.742266	687 -1.005145	738 -2.496732	789 -3.352869
637 0.708369	688 -1.038268	739 -2.520653	790 -3.361499
638 0.674402	689 -1.071288	740 -2.544322	791 -3.369792
639 0.640368	690 -1.104201	741 -2.567736	792 -3.377748
640 0.606269	691 -1.137003	742 -2.590893	793 -3.385367
641 0.572110	692 -1.169691	743 -2.613791	794 -3.392647
642 0.537894	693 -1.202263	744 -2.636428	795 -3.399588
643 0.503623	694 -1.234714	745 -2.658801	796 -3.406189
644 0.469303	695 -1.267042	746 -2.680909	797 -3.412449
645 0.434935	696 -1.299243	747 -2.702748	798 -3.418368
646 0.400524	697 -1.331315	748 -2.724317	799 -3.423945
647 0.366073	698 -1.363253	749 -2.745613	800 -3.429179
648 0.331586	699 -1.395055	750 -2.766635	801 -3.434071
649 0.297065	700 -1.426717	751 -2.787381	802 -3.438620
650 0.262515	701 -1.458236	752 -2.807847	803 -3.442824
651 0.227938	702 -1.489610	753 -2.828033	804 -3.446684
652 0.193338	703 -1.520835	754 -2.847936	805 -3.450200
653 0.158719	704 -1.551908	755 -2.867554	806 -3.453371
654 0.124085	705 -1.582825	756 -2.886886	807 -3.456196
655 0.089438	706 -1.613585	757 -2.905928	808 -3.458675
656 0.054781	707 -1.644182	758 -2.924680	809 -3.460809
657 0.020120	708 -1.674616	759 -2.943140	810 -3.462597
658 -0.014544	709 -1.704882	760 -2.961306	811 -3.464038
659 -0.049206	710 -1.734977	761 -2.979175	812 -3.465133
660 -0.083863	711 -1.764900	762 -2.996746	813 -3.465882
661 -0.118512	712 -1.794645	763 -3.014018	814 -3.466284
662 -0.153149	713 -1.824211	764 -3.030988	815 -3.466339

816 -3.466047	867 -3.002285	918 -1.774402	969 -0.094911
817 -3.465409	868 -2.984809	919 -1.744536	970 -0.060256
818 -3.464425	869 -2.967035	920 -1.714496	971 -0.025595
819 -3.463094	870 -2.948964	921 -1.684284	972 0.009068
820 -3.461417	871 -2.930598	922 -1.653903	973 0.043731
821 -3.459393	872 -2.911939	923 -1.623358	974 0.078389
822 -3.457024	873 -2.892988	924 -1.592649	975 0.113040
823 -3.454309	874 -2.873749	925 -1.561782	976 0.147679
824 -3.451248	875 -2.854222	926 -1.530758	977 0.182303
825 -3.447843	876 -2.834409	927 -1.499582	978 0.216909
826 -3.444092	877 -2.814314	928 -1.468255	979 0.251493
827 -3.439998	878 -2.793936	929 -1.436782	980 0.286053
828 -3.435559	879 -2.773280	930 -1.405165	981 0.320583
829 -3.430776	880 -2.752346	931 -1.373407	982 0.355082
830 -3.425651	881 -2.731136	932 -1.341512	983 0.389545
831 -3.420183	882 -2.709654	933 -1.309483	984 0.423969
832 -3.414373	883 -2.687901	934 -1.277323	985 0.458351
833 -3.408221	884 -2.665879	935 -1.245035	986 0.492686
834 -3.401729	885 -2.643590	936 -1.212623	987 0.526973
835 -3.394897	886 -2.621037	937 -1.180089	988 0.561207
836 -3.387725	887 -2.598222	938 -1.147437	989 0.595385
837 -3.380214	888 -2.575147	939 -1.114671	990 0.629503
838 -3.372365	889 -2.551814	940 -1.081793	991 0.663558
839 -3.364179	890 -2.528227	941 -1.048807	992 0.697547
840 -3.355657	891 -2.504386	942 -1.015716	993 0.731466
841 -3.346799	892 -2.480295	943 -0.982524	994 0.765312
842 -3.337606	893 -2.455956	944 -0.949233	995 0.799082
843 -3.328080	894 -2.431372	945 -0.915848	996 0.832772
844 -3.318221	895 -2.406544	946 -0.882370	997 0.866378
845 -3.308030	896 -2.381476	947 -0.848805	998 0.899898
846 -3.297508	897 -2.356169	948 -0.815155	999 0.933327
847 -3.286656	898 -2.330627	949 -0.781423	1000 0.966664
848 -3.275476	899 -2.304852	950 -0.747613	
849 -3.263968	900 -2.278846	951 -0.713728	
850 -3.252134	901 -2.252613	952 -0.679772	
851 -3.239975	902 -2.226154	953 -0.645748	
852 -3.227491	903 -2.199473	954 -0.611659	
853 -3.214685	904 -2.172571	955 -0.577510	
854 -3.201557	905 -2.145453	956 -0.543302	
855 -3.188110	906 -2.118119	957 -0.509040	
856 -3.174343	907 -2.090574	958 -0.474727	
857 -3.160259	908 -2.062820	959 -0.440367	
858 -3.145859	909 -2.034860	960 -0.405963	
859 -3.131144	910 -2.006696	961 -0.371518	
860 -3.116117	911 -1.978332	962 -0.337036	
861 -3.100777	912 -1.949769	963 -0.302520	
862 -3.085128	913 -1.921012	964 -0.267974	
863 -3.069170	914 -1.892063	965 -0.233401	
864 -3.052905	915 -1.862924	966 -0.198805	
865 -3.036335	916 -1.833599	967 -0.164189	
866 -3.019461	917 -1.804091	968 -0.129556	