

# **Основы программирования (на языке Си)**

## **Тема 7. Линейные программы “разбор полетов”**

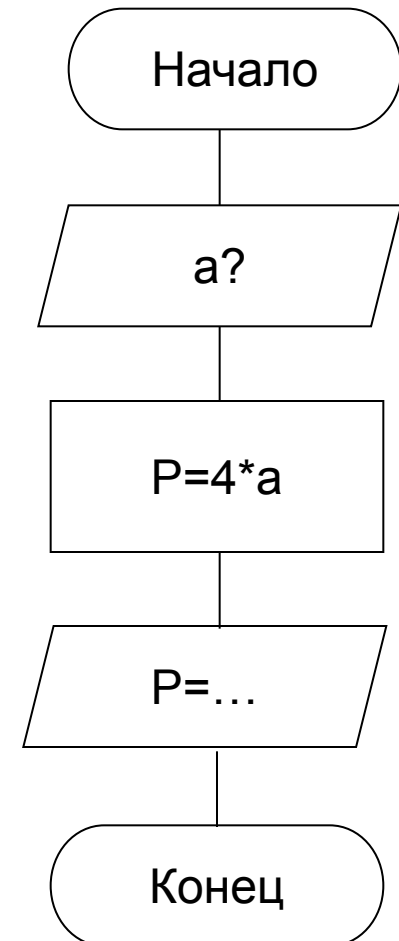
# Примеры решения задач лаб. №2, В-22, 3-22

## 1. Задание

Дана сторона квадрата  $a$ . Найти его периметр  $P = 4 \cdot a$

### Алгоритм решения:

- 1) Считываем значение стороны  $a$
- 2) Рассчитываем по формуле  $P=4 \cdot a$
- 3) Выводим результат  $P$



# Примеры решения задач лаб. №2, В-22, 3-22

```
1  #include <stdio.h>
2
3  main() {
4  float a, P;
5
6  printf("Задача №22. Дана сторона квадрата a. Найти его периметр P = 4·a.\n\n");
7  printf("Введите длину стороны квадрата,\n a=");
8  scanf("%f",&a);
9
10 P=4*a;
11
12 printf("Периметр квадрата со стороной a=%3.2f равен:\n %3.2f\n",a,P);
13
14 getch();
15
16 }
```

```
L:\_хсп\+эют\ яёюёрььшёютрэш \11.03.03 --ёюхёшёютрэшх | ТТ\ТхьУшш\ТхьУш ...
Задача №22. Дана сторона квадрата a. Найти его периметр P = 4·a.
Введите длину стороны квадрата,
a=25
Периметр квадрата со стороной a=25.00 равен:
100.00

-----
Process exited after 3.14 seconds with return value 0
Для продолжения нажмите любую клавишу . . .
```

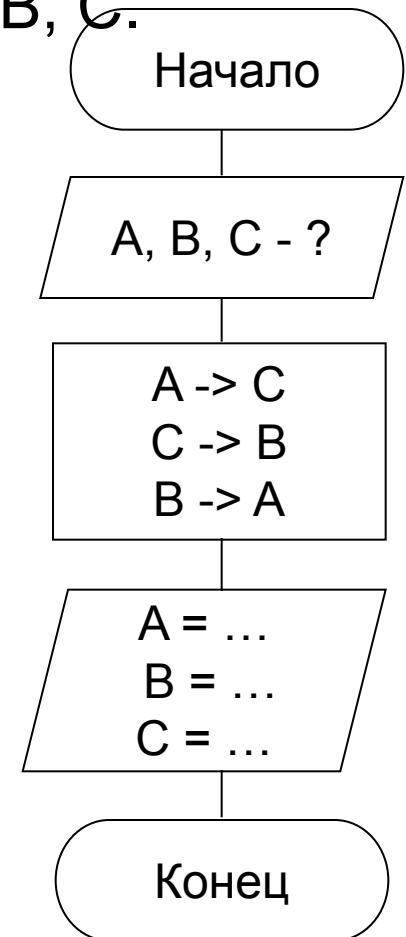
# Примеры решения задач лаб. №2, В-22, 3-44

## 2. Задание

Даны переменные  $A$ ,  $B$ ,  $C$ . Изменить их значения, переместив содержимое  $A$  в  $C$ ,  $C$  - в  $B$ ,  $B$  - в  $A$ , и вывести новые значения переменных  $A$ ,  $B$ ,  $C$ .

### Алгоритм решения:

- 1) Считываем значения  $A$ ,  $B$ ,  $C$
- 2) Меняем значения  $A$ ,  $B$  и  $C$  местами
- 3) Выводим результат  $A$ ,  $B$ ,  $C$

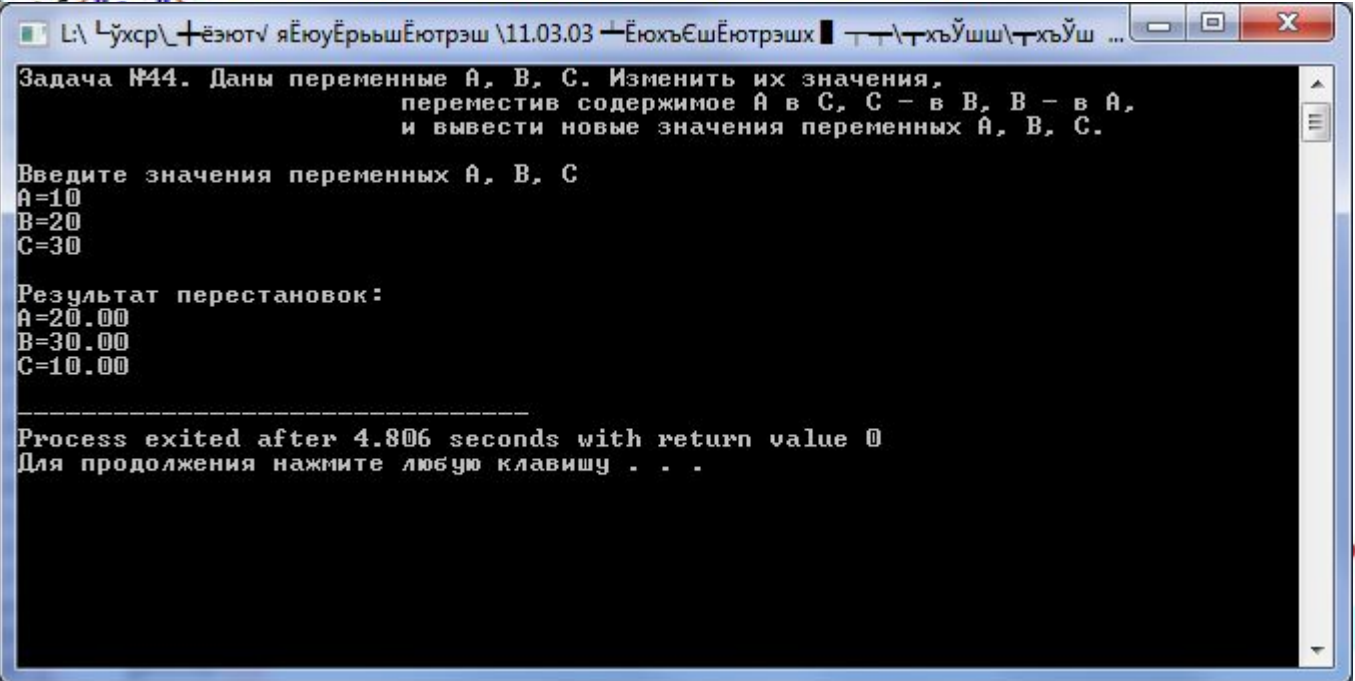


# Примеры решения задач лаб. №2, В-22, 3-44

```

1  #include <stdio.h>
2
3  main()
4  {
5      float a, b, c, d;
6
7      printf("Задача №44. Даны переменные A, B, C. Изменить их значения, \n\
8          .....   переместив содержимое A в C, C – в B, B – в A, \n\
9          .....   и вывести новые значения переменных A, B, C. \n\n");
10
11     printf("Введите значения переменных A, B, C \n");
12     pr
13     sca
14     pr
15     sca
16     pr
17     sca
18
19     d=a
20     a=b
21     b=c
22     c=0
23
24     pr
25
26     get
27 }

```



Задача №44. Даны переменные A, B, C. Изменить их значения,  
 переместив содержимое A в C, C – в B, B – в A,  
 и вывести новые значения переменных A, B, C.

Введите значения переменных A, B, C

A=10  
B=20  
C=30

Результат перестановок:

A=20.00  
B=30.00  
C=10.00

-----

Process exited after 4.806 seconds with return value 0  
 Для продолжения нажмите любую клавишу . . .

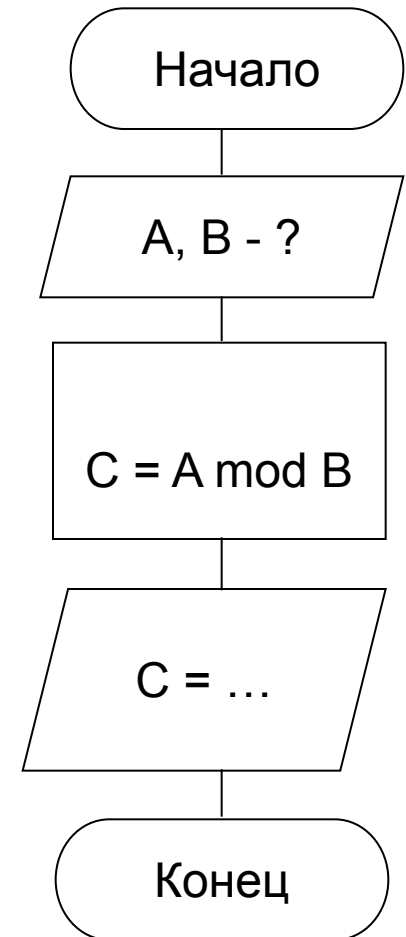
# Примеры решения задач лаб. №2, В-22, 3-66

## 3. Задание

Даны целые положительные числа  $A$  и  $B$  ( $A > B$ ). На отрезке длины  $A$  размещено максимально возможное количество отрезков длины  $B$  (без наложений). Используя операцию взятия остатка от деления нацело, найти длину незанятой части отрезка  $A$ .

### Алгоритм решения:

- 1) Считываем значения **A**, **B**
- 2) Находим длину незанятой части (**C**)
- 3) Выводим результат **C**



# Примеры решения задач лаб. №2, В-22, 3-66

```

1  #include <stdio.h>
2
3  main()
4  {
5      int a, b, c;
6
7      printf("Задача №66. Даны целые положительные числа A и B (A > B).\n"\
8             "На отрезке длины A размещено максимально возможное\n"\
9             "количество отрезков длины B (без наложений) \n");
10
11     Введите целое положительное число A
12     A=12
13     Введите целое положительное число B в диапазоне от 1 до 12
14     B=9
15
16     Длина незанятой части равна:
17     C=3
18
19     Process exited after 5.623 seconds with return value 0
20     Для продолжения нажмите любую клавишу . . . _
21
22
23
24  getchar();
25  }

```

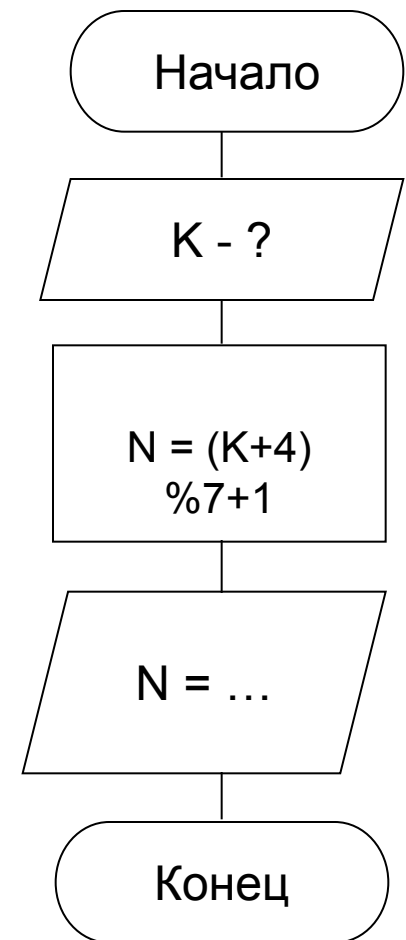
# Примеры решения задач лаб. №2, В-22, 3-88

## 4. Задание

Дни недели пронумерованы следующим образом: 1 - понедельник, 2 - вторник, ... , 6 - суббота, 7 - воскресенье. Дано целое число  $K$ , лежащее в диапазоне 1 - 365. Определить номер дня недели для  $K$ -го дня года, если известно, что в этом году 1 января было субботой.

### Алгоритм решения:

- 1) Считываем значение  $K$
- 2) Находим номер дня недели ( $N$ )
- 3) Выводим результат  $N$





# Примеры решения задач лаб. №2, В-22, 3-88

```

1  #include <stdio.h>
2
3  main()
4  {
5      int k, n;
6
7      printf("Задача №88. Дни недели пронумерованы следующим образом:\n"\  

8      "1 - понедельник, \n"\  

9      "2 - вторник, \n"\  

10     "3 - среда, \n"\  

11     "4 - четверг, \n"\  

12
13     Задача №88. Дни недели пронумерованы следующим образом:
14     1 - понедельник,
15     2 - вторник,
16     3 - среда,
17     4 - четверг,
18     5 - пятница,
19     6 - суббота,
20     7 - воскресенье.
21     Дано целое число K, лежащее в диапазоне 1 - 365.
22     Определить номер дня недели для K-го дня года,
23     если известно, что в этом году 1 января было субботой.
24     Номер дня года <K>
25     K=2
26     Номер дня недели равен:
27     N=7
28     -----
29     Process exited after 3.053 seconds with return value 0
30     Для продолжения нажмите любую клавишу . . .
31     \
32     \
33     \
34     \
35     \
36     \
37     \
38     \
39     \
40     \
41     \
42     \
43     \
44     \
45     \
46     \
47     \
48     \
49     \
50     \
51     \
52     \
53     \
54     \
55     \
56     \
57     \
58     \
59     \
60     \
61     \
62     \
63     \
64     \
65     \
66     \
67     \
68     \
69     \
70     \
71     \
72     \
73     \
74     \
75     \
76     \
77     \
78     \
79     \
80     \
81     \
82     \
83     \
84     \
85     \
86     \
87     \
88     \
89     \
90     \
91     \
92     \
93     \
94     \
95     \
96     \
97     \
98     \
99     \
100    \
101    \
102    \
103    \
104    \
105    \
106    \
107    \
108    \
109    \
110    \
111    \
112    \
113    \
114    \
115    \
116    \
117    \
118    \
119    \
120    \
121    \
122    \
123    \
124    \
125    \
126    \
127    \
128    \
129    \
130    \
131    \
132    \
133    \
134    \
135    \
136    \
137    \
138    \
139    \
140    \
141    \
142    \
143    \
144    \
145    \
146    \
147    \
148    \
149    \
150    \
151    \
152    \
153    \
154    \
155    \
156    \
157    \
158    \
159    \
160    \
161    \
162    \
163    \
164    \
165    \
166    \
167    \
168    \
169    \
170    \
171    \
172    \
173    \
174    \
175    \
176    \
177    \
178    \
179    \
180    \
181    \
182    \
183    \
184    \
185    \
186    \
187    \
188    \
189    \
190    \
191    \
192    \
193    \
194    \
195    \
196    \
197    \
198    \
199    \
200    \
201    \
202    \
203    \
204    \
205    \
206    \
207    \
208    \
209    \
210    \
211    \
212    \
213    \
214    \
215    \
216    \
217    \
218    \
219    \
220    \
221    \
222    \
223    \
224    \
225    \
226    \
227    \
228    \
229    \
230    \
231    \
232    \
233    \
234    \
235    \
236    \
237    \
238    \
239    \
240    \
241    \
242    \
243    \
244    \
245    \
246    \
247    \
248    \
249    \
250    \
251    \
252    \
253    \
254    \
255    \
256    \
257    \
258    \
259    \
260    \
261    \
262    \
263    \
264    \
265    \
266    \
267    \
268    \
269    \
270    \
271    \
272    \
273    \
274    \
275    \
276    \
277    \
278    \
279    \
280    \
281    \
282    \
283    \
284    \
285    \
286    \
287    \
288    \
289    \
290    \
291    \
292    \
293    \
294    \
295    \
296    \
297    \
298    \
299    \
300    \
301    \
302    \
303    \
304    \
305    \
306    \
307    \
308    \
309    \
310    \
311    \
312    \
313    \
314    \
315    \
316    \
317    \
318    \
319    \
320    \
321    \
322    \
323    \
324    \
325    \
326    \
327    \
328    \
329    \
330    \
331    \
332    \
333    \
334    \
335    \
336    \
337    \
338    \
339    \
340    \
341    \
342    \
343    \
344    \
345    \
346    \
347    \
348    \
349    \
350    \
351    \
352    \
353    \
354    \
355    \
356    \
357    \
358    \
359    \
360    \
361    \
362    \
363    \
364    \
365    \
366    \
367    \
368    \
369    \
370    \
371    \
372    \
373    \
374    \
375    \
376    \
377    \
378    \
379    \
380    \
381    \
382    \
383    \
384    \
385    \
386    \
387    \
388    \
389    \
390    \
391    \
392    \
393    \
394    \
395    \
396    \
397    \
398    \
399    \
400    \
401    \
402    \
403    \
404    \
405    \
406    \
407    \
408    \
409    \
410    \
411    \
412    \
413    \
414    \
415    \
416    \
417    \
418    \
419    \
420    \
421    \
422    \
423    \
424    \
425    \
426    \
427    \
428    \
429    \
430    \
431    \
432    \
433    \
434    \
435    \
436    \
437    \
438    \
439    \
440    \
441    \
442    \
443    \
444    \
445    \
446    \
447    \
448    \
449    \
450    \
451    \
452    \
453    \
454    \
455    \
456    \
457    \
458    \
459    \
460    \
461    \
462    \
463    \
464    \
465    \
466    \
467    \
468    \
469    \
470    \
471    \
472    \
473    \
474    \
475    \
476    \
477    \
478    \
479    \
480    \
481    \
482    \
483    \
484    \
485    \
486    \
487    \
488    \
489    \
490    \
491    \
492    \
493    \
494    \
495    \
496    \
497    \
498    \
499    \
500    \
501    \
502    \
503    \
504    \
505    \
506    \
507    \
508    \
509    \
510    \
511    \
512    \
513    \
514    \
515    \
516    \
517    \
518    \
519    \
520    \
521    \
522    \
523    \
524    \
525    \
526    \
527    \
528    \
529    \
530    \
531    \
532    \
533    \
534    \
535    \
536    \
537    \
538    \
539    \
540    \
541    \
542    \
543    \
544    \
545    \
546    \
547    \
548    \
549    \
550    \
551    \
552    \
553    \
554    \
555    \
556    \
557    \
558    \
559    \
560    \
561    \
562    \
563    \
564    \
565    \
566    \
567    \
568    \
569    \
570    \
571    \
572    \
573    \
574    \
575    \
576    \
577    \
578    \
579    \
580    \
581    \
582    \
583    \
584    \
585    \
586    \
587    \
588    \
589    \
590    \
591    \
592    \
593    \
594    \
595    \
596    \
597    \
598    \
599    \
600    \
601    \
602    \
603    \
604    \
605    \
606    \
607    \
608    \
609    \
610    \
611    \
612    \
613    \
614    \
615    \
616    \
617    \
618    \
619    \
620    \
621    \
622    \
623    \
624    \
625    \
626    \
627    \
628    \
629    \
630    \
631    \
632    \
633    \
634    \
635    \
636    \
637    \
638    \
639    \
640    \
641    \
642    \
643    \
644    \
645    \
646    \
647    \
648    \
649    \
650    \
651    \
652    \
653    \
654    \
655    \
656    \
657    \
658    \
659    \
660    \
661    \
662    \
663    \
664    \
665    \
666    \
667    \
668    \
669    \
670    \
671    \
672    \
673    \
674    \
675    \
676    \
677    \
678    \
679    \
680    \
681    \
682    \
683    \
684    \
685    \
686    \
687    \
688    \
689    \
690    \
691    \
692    \
693    \
694    \
695    \
696    \
697    \
698    \
699    \
700    \
701    \
702    \
703    \
704    \
705    \
706    \
707    \
708    \
709    \
710    \
711    \
712    \
713    \
714    \
715    \
716    \
717    \
718    \
719    \
720    \
721    \
722    \
723    \
724    \
725    \
726    \
727    \
728    \
729    \
730    \
731    \
732    \
733    \
734    \
735    \
736    \
737    \
738    \
739    \
740    \
741    \
742    \
743    \
744    \
745    \
746    \
747    \
748    \
749    \
750    \
751    \
752    \
753    \
754    \
755    \
756    \
757    \
758    \
759    \
760    \
761    \
762    \
763    \
764    \
765    \
766    \
767    \
768    \
769    \
770    \
771    \
772    \
773    \
774    \
775    \
776    \
777    \
778    \
779    \
780    \
781    \
782    \
783    \
784    \
785    \
786    \
787    \
788    \
789    \
790    \
791    \
792    \
793    \
794    \
795    \
796    \
797    \
798    \
799    \
800    \
801    \
802    \
803    \
804    \
805    \
806    \
807    \
808    \
809    \
810    \
811    \
812    \
813    \
814    \
815    \
816    \
817    \
818    \
819    \
820    \
821    \
822    \
823    \
824    \
825    \
826    \
827    \
828    \
829    \
830    \
831    \
832    \
833    \
834    \
835    \
836    \
837    \
838    \
839    \
840    \
841    \
842    \
843    \
844    \
845    \
846    \
847    \
848    \
849    \
850    \
851    \
852    \
853    \
854    \
855    \
856    \
857    \
858    \
859    \
860    \
861    \
862    \
863    \
864    \
865    \
866    \
867    \
868    \
869    \
870    \
871    \
872    \
873    \
874    \
875    \
876    \
877    \
878    \
879    \
880    \
881    \
882    \
883    \
884    \
885    \
886    \
887    \
888    \
889    \
890    \
891    \
892    \
893    \
894    \
895    \
896    \
897    \
898    \
899    \
900    \
901    \
902    \
903    \
904    \
905    \
906    \
907    \
908    \
909    \
910    \
911    \
912    \
913    \
914    \
915    \
916    \
917    \
918    \
919    \
920    \
921    \
922    \
923    \
924    \
925    \
926    \
927    \
928    \
929    \
930    \
931    \
932    \
933    \
934    \
935    \
936    \
937    \
938    \
939    \
940    \
941    \
942    \
943    \
944    \
945    \
946    \
947    \
948    \
949    \
950    \
951    \
952    \
953    \
954    \
955    \
956    \
957    \
958    \
959    \
960    \
961    \
962    \
963    \
964    \
965    \
966    \
967    \
968    \
969    \
970    \
971    \
972    \
973    \
974    \
975    \
976    \
977    \
978    \
979    \
980    \
981    \
982    \
983    \
984    \
985    \
986    \
987    \
988    \
989    \
990    \
991    \
992    \
993    \
994    \
995    \
996    \
997    \
998    \
999    \
1000   \
1001   \
1002   \
1003   \
1004   \
1005   \
1006   \
1007   \
1008   \
1009   \
1010   \
1011   \
1012   \
1013   \
1014   \
1015   \
1016   \
1017   \
1018   \
1019   \
1020   \
1021   \
1022   \
1023   \
1024   \
1025   \
1026   \
1027   \
1028   \
1029   \
1030   \
1031   \
1032   \
1033   \
1034   \
1035   \
1036   \
1037   \
1038   \
1039   \
1040   \
1041   \
1042   \
1043   \
1044   \
1045   \
1046   \
1047   \
1048   \
1049   \
1050   \
1051   \
1052   \
1053   \
1054   \
1055   \
1056   \
1057   \
1058   \
1059   \
1060   \
1061   \
1062   \
1063   \
1064   \
1065   \
1066   \
1067   \
1068   \
1069   \
1070   \
1071   \
1072   \
1073   \
1074   \
1075   \
1076   \
1077   \
1078   \
1079   \
1080   \
1081   \
1082   \
1083   \
1084   \
1085   \
1086   \
1087   \
1088   \
1089   \
1090   \
1091   \
1092   \
1093   \
1094   \
1095   \
1096   \
1097   \
1098   \
1099   \
1100   \
1101   \
1102   \
1103   \
1104   \
1105   \
1106   \
1107   \
1108   \
1109   \
1110   \
1111   \
1112   \
1113   \
1114   \
1115   \
1116   \
1117   \
1118   \
1119   \
1120   \
1121   \
1122   \
1123   \
1124   \
1125   \
1126   \
1127   \
1128   \
1129   \
1130   \
1131   \
1132   \
1133   \
1134   \
1135   \
1136   \
1137   \
1138   \
1139   \
1140   \
1141   \
1142   \
1143   \
1144   \
1145   \
1146   \
1147   \
1148   \
1149   \
1150   \
1151   \
1152   \
1153   \
1154   \
1155   \
1156   \
1157   \
1158   \
1159   \
1160   \
1161   \
1162   \
1163   \
1164   \
1165   \
1166   \
1167   \
1168   \
1169   \
1170   \
1171   \
1172   \
1173   \
1174   \
1175   \
1176   \
1177   \
1178   \
1179   \
1180   \
1181   \
1182   \
1183   \
1184   \
1185   \
1186   \
1187   \
1188   \
1189   \
1190   \
1191   \
1192   \
1193   \
1194   \
1195   \
1196   \
1197   \
1198   \
1199   \
1200   \
1201   \
1202   \
1203   \
1204   \
1205   \
1206   \
1207   \
1208   \
1209   \
1210   \
1211   \
1212   \
1213   \
1214   \
1215   \
1216   \
1217   \
1218   \
1219   \
1220   \
1221   \
1222   \
1223   \
1224   \
1225   \
1226   \
1227   \
1228   \
1229   \
1230   \
1231   \
1232   \
1233   \
1234   \
1235   \
1236   \
1237   \
1238   \
1239   \
1240   \
1241   \
1242   \
1243   \
1244   \
1245   \
1246   \
1247   \
1248   \
1249   \
1250   \
1251   \
1252   \
1253   \
1254   \
1255   \
1256   \
1257   \
1258   \
1259   \
1260   \
1261   \
1262   \
1263   \
1264   \
1265   \
1266   \
1267   \
1268   \
1269   \
1270   \
1271   \
1272   \
1273   \
1274   \
1275   \
1276   \
1277   \
1278   \
1279   \
1280   \
1281   \
1282   \
1283   \
1284   \
1285   \
1286   \
1287   \
1288   \
1289   \
1290   \
1291   \
1292   \
1293   \
1294   \
1295   \
1296   \
1297   \
1298   \
1299   \
1300   \
1301   \
1302   \
1303   \
1304   \
1305   \
1306   \
1307   \
1308   \
1309   \
1310   \
1311   \
1312   \
1313   \
1314   \
1315   \
1316   \
1317   \
1318   \
1319   \
1320   \
1321   \
1322   \
1323   \
1324   \
1325   \
1326   \
1327   \
1328   \
1329   \
1330   \
1331   \
1332   \
1333   \
1334   \
1335   \
1336   \
1337   \
1338   \
1339   \
1340   \
1341   \
1342   \
1343   \
1344   \
1345   \
1346   \
1347   \
1348   \
1349   \
1350   \
1351   \
1352   \
1353   \
1354   \
1355   \
1356   \
1357   \
1358   \
1359   \
1360   \
1361   \
1362   \
1363   \
1364   \
1365   \
1366   \
1367   \
1368   \
1369   \
1370   \
1371   \
1372   \
1373   \
1374   \
1375   \
1376   \
1377   \
1378   \
1379   \
1380   \
1381   \
1382   \
1383   \
1384   \
1385   \
1386   \
1387   \
1388   \
1389   \
1390   \
1391   \
1392   \
1393   \
1394   \
1395   \
1396   \
1397   \
1398   \
1399   \
1400   \
1401   \
1402   \
1403   \
1404   \
1405   \
1406   \
1407   \
1408   \
1409   \
1410   \
1411   \
1412   \
1413   \
1414   \
1415   \
1416   \
1417   \
1418   \
1419   \
1420   \
1421   \
1422   \
1423   \
1424   \
1425   \
1426   \
1427   \
1428   \
1429   \
1430   \
1431   \
1432   \
1433   \
1434   \
1435   \
1436   \
1437   \
1438   \
1439   \
1440   \
1441   \
1442   \
1443   \
1444   \
1445   \
1446   \
1447   \
1448   \
1449   \
1450   \
1451   \
1452   \
1453   \
1454   \
1455   \
1456   \
1457   \
1458   \
1459   \
1460   \
1461   \
1462   \
1463   \
1464   \
1465   \
1466   \
1467   \
1468   \
1469   \
1470   \
1471   \
1472   \
1473   \
1474   \
1475   \
1476   \
1477   \
1478   \
1479   \
1480   \
1481   \
1482   \
1483   \
1484   \
1485   \
1486   \
1487   \
1488   \
1489   \
1490   \
1491   \
1492   \
1493   \
1494   \
1495   \
1496   \
1497   \
1498   \
1499   \
1500   \
1501   \
1502   \
1503   \
1504   \
1505   \
1506   \
1507   \
1508   \
1509   \
1510   \
1511   \
1512   \
1513   \
1514   \
1515   \
1516   \
1517   \
1518   \
1519   \
1520   \
1521   \
1522   \
1523   \
1524   \
1525   \
1526   \
1527   \
1528   \
1529   \
1530   \
1531   \
1532   \
1533   \
1534   \
1535   \
1536   \
1537   \
1538   \
1539   \
1540   \
1541   \
1542   \
1543   \
1544   \
1545   \
1546   \
1547   \
1548   \
1549   \
1550   \
1551   \
1552   \
1553   \
1554   \
1555   \
1556   \
1557   \
1558   \
1559   \
1560   \
1561   \
1562   \
1563   \
1564   \
1565   \
1566   \
1567   \
1568   \
1569   \
1570   \
1571   \
1572   \
1573   \
1574   \
1575   \
1576   \
1577   \
1578   \
1579   \
1580   \
1581   \
1582   \
1583   \
1584   \
1585   \
1586   \
1587   \
1588   \
1589   \
1590   \
1591   \
1592   \
1593   \
1594   \
1595   \
1596   \
1597   \
1598   \
1599   \
1600   \
1601   \
1602   \
1603   \
1604   \
1605   \
1606   \
1607   \
1608   \
1609   \
1610   \
1611   \
1612   \
1613   \
1614   \
1615   \
1616   \
1617   \
1618   \
1619   \
1620   \
1621   \
1622   \
1623   \
1624   \
1625   \
1626   \
1627   \
1628   \
1629   \
1630   \
1631   \
1632   \
1633   \
1634   \
1635   \
1636   \
1637   \
1638   \
1639   \
1640   \
1641   \
1642   \
1643   \
1644   \
1645   \
1646   \
1647   \
1648   \
1649   \
1650   \
1651   \
1652   \
1653   \
1654   \
1655   \
1656   \
1657   \
1658   \
1659   \
1660   \
1661   \
1662   \
1663   \
1664   \
1665   \
1666   \
1667   \
1668   \
1669   \
1670   \
1671   \
1672   \
1673   \
1674   \
1675   \
1676   \
1677   \
1678   \
1679   \
1680   \
1681   \
1682   \
1683   \
1684   \
1685   \
1686   \
1687   \
1688   \
1689   \
1690   \
1691   \
1692   \
1693   \
1694   \
1695   \
1696   \
1697   \
1698   \
1699   \
1700   \
1701   \
1702   \
1703   \
1704   \
1705   \
1706   \
1707   \
1708   \
1709   \
1710   \
1711   \
1712   \
1713   \
1714   \
1715   \
1716   \
1717   \
1718   \
1719   \
1720   \
1721   \
1722   \
1723   \
1724   \
1725   \
1726   \
1727   \
1728   \
1729   \
1730   \
1731   \
1732   \
1733   \
1734   \
1735   \
1736   \
1737   \
1738   \
1739   \
1740   \
1741   \
1742   \
1743   \
1744   \
1745   \
1746   \
1747   \
1748   \
1749   \
1750   \
1751   \
1752   \
1753   \
1754   \
1755   \
1756   \
1757   \
1758   \
1759   \
1760   \
1761   \
1762   \
1763   \
1764   \
1765   \
1766   \
1767   \
1768   \
1769   \
1770   \
1771   \
1772   \
1773   \
1774   \
1775   \
1776   \
1777   \
1778   \
1779   \
1780   \
1781   \
1782   \
1783   \
1784   \
1785   \
1786   \
1787   \
1788   \
1789   \
1790   \
1791   \
1792   \
1793   \
1794   \
1795   \
1796   \
1797   \
1798   \
1799   \
1800   \
1801   \
1802   \
1803   \
1804   \
1805   \
1806   \
1807   \
1808   \
1809   \
1810   \
1811   \
1812   \
1813   \
1814   \
1815   \
1816   \
1817   \
1818   \
1819   \
1820   \
1821   \
1822   \
1823   \
1824   \
1825   \
1826   \
1827   \
1828   \
1829   \
1830   \
1831   \
1832   \
1833   \
1834   \
1835   \
1836   \
1837   \
1838   \
1839   \
1840   \
1841   \
1842   \
1843   \
1844   \
1845   \
1846   \
1847   \
1848   \
1849   \
1850   \
1851   \
1852   \
1853   \
1854   \
1855   \
1856   \
1857   \
1858   \
1859   \
1860   \
1861   \
1862   \
1863   \
1864   \
1865   \
1866   \
1867   \
1868   \
1869   \
1870   \
1871   \
1872   \
1873   \
1874   \
1875   \
1876   \
1877   \
1878   \
1879   \
1880   \
1881   \
1882   \
```

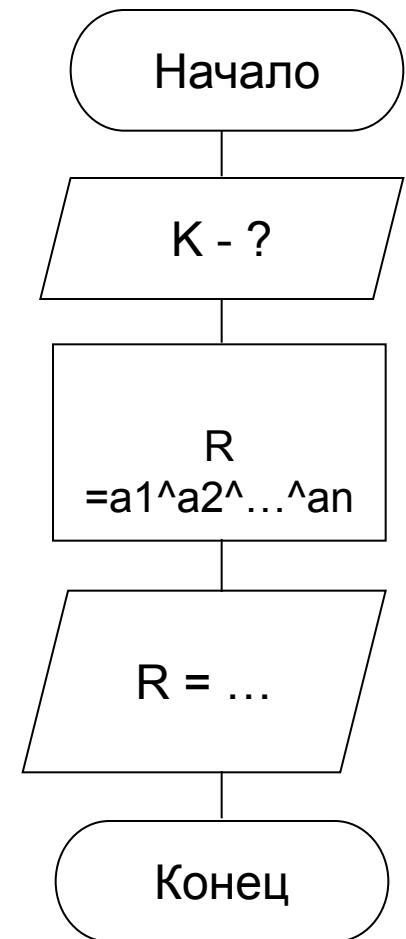
# Примеры решения задач лаб. №2, В-22, 3-110

## 5. Задание

Дано  $n$  чисел, каждое из которых встречается в последовательности два или кратное двум число раз, кроме одного, которое встречается нечетное число раз. Необходимо найти это число

### Алгоритм решения:

- 1) Считываем значения  $a_1 \dots a_n$
- 2) Находим искомое число (**R**)
- 3) Выводим результат **N**



# Примеры решения задач лаб. №2, В-22, 3-110

```

1  #include <stdio.h>
2
3  main()
4  {
5      int a1,a2,a3,a4,a5,a6,a7,R;
6
7      printf("Задача №10. Дано n чисел, каждое из которых встречается
8      в последовательности два или кратное двум число раз,
9      кроме одного, которое встречается нечетное число раз.
10     Необходимо найти это число.
11     Введите последовательность из 7 чисел,
12     при этом лишь одно число из последовательности
13     должно встречаться нечетное число раз.
14     Искомое число равно:
15     1 5 3 2 1 3 5
16     2
17     -----
18     Process exited after 20.37 seconds with return value 0
19     Для продолжения нажмите любую клавишу . . . _
20
21     printf("Искомое число равно:\n %d",R);
22
23     getch();
24 }

```

# Основы программирования (на языке Си)

## Тема 8. Ветвления “разбор полетов”

# Примеры решения задач лаб. №3, В-22, 3-22

## 1. Задание

Даны координаты точки, не лежащей на координатных осях  $OX$  и  $OY$ . Определить номер координатной четверти, в которой находится данная точка.

### Алгоритм решения:

1) Считываем координаты  $X$  и  $Y$

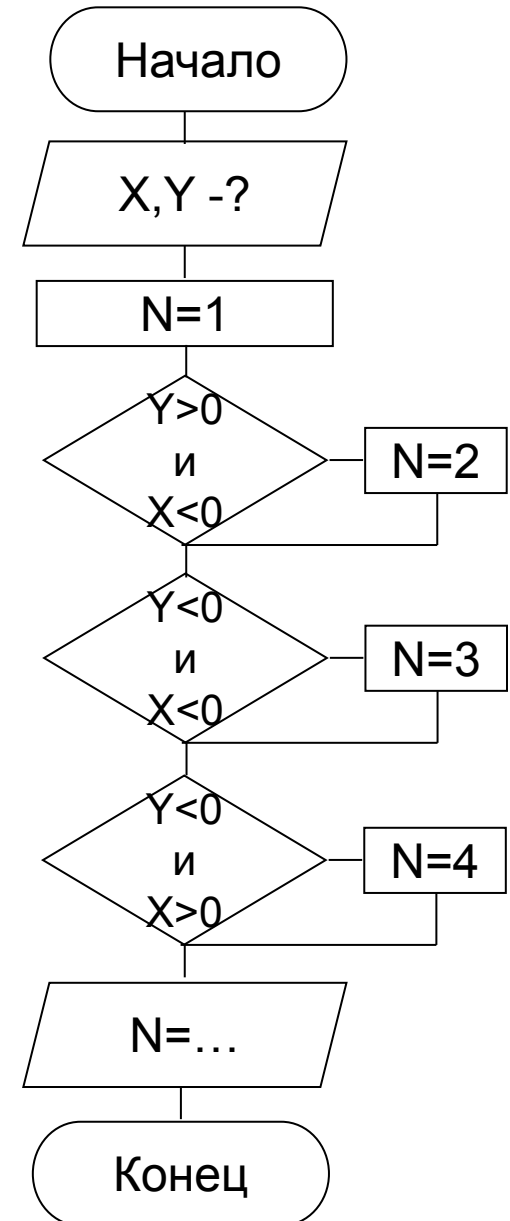
2)  $N=1$

если  $Y > 0$  и  $X < 0$  то  $N=2$

если  $Y < 0$  и  $X < 0$  то  $N=3$

если  $Y < 0$  и  $X > 0$  то  $N=4$

3) Выводим результат  $N$



# Примеры решения задач лаб. №3, В-22, 3-22

```

1  #include <stdio.h>
2
3  main()
4  {
5      float x,y;
6      int n;
7
8      printf("Даны координаты точки, не лежащей на координатных осях OX и OY.\n");
9      printf("0 Определить номер координатной четверти, в которой находится данная точка.\n\n");
10     printf("Введите координаты X и Y\n");
11     scanf("%f %f", &x, &y);
12     printf("Точка находится в четверти №:\n");
13     scanf("%d", &n);
14     printf("-----\n");
15     printf("Process exited after 13.77 seconds with return value 0\n");
16     printf("Для продолжения нажмите любую клавишу . . . \n");
17
18     if (x > 0 && y > 0)
19         n = 1;
20     else if (x < 0 && y > 0)
21         n = 2;
22     else if (x < 0 && y < 0)
23         n = 3;
24     else if (x > 0 && y < 0)
25         n = 4;

```

```

y.\n"\  

очка.\n\n");

```

```

getchar();

```

# Примеры решения задач лаб. №3, В-22, 3-44

## 2. Задание

Мастям игральных карт присвоены порядковые номера: 1 - пики, 2 - трефы, 3 - бубны, 4 - червы. Достоинству карт, старших десятки, присвоены номера: 11 - валет, 12 - дама, 13 - король, 14 - туз.

Даны два целых числа:

$N$  - достоинство ( $6 \leq N \leq 14$ ) и  $M$  - масть карты ( $1 \leq M \leq 4$ ).

Вывести название соответствующей карты вида «шестерка бубен», «дама червей», «туз треф» и т. п.

### Алгоритм решения:

1) Считываем значения  $M$  и

2) Если  $N$  принимает значение

6: выводим «Шестерка»;

7: выводим «Семерка»;

8: выводим «Восьмерка»;

9: выводим «Девятка»;

10: выводим «Десятка»;

11: выводим «Валет»;

12: выводим «Дама»;

13: выводим «Король»;

14: выводим «Туз».

3) Если  $M$  принимает значение

1: выводим «пики»;

2: выводим «трефы»;

3: выводим «бубны»;

4: выводим «червы».



Что забыли?

Если  $M$  или  $N$  некорректные, выводим «Нет такой карты».

# Примеры решения задач лаб. №3, В-22, 3-44

```
1  #include <stdio.h>
2
3  main()
4  {
5      int m,n;
6
7      printf("Мастям игральных карт присвоены порядковые номера M:\n"\
8      "1 - пики;\n"\
9      "2 - трефы;\n"\
10     "3 - бубны;\n"\
11     "4 - червы.\n"\
12     "Достоинству карт, старших десятки, присвоены номера N:\n"\
13     "11 - валет;\n"\
14     "12 - дама;\n"\
15     "13 - король;\n"\
16     "14 - туз.\n"\
17     "Даны два целых числа:\n"\
18     "N - достоинство (6 <= N <= 14) и M - масть карты (1 <= M <= 4).\n"\
19     "Вывести название соответствующей карты вида «шестерка бубен», «дама червей», и т. п.\n\n");
20
21     printf("Введите номер масти M и достоинства карты N\n");
22     printf("Масть, M=");
23     scanf("%d", &m);
24     printf("\nДостоинство, N=");
25     scanf("%d", &n);
26     printf("\nВаша карта:\n");
27
```



# Примеры решения задач лаб. №3, В-22, 3-44

```

28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50

```

```

if ((m<=4&&м>=1) && (n<=14&&n>=6))
{
    switch (n)
    {
        case 6: printf("Шестерка ");
                break;
        case 7: printf("Семерка ");
                break;
        case 8: printf("Восьмерка ");
                break;
        case 9: printf("Девятка ");
                break;
    }
}

```

```

51
52
53
54

```

```

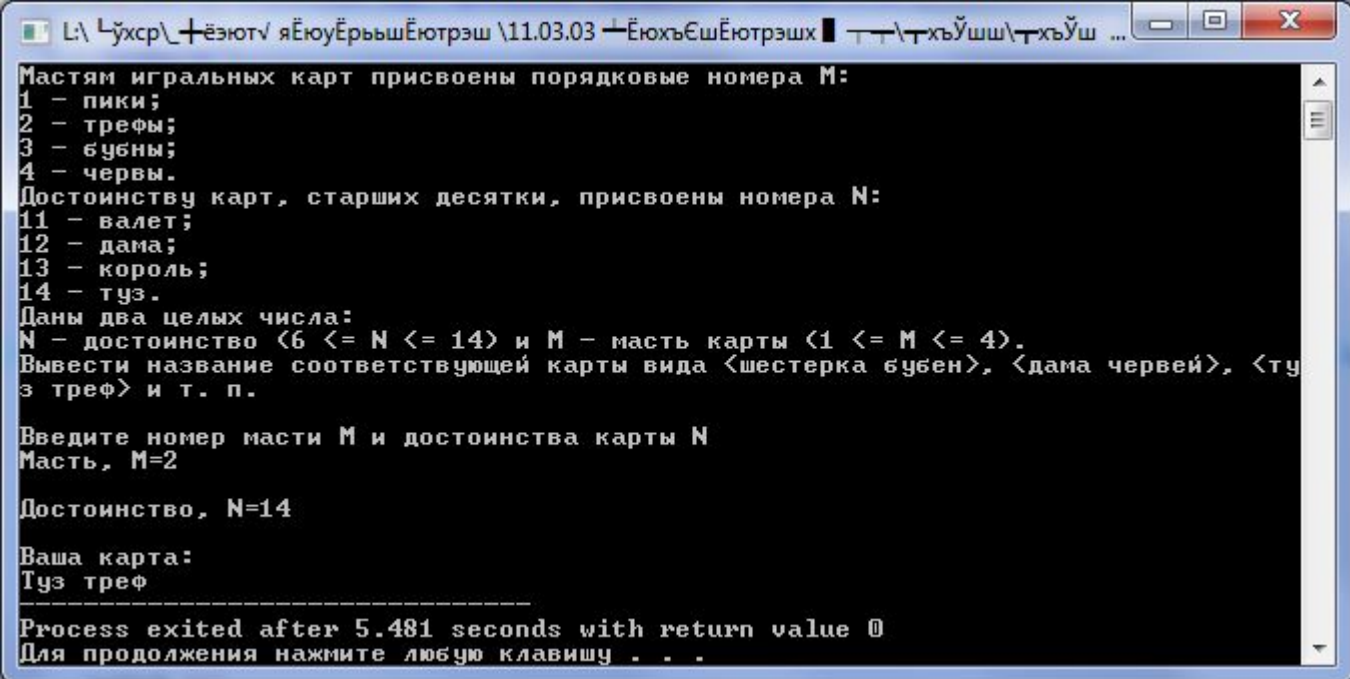
switch (m)
{
    case 1: printf("пик");
            break;
}

```

```

printf(" ");
printf("бубен");
printf("червей");
printf("ты!\n");

```



```

L:\Ухсп\+эют\ яёюёрьшёютрэш \11.03.03 -ёюхёшёютрэш ...
Мастяи игральных карт присвоены порядковые номера М:
1 - пики;
2 - трефы;
3 - бубны;
4 - червы.
Достоинству карт, старших десятки, присвоены номера N:
11 - валет;
12 - дама;
13 - король;
14 - туз.
Даны два целых числа:
N - достоинство (6 <= N <= 14) и M - масть карты (1 <= M <= 4).
Вывести название соответствующей карты вида <шестерка бубен>, <дама червей>, <туз треф> и т. п.

Введите номер масти M и достоинства карты N
Масть, M=2
Достоинство, N=14

Ваша карта:
Туз треф

Process exited after 5.481 seconds with return value 0
Для продолжения нажмите любую клавишу . . .

```

# Примеры решения задач лаб. №3, В-22, 3-66

---

## 3. Задание

Дан номер некоторого года (положительное целое число). Вывести число дней в этом году, учитывая, что обычный год насчитывает 365 дней, а високосный — 366 дней.

Високосным считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400 (например, годы 300, 1300 и 1900 не являются високосными, а 1200 и 2000 — являются).

## Алгоритм решения:

1) Считываем

2) Если <sup>год</sup> не делится на 4, значит он обычный.

Иначе надо проверить не делится ли год на 100.

3) Если не делится, значит это не столетие и можно сделать вывод, что год високосный.

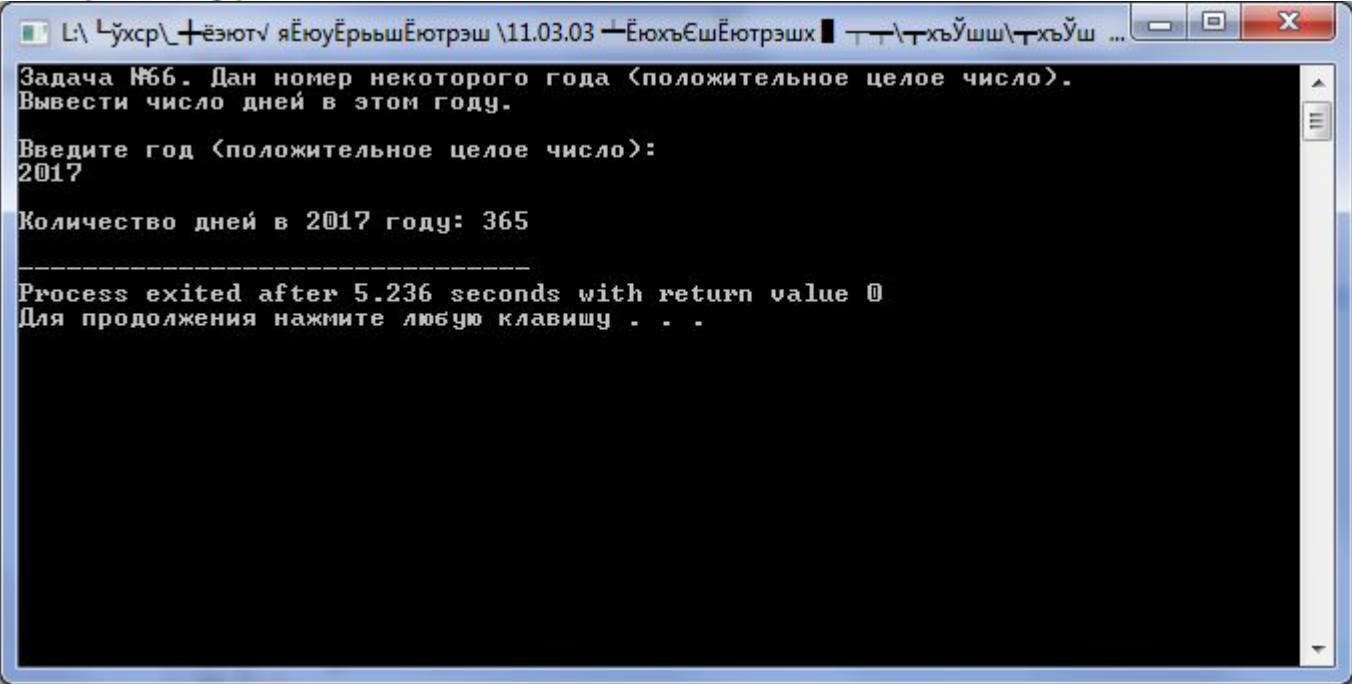
4) Если делится на 100, значит это столетие и его следует проверить его делимость на 400.

5) Если год делится на 400, то он високосный. Иначе год обычный.

6) Выводим результат

# Примеры решения задач лаб. №3, В-22, 3-66

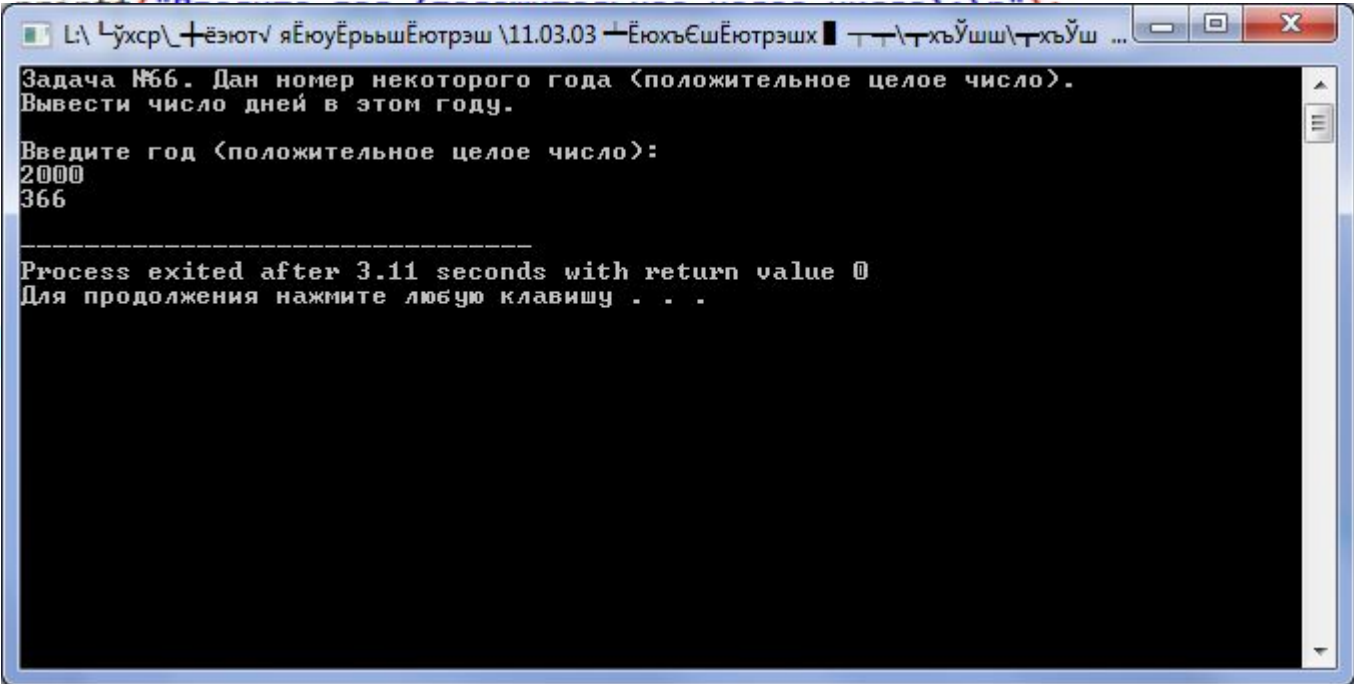
```
1  #include <stdio.h>
2
3  main()
4  {
5      int y;
6
7      printf("Задача №66. Дан номер некоторого года (положительное целое число).\n"\
8             "Вывести число дней в этом году.\n\n");
9
10     printf("Введите год (положительное целое число):\n");
11     scanf("%d",&y);
12
13     i
14     e
15     .....
16     p
17     p
18     g
19     g
20 }
```



```
Задача №66. Дан номер некоторого года (положительное целое число).
Вывести число дней в этом году.
Введите год (положительное целое число):
2017
Количество дней в 2017 году: 365
-----
Process exited after 5.236 seconds with return value 0
Для продолжения нажмите любую клавишу . . .
```

# Примеры решения задач лаб. №3, В-22, 3-66

```
1  #include <stdio.h>
2
3  main()
4  {
5      int y;
6
7      printf("Задача №66. Дан номер некоторого года (положительное целое число).\n"\
8             "Вывести число дней в этом году.\n\n");
9
10
11
12
13
14
15
16
17
18
19 }
```



Задача №66. Дан номер некоторого года (положительное целое число).  
Вывести число дней в этом году.

Введите год (положительное целое число):  
2000  
366

-----  
Process exited after 3.11 seconds with return value 0  
Для продолжения нажмите любую клавишу . . .

# Примеры решения задач лаб. №3, В-22, 3-88

## 4. Задание

Даны три числа  $a$ ,  $b$ ,  $c$ . Определить, какое из них равно  $d$ .

Если ни одно не равно  $d$ , то найти  $\max(d-a, d-b, d-c)$

### Алгоритм решения:

1) Считываем четыре числа  $a$ ,  $b$ ,  $c$  и  $d$ .

2) Если ( $a$  не равно  $d$ ) и ( $b$  не равно  $d$ ) и ( $c$  не равно  $d$ ), считаем:

$$da = d - a$$

$$db = d - b$$

$$dc = d - c$$

2.1) Если  $da > db$  и  $da > dc$   
выводим значение  $da$

2.2) иначе если  $db > dc$   
выводим значение

2.3)  <sup>$db$</sup>  иначе выводим  $dc$

3) иначе (см. п.2)

если  $a = d$  выводим: "Число  $a = d$ "

если  $b = d$  выводим: "Число  $b = d$ "

если  $c = d$  выводим: "Число  $c = d$ "

# Примеры решения задач лаб. №3, В-22, 3-88

```
1  #include <stdio.h>
2
3  main()
4  {
5      float a, b, c, d, da, db, dc, max;
6
7      printf("Задача №88. Даны три числа a, b, c. Определить, какое из них равно d.\n"\
8             "Если ни одно не равно d, то найти max(d-a, d-b, d-c).\n\n");
9
10     printf("Введите числа a, b, c и d:\n");
11     scanf("%f %f %f %f",&a,&b,&c,&d);
12
13     if ((a!=d)&&(b!=d)&&(c!=d))
14     {
15         da=d-a;
16         db=d-b;
17         dc=d-c;
18
19         if ((da>db)&&(da>dc))
20             max=da;
21         else
22             if (db>dc)
23                 max=db;
24             else
25                 max=dc;
26
27         printf("Максимум разности равен %3.2f\n", max);
28
29     }
30     else
31     {
32         if (a==d)
33             printf("Число a = d\n");
34         if (b==d)
35             printf("Число b = d\n");
36         if (c==d)
37             printf("Число c = d\n");
38     }
39
40     getchar();
41 }
```

# Примеры решения задач лаб. №3, В-22, 3-110

---

## 5. Задание

Сможет ли покупатель, располагающий суммой **S** рублей, приобрести **a** граммов конфет по цене **b** рублей за 1 кг?

### Алгоритм решения:

- 1) Считываем четыре числа  $a$ ,  $b$  и  $S$ .
- 2) Если  $b \cdot a / 1000 \leq S$  то выводим “Сможет!”  
иначе выводим “Не сможет!”



# Примеры решения задач лаб. №3, В-22, 3-110

```

1  #include <stdio.h>
2
3  main()
4  {
5      float S, a, b;
6
7      printf("Задача №110. Может ли покупатель, располагающий суммой S рублей,\n"
8             "приобрести a граммов конфет по цене b рублей за 1 кг?\n\n");
9
10     printf("Введите массу конфет (в граммах):\n");
11     S = 500;
12     P: printf("Задача №110. Может ли покупатель, располагающий суммой S рублей,\n"
13             "приобрести a граммов конфет по цене b рублей за 1 кг?");
14     P: printf("Введите массу конфет (в граммах):");
15     S = 500;
16     P: printf("Введите стоимость 1 кг конфет(в рублях):");
17     S = 120;
18     P: printf("Введите сумму планируемой покупки (в рублях):");
19     S = 35;
20     P: printf("Не сможет!\n");
21     P: printf("-----\n");
22     P: printf("Process exited after 10.25 seconds with return value 0\n");
23     P: printf("Для продолжения нажмите любую клавишу . . .\n");
24     P: printf("i");
25     P: printf("-----\n");
26     P: printf("Process exited after 10.25 seconds with return value 0\n");
27     P: printf("Для продолжения нажмите любую клавишу . . .\n");
28     P: printf("i");

```

