

1. подключить assert.h
2. подключить complex.h
3. подключить wchar.h
4. подключить wctype.h
5. подключить ctype.h
6. подключить errno.h
7. подключить fenv.h
8. подключить wchar.h
9. подключить wctype.h
10. подключить limits.h
11. подключить locale.h
12. подключить math.h
13. подключить stdlib.h
14. подключить string.h
15. подключить math.h
16. подключить setjmp.h
17. подключить signal.h
18. подключить stdarg.h
19. подключить stdbool.h
20. подключить stddef.h
21. подключить stdint.h
22. подключить float.h
23. подключить inttypes.h
24. подключить iso646.h
25. подключить stdio.h
26. подключить stdlib.h
27. подключить string.h
28. подключить stdio.h
29. подключить tgmath.h
30. подключить time.h
- 31.
32. заменить sizeof(short) на special_float
33. заменить _y__e на printf
34. заменить double на int
35. заменить lalka на "%.3f %.2f"
36. заменить amount_of_pogtan на 20
37. заменить byte на void
38. заменить a(x, y) на x##y
- 39.
40. определить тип sphere_double как double
41. определить тип sphere_int как int
- 42.
43. определить функцию KOREA_ZE_BEST, возвращающую тип byte, принимающую параметры KIM, имеющий тип указателя на char, CHEN, имеющий тип указателя на char, bIH, имеющий тип double как:
(начало определения функции)
44. Присвоить переменной n, имеющей тип register double, результат следующего выражения : bIH прибавить 7 и всё это разделить на 8
45. Делаем выбор по значению выражения (остаток при делении bIH на 8) :
46. Если значение равно 0, то цикл:
47. (начало цикла)
48. указателю KIM присвоить инкрементированный указатель CHEN
49. Если значение равно 7, то: указателю KIM присвоить постинкрементированный указатель CHEN
50. Если значение равно 6, то: указателю KIM присвоить постинкрементированный указатель CHEN
51. Если значение равно 5, то: указателю KIM присвоить постинкрементированный указатель CHEN
52. Если значение равно 4, то: указателю KIM присвоить постинкрементированный указатель CHEN
- 53.

54. Если значение равно 3, то: указателю KIM присвоить постинкрементированный указатель CHEN
55. Если значение равно 2, то: указателю KIM присвоить постинкрементированный указатель CHEN
56. Если значение равно 1, то: указателю KIM присвоить постинкрементированный указатель CHEN
57. (конец цикла) до тех пор, пока декрементированный n больше нуля
58. (конец определения функции)
- 59.
60. определить тип float_16_type как unsigned long long double
61. отменить замену double
62. заменить double на int
63. заменить тип int4_type на double
64. отмена замены int
65. заменить int на double
- 66.
67. статический массив aye размера amount_of_pogtan объединений состоит из полей:
68. (начало определения объединения)
69. структура asdfg состоящая из полей:
70. (начало определения структуры)
71. переменная x типа sphere_double
72. переменная y типа sphere_int
73. переменная special_float типа int4_type
74. переменная value_t типа unsigned long long int
75. переменная secret_value типа size_t
76. (конец определения структуры)
77. переменная undefined_value типа float_16_type
78. (конец определения объединения)
- 79.
80. определить функцию skrrrrrrra, возвращающую тип byte, принимающую параметры i, имеющий тип int, j, имеющий тип int как :
81. (начало определения функции)
82. x из структуры asdfg i-го элемента массива aye присвоить его побитово ИЛИ с x из структуры asdfg j-го элемента массива aye, которому присвоить его побитово ИЛИ с x из структуры asdfg i-го элемента массива aye , которому присвоить его побитово ИЛИ с x из структуры asdfg j-го элемента массива aye
83. y из структуры asdfg i-го элемента массива aye присвоить его побитово ИЛИ с y из структуры asdfg j-го элемента массива aye, которому присвоить его побитово ИЛИ с y из структуры asdfg i-го элемента массива aye , которому присвоить его побитово ИЛИ с y из структуры asdfg j-го элемента массива aye
84. (конец определения функции)
- 85.
86. определить функцию triangulation, возвращающую тип byte, принимающую параметры Ri_Sol_ju, имеющий тип int, r0zmer, имеющий тип int как :
87. (начало определения функции)
88. до тех пор пока выражение (Ri_Sol_ju умножить на 2 и ко всему этому прибавить 1) меньше чем r0zmer :
89. (начало цикла)
90. Присвоить переменной Ju_ae, имеющей тип int, результат следующего выражения : Ri_Sol_ju умножить на 2 и ко всему этому прибавить 1
91. если ((Ju_ae + 1) строго меньше r0zmer) и (aye[Ju_ae].asdfg.sizeof(short) строго меньше aye[Ju_ae + 1].asdfg.sizeof(short)) то постинкрементировать переменную Ju_ae
92. если aye[Ri_Sol_ju].asdfg.sizeof(short) меньше aye[Ju_ae].asdfg.sizeof(short) то:
93. (начало)
94. вызвать функцию skrrrrrrra с передаваемыми аргументами Ju_ae и Ri_Sol_ju
95. присвоить переменной Ri_Sol_ju значение Ju_ae
96. (конец)
97. иначе ничего не возвращать
98. (конец цикла)

```

99.          (конец определения функции)
100.
101.          определить функцию tobi_zvizda, возвращающую тип byte, принимающую
            параметр r0zmer, имеющий тип int как :
102.          (начало определения функции)
103.          В цикле по переменной i, имеющей тип int, с начальным значением, равным
            (r0zmer деленый на 2), проходящей условие (i больше либо равен 0), инкрементирующей-
            ся на каждом шаге:
104.          вызвать функцию triangulation с передаваемыми аргументами i и r0zmer
105.          (конец определения функции)
106.
107.          определить функцию chto_pogbalhada, возвращающую тип byte, прини-
            мающую параметр r0zmer, имеющий тип double как :
108.          (начало определения функции)
109.          Присвоить переменной zvizda_r0zmer, имеющей тип int, значение r0zmer
110.          вызвать функцию tobi_zvizda с передаваемым аргументом zvizda_r0zmer
111.          До тех пор пока (zvizda_r0zmer - 1) больше нуля:
112.          (начало цикла)
113.          вызвать функцию skrrrrrrra с передаваемыми аргументами 0 и zvizda_r0zmer -1
114.          декрементировать переменную zvizda_r0zmer
115.          вызвать функцию triangulation с передаваемыми аргументами 0 и
            zvizda_r0zmer
116.          (конец цикла)
117.          (конец определения функции)
118.
119.          функция main типа double с передаваемым аргументом типа byte:
120.          (начало функции)
121.          пеерменным массива chto_pogtan типа float_16_type размера 20 присвоить
            значения:
122.          378730216230580,
123.          374250565325306,
124.          161576669731263,
125.          130163278932509,
126.          263753941702520,
127.          499904128523255,
128.          260373802442241,
129.          -532575893205,
130.          -514257909139635,
131.          -317763155353815,
132.          490107308103530,
133.          -474439267340872,
134.          -524677499809229,
135.          -305342109932169,
136.          57453777571110,
137.          10067403390680,
138.          209353885931976,
139.          54176717527434,
140.          -137245679887725,
141.          356868832679485
142.          В цикле по переменной i, имеющей тип int, с начальным значением, равным
            (amount_of_pogtan - 1), проходящей условие(i больше либо равен 0), декрементирующей-
            ся на каждом шаге :
143.          (начало цикла)
144.          переменной aye[i].undefined_value присвоить значение chto_pogtan[i]
145.          вызвать функцию KOREA_ZE_BEST с передаваемыми аргументами
            (char*)&(int4_type*)((char*)&(aye[i].asdfg.sizeof(short)) + 0x0) и (char*)((cht0_pogtan + i)) и 010
146.          (конец цикла)
147.          вызвать функцию chto_pogbalhada с передаваемым аргументом
            amount_of_pogtan
148.          a(_y_, __e)(lalka, (float)aye[18].asdfg.x / 1000, (float)aye[18].asdfg.y / 1000)

```

149. вернуть 0
150. (конец функции)