

```
1  /*-----*/
2  Copyright (c) 2018 Author: Jagadeesh Vasudevamurthy
3  file: p1test.cpp
4
5  On linux:
6  g++ p1.cpp p1test.cpp
7  valgrind a.out
8
9  -----*/
10
11 /*-----*/
12         YOU CANNOT CHANGE ANYTHING IN THIS FILE
13 -----*/
14
15 /*-----*/
16 This file test p1 object
17 -----*/
18
19 /*-----*/
20 All includes here
21 -----*/
22 #include "p1.h"
23
24 /*-----*/
25 print n empty lines
26 -----*/
27 void emptylines(int n) {
28     for (int i = 0; i < n ; ++i) {
29         cout << endl;
30     }
31 }
32
33 /*-----*/
34 test bed
35 -----*/
36 void testbed() {
37     const int n = 3 ;
38     p1 a ;
39
40     a.print_usa() ;
41     emptylines(n) ;
42     a.print_n_n2_n3();
43     emptylines(n) ;
44     a.a_power_b();
45     emptylines(n) ;
46     a.two_power_n();
47     emptylines(n) ;
48     a.a1(3,8);
49     emptylines(n) ;
```

```

50     a.a2(3,8);
51     emptylines(n) ;
52     a.a3(3,8);
53     emptylines(n) ;
54     a.a4(3,8);
55     emptylines(n) ;
56 }
57
58 /*-----
59 main
60 -----*/
61 int main() {
62     testbed() ;
63     return 0 ;
64 }
65
66 #if 0
67 /*
68 XXX   XXX   XXXX X   XXX
69 X     X X   XX     X
70 X     X X   X     X X
71 X     X X           X X
72 X     X   XXX     X  X
73 X     X     XX    X  X
74 X     X       X   XXXXX
75 X     X X   X X   X
76 XX  XX XX   X X   X
77 XXX   X XXXX  XXX   XXX
78
79
80
81 n     n^2   n^3
82 1     1     1
83 2     4     8
84 3     9    27
85 4    16    64
86 5    25   125
87 6    36   216
88 7    49   343
89 8    64   512
90 9    81   729
91
92
93
94 a     b           a^b
95 1     2           1
96 2     3           8
97 3     4           81
98 4     5          1024

```

```
99 5 6 15625
100 6 7 279936
101 7 8 5764801
102
103
104
105 n 2^n
106 0 1
107 1 2
108 2 4
109 3 8
110 4 16
111 5 32
112 6 64
113 7 128
114 8 256
115 9 512
116 10 1024
117 11 2048
118 12 4096
119 13 8192
120 14 16384
121 15 32768
122 16 65536
123 17 131072
124 18 262144
125 19 524288
126 20 1048576
127
128
129
130 3
131 3 4
132 3 4 5
133 3 4 5 6
134 3 4 5 6 7
135 3 4 5 6 7 8
136
137
138
139 3
140 4 3
141 5 4 3
142 6 5 4 3
143 7 6 5 4 3
144 8 7 6 5 4 3
145
146
147
```

```
148 3 4 5 6 7 8
149 3 4 5 6 7
150 3 4 5 6
151 3 4 5
152 3 4
153 3
154
155
156
157 3 4 5 6 7 8
158 3 4 5 6 7
159 3 4 5 6
160 3 4 5
161 3 4
162 3
163
164 */
165 #endif
166
167 //EOF
168
```