Problem 6.25.2. Implement a **integer matrix class**, as shown in the figure 6.38. You are given the test program in **intmatrix1test.cpp**.

VASUDEVAMURTHY

Copyright © Jagadeesh Vasudevamurthy - Do not photocopy.

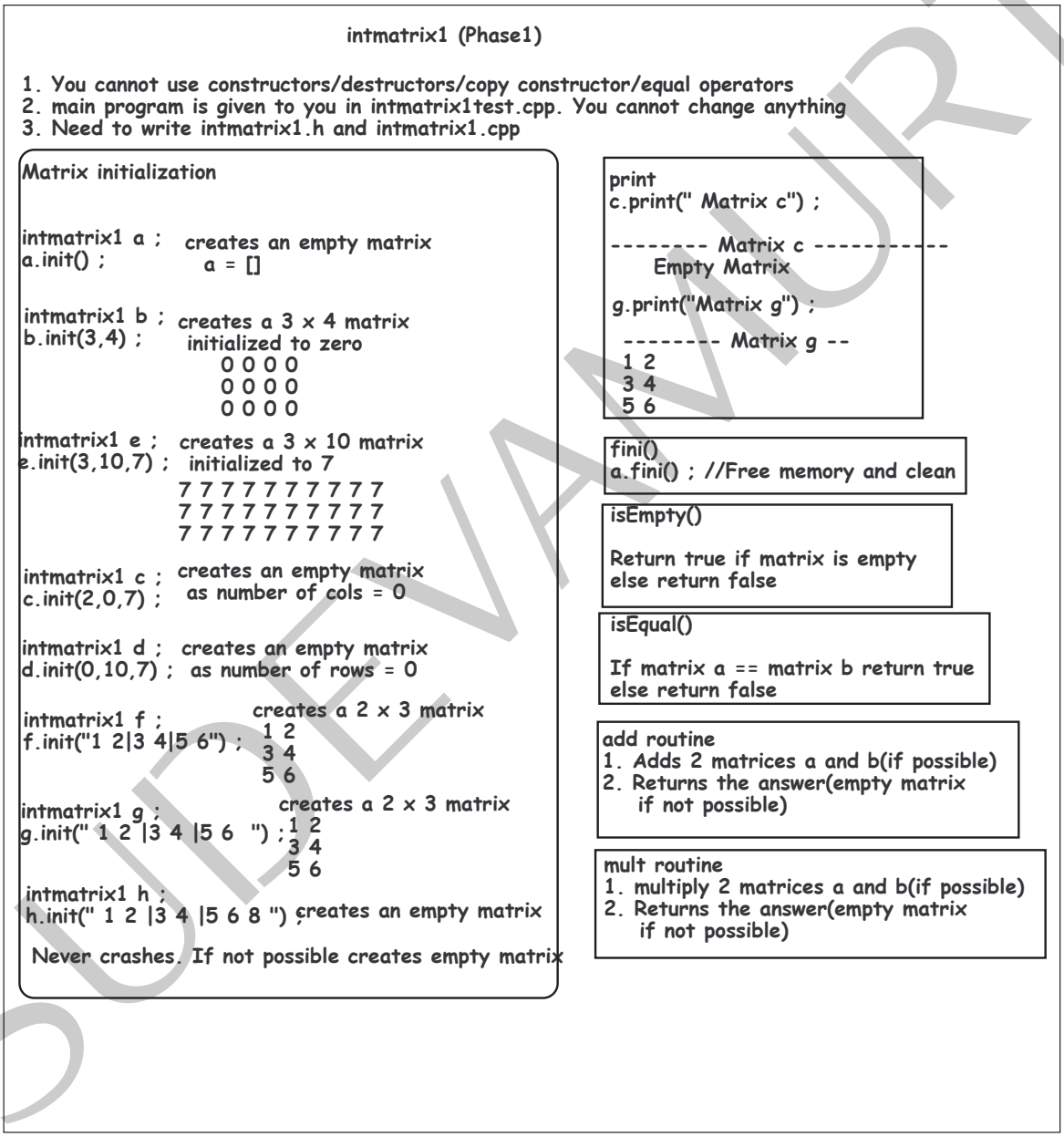

Figure 6.38: Integer matrix class(Phase 1)

VASUDEVAMURTHY

Copyright © Jagadeesh Vasudevamurthy - Do not photocopy.

```
/*-------------------------------------------------------------------
Copyright (c) 2013 Author: Jagadeesh Vasudevamurthy
file: intmatrix1test.cpp

On linux:
g++ intmatrix1.cpp intmatrix1test.cpp
valgrind a.out

-------------------------------------------------------------------*/

/*-------------------------------------------------------------------
This file test intmatrix1 object
-------------------------------------------------------------------*/

/*-------------------------------------------------------------------
All includes here
-------------------------------------------------------------------*/
#include "intmatrix1.h"

/*-------------------------------------------------------------------
test init and fini
-------------------------------------------------------------------*/
void test_init_fini() {
  intmatrix1 a ;
  a.init() ;
  a.print("Matrix a") ;

  intmatrix1 b ;
  b.init(3,4) ;
  b.print("Matrix b") ;

  intmatrix1 c ;
  c.init(2,0,7) ;
  c.print("Matrix c") ;

  intmatrix1 d ;
  d.init(0,10,7) ;
  d.print("Matrix d") ;

  intmatrix1 e ;
  e.init(3,10,7) ;
  e.print("Matrix e") ;


  intmatrix1 f ;
  f.init("1 2|3 4|5 6") ;
  f.print("Matrix e") ;

  intmatrix1 g ;
  g.init(" 1 2 |3 4 |5 6  ") ;
  g.print("Matrix g") ;
  assert(f.isEqual(g)) ;
  assert(g.isEqual(f)) ;

  intmatrix1 h ;
  h.init(" 1 2 |3 4 |5 6 8 ") ;
  h.print("Matrix h") ;
  assert(h.isEmpty()) ;
  assert(!f.isEqual(h)) ;

  a.fini() ;
  b.fini() ;
  c.fini() ;
  d.fini() ;
  e.fini() ;
  f.fini() ;
```

JAGADEESH VASUDEVAMURTHY

```
67    g.fini() ;
68  }
69
70  /*-----------------------------------------------------------------
71  test add1
72  -----------------------------------------------------------------*/
73  void test_add1(const char* as, const char* bs, const char* anss) {
74    intmatrix1 a ;
75    a.init(as) ;
76    a.print("Matrix a") ;
77    intmatrix1 b ;
78    b.init(bs) ;
79    b.print("Matrix b") ;
80    intmatrix1 s = a.add(b) ;
81    s.print("matrix s") ;
82    intmatrix1 ans;
83    ans.init(anss) ;
84    ans.print("matrix expected ans") ;
85    assert(s.isEqual(ans)) ;
86    assert(ans.isEqual(s)) ;
87    a.fini() ;
88    b.fini() ;
89    s.fini() ;
90    ans.fini() ;
91  }
92
93  /*-----------------------------------------------------------------
94  test add
95  -----------------------------------------------------------------*/
96  void test_add() {
97    test_add1("7 9 11|13 15 17 "," 6 8 10| 12 14 16 ","13 17 21 | 25 29 33") ;
98    test_add1("1 2 3|4 5  6 ","1 2  ","") ;
99  }
100
101 /*-----------------------------------------------------------------
102 test mult1
103 -----------------------------------------------------------------*/
104 void test_mult1(const char* as, const char* bs, const char* anss) {
105   intmatrix1 a ;
106   a.init(as) ;
107   a.print("Matrix a") ;
108   intmatrix1 b ;
109   b.init(bs) ;
110   b.print("Matrix b") ;
111   intmatrix1 s = a.mult(b) ;
112   s.print("matrix s") ;
113   intmatrix1 ans;
114   ans.init(anss) ;
115   ans.print("matrix expected ans") ;
116   assert(s.isEqual(ans)) ;
117   assert(ans.isEqual(s)) ;
118   a.fini() ;
119   b.fini() ;
120   s.fini() ;
121   ans.fini() ;
122   cout <<"---------------------------------\n" ;
123 }
124
125 /*-----------------------------------------------------------------
126 test mult
127 -----------------------------------------------------------------*/
128 void test_mult() {
129   test_mult1("1 2 3"," 2 1 3 | 3 3 2| 4 1 2  ","20 10 13") ;
130   test_mult1("3 4 2","13 9 7 15|8 7 4 6| 6 4 0 3 ","83 63 37 75") ;
131   test_mult1("3","5 2 11|9 4 14","15 6 33|27 12 |42") ;
132   const char* a = "3 9 0 2 2 9 5 2|0 2 2 1 9 6 6 8|7 5 6 1 4 9 8 9|3 3 2 9 2 1 7 4|1 9 0 1 2 9 5 2|4 2 ↙
```

```
  0 3 7 3 9 1|5 9 0 6 6 7 8 2|9 3 4 6 8 4 9 1" ;
133   const char* b = "6 1 6 0 8 3 0 0|6 8 9 0 6 6 7 2|4 8 2 0 5 4 6 7|2 4 4 2 2 6 9 8|4 8 2 2 4 6 4 1|1 5 ↙
      5 6 4 7 5 5|7 4 6 5 0 6 5 3|2 3 7 0 1 3 8 5";
134   const char* s = "132 170 200  87 128 186 175 106|122 186 166  86  92 182 195 123|197 235 267 104 179 ↙
      243 253 178|128 140 164  63  86 162 194 140|118 164 184  85 110 174 166  98|138 142 144  83  91 162 ↙
      137  82|187 222 244 106 160 244 232 141|201 212 210  97 171 230 204 142";
135   test_mult1(a,b,s) ;
136   test_mult1("7 3|2 5 | 6 8| 9 0","8 14 0 3 1|7 11 5 91 3|8 4 19 5 57","") ;
137 }
138
139 /*---------------------------------------------------------------
140 test bed
141 ---------------------------------------------------------------*/
142 void testbed() {
143   test_init_fini();
144   test_add() ;
145   test_mult() ;
146 }
147
148 /*---------------------------------------------------------------
149 main
150 ---------------------------------------------------------------*/
151 int main() {
152   testbed() ;
153   return 0 ;
154 }
155
156 //EOF
157
158
159
```

ASUDEVAMURTHY