

VG100 — Introduction to Engineering

Project 1 Report (Team 16)

Rubric

- Game Design (10 pts)
- Code Quality (50 pts)
- Readme (15 pts)
- Personal work (20 pts)

1 Game Design

Not included in this report.

2 Code Quality

Your total score of this part is 24/50.

All related information is listed below:

1 point(s) **deduction**, hard-coded contents, in file [Main.elm](#), lines 227-232.

```
227    [[1,2,3],[1,2,4],[1,2,5],[1,3,2],[1,3,4],[1,3,5],[1,4,2],[1,4,3],[1,4,5],[1,5,2],[1,5,3],[1,5,4]
228     ,[2,1,3],[2,1,4],[2,1,5],[2,3,1],[2,3,4],[2,3,5],[2,4,1],[2,4,3],[2,4,5],[2,5,1],[2,5,3],[2,5,4]
229     ,[3,1,2],[3,1,4],[3,1,5],[3,2,1],[3,2,4],[3,2,5],[3,4,1],[3,4,2],[3,4,5],[3,5,1],[3,5,2],[3,5,4]
230     ,[4,1,2],[4,1,3],[4,1,5],[4,2,1],[4,2,3],[4,2,5],[4,3,1],[4,3,2],[4,3,5],[4,5,1],[4,5,2],[4,5,3]
231     ,[5,1,2],[5,1,3],[5,1,4],[5,2,1],[5,2,3],[5,2,4],[5,3,1],[5,3,2],[5,3,4],[5,4,1],[5,4,2],[5,4,3]
232    ]
```

2 point(s) **deduction**, duplicate code and hard-coded contents, in file [Main.elm](#), lines 259-502.

```
259    initBricks1 : List Brick
260    initBricks1 =
261        [ { position = ( 55, 298 ) , status = 1 }
262          , { position = ( 136, 298 ) , status = 1 }
263          , { position = ( 217, 298 ) , status = 1 }
264          , { position = ( 298, 298 ) , status = 1 }
265          , { position = ( 379, 298 ) , status = 2 }
266          , { position = ( 460, 298 ) , status = 4 }
267          , { position = ( 541, 298 ) , status = 2 }
268          , { position = ( 622, 298 ) , status = 1 }
269
270        ...
493    , { position = ( 784, 236 ) , status = 0 }
494    , { position = ( 784, 205 ) , status = 0 }
495    , { position = ( 784, 174 ) , status = 2 }
496    , { position = ( 865, 236 ) , status = 2 } --coll1
497    , { position = ( 865, 205 ) , status = 5 }
498    , { position = ( 865, 174 ) , status = 0 }
499    , { position = ( 865, 143 ) , status = 5 }
500    , { position = ( 865, 112 ) , status = 0 }
501    , { position = ( 865, 81 ) , status = 5 }
502    ]
```

1 point(s) **deduction**, too many nested List.append, use ++ instead, in file [Main.elm](#), lines 529-560.

```
529         (List.append
530         (List.append
531         (List.append
532         (List.append
533         (List.append
534         [dispBackground --This box will be under white screen
535         , dispWhite
536         , dispWhite2 model
537         , dispCeiling
538         , dispRightWall
539
540         ...
541
551         , displayPaddle model.paddle]
552         )
553         (displayBricks model.brick)
554         )
555         (displayItems model.items)
556         )
557         (displayBall model.ball)
558         )
559         (displayBall model.laserball)
560         )
```

2 point(s) **deduction**, duplicate code, in file [Main.elm](#), lines 580-807.

```
580 dispBackground : Svg.Svg Msg
581 dispBackground =
582     Svg.image
583     [ x "54"
584     , y "49"
585     , width "892"
586     , height "551"
587     , xlinkHref "https://s1.ax1x.com/2020/06/17/NAPkRJ.jpg"
588     ] []
589     --
590
591 ...
592
798 if stageA 5 model then
799     Svg.image
800     [ x "760"
801     , y "200"
802     , width "150"
803     , height "150"
804     , xlinkHref "https://s1.ax1x.com/2020/06/17/NEQIBT.png"
805     ] []
806 else
807     Svg.rect [] []
```

1 point(s) **deduction**, duplicate code, in file [Main.elm](#), lines 894-1005.

```
894 stage1Button : Model -> Html Msg
895 stage1Button model =
```

```

896     let
897         (txt , (x,y), msg) =
898             if model.gameStatus == (Playing<|NextStage) then
899                 ("Entrance Corridor", ("970px", "200px") , ChooseStage(Stage1))
900             else
901                 ("", ("100px", "100px") , Backup)
902
903         (width,height) =
904
905     ...
906
907         ("160px", "40px")
908     else
909         ("0px", "0px")
910
911     color =
912         if stageA 5 model then
913             "#1B358E"
914         else
915             "#740505"
916
917     in
918     makeButton txt (x,y) (width,height) color ("18px", "#ffff") msg

```

1 point(s) deduction, duplicate code, in file [Main.elm](#), lines 1007-1117.

```

1007     bonus1Button : Model -> Html Msg
1008     bonus1Button model =
1009         let
1010             (txt , (x,y), msg) =
1011                 if model.gameStatus == (Playing<|BonusSelection) then
1012                     ("Treasure 1", ("970px", "200px") , ChooseBonus 1)
1013                 else
1014                     ("", ("100px", "100px") , Backup)
1015
1016             (width,height) =
1017
1018         ...
1019
1020             ("160px", "40px")
1021         else
1022             ("0px", "0px")
1023
1024         color =
1025             if bonusA 5 model then
1026                 "#1B358E"
1027             else
1028                 "#740505"
1029
1030         in
1031         makeButton txt (x,y) (width,height) color ("18px", "#ffff") msg

```

1 point(s) deduction, duplicate code, in file [Main.elm](#), lines 1173-1245.

```

1173     7 -> Svg.rect
1174     [ x <| String.fromFloat (brick.position |> Tuple.first)
1175       , y <| String.fromFloat (brick.position |> Tuple.second)
1176       , width "80"
1177       , height "30"
1178       , fill "#679008"

```

```

1179     ]
1180     []
1181
1182     --short
...
1236     []
1237
1238     _ -> Svg.rect
1239         [ x <| String.fromFloat (brick.position |> Tuple.first)
1240           , y <| String.fromFloat (brick.position |> Tuple.second)
1241           , width "80"
1242           , height "30"
1243           , fill "#ffb52b"
1244         ]
1245     []

```

1 point(s) deduction, duplicate code, in file [Main.elm](#), lines 1250-1303.

```

1250     0 ->      -- effect: shorter paddle
1251     Svg.rect
1252         [ x <| String.fromFloat item.x
1253           , y <| String.fromFloat item.y
1254           , width "20"
1255           , height "20"
1256           , fill "#d94040" --#d94040
1257         ]
1258     []
1259     1 ->      -- effect: longer paddle
...
1294     []
1295     _ ->      -- effect: none
1296     Svg.rect
1297         [ x <| String.fromFloat item.x
1298           , y <| String.fromFloat item.y
1299           , width "20"
1300           , height "20"
1301           , fill "#404040" --#40404020
1302         ]
1303     []

```

1 point(s) deduction, duplicate code, in file [Main.elm](#), lines 1428-1447.

```

1428     stage1 =
1429         if List.member 1 stage then
1430             "Entrance Corridor/ "
1431         else ""
1432     stage2 =
1433         if List.member 2 stage then
1434             "Antechamber/ "
1435         else ""
1436     stage3 =
1437         if List.member 3 stage then

```

```

1438     "Annex/ "
1439     else ""
1440 stage4 =
1441     if List.member 4 stage then
1442         "Treasury/ "
1443     else ""
1444 stage5 =
1445     if List.member 5 stage then
1446         "Burial Chamber"
1447     else ""

```

1 point(s) **deduction**, too many if and else if, should use Type, in file [Main.elm](#), lines 1453-1460.

```

1453 if model.extraballactivation then
1454     "Extra Ball"
1455 else if model.fireballactivation then
1456     "Fire Ball"
1457 else if model.laseractivation then
1458     "Laser Ball"
1459 else
1460     "Nothing"

```

1 point(s) **deduction**, too many if and else if, should use Type, in file [Main.elm](#), lines 1506-1516.

```

1506 if model.twisttime >= 1 then
1507     String.fromInt (model.twisttime // 60)
1508 else if model.longertime >= 1 then
1509     String.fromInt (model.longertime // 60)
1510 else if model.shortertime >= 1 then
1511     String.fromInt (model.shortertime // 60)
1512 else if model.transparenttime >= 1 then
1513     String.fromInt (model.transparenttime // 60)
1514 else if model.biggertime >= 1 then
1515     String.fromInt (model.biggertime // 60)
1516 else ""

```

2 point(s) **deduction**, too long function with many duplicate code, in file [Main.elm](#), lines 1529-2002.

```

1529 update msg model =
1530     case msg of
1531         Backup ->
1532             ( model , Cmd.none )
1533
1534         NewRandomBonus number ->
1535             let
1536                 bonuslist =
1537                     case model.bonuslist of
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
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002

```

```

1998         , paddle = initPaddle
1999         , paddleMove = NotMoving
2000         , gameStatus = (Playing <| Waiting)
2001         , score_init = model.score
2002         , life_init = model.life

```

1 point(s) deduction, too long function, in file [Main.elm](#), lines 2015-2136.

```

2015 calculateLaser : Ball -> Msg -> Model -> List Ball -> (Model, List Ball)
2016 calculateLaser ball __ model newList=
2017     let
2018         bricks =
2019             model.brick
2020
2021         shouldDeleteHorizontalBrick =
2022             case Tuple.first (Tuple.second (clearBricks_Horizontal ball bricks)) of
2023                 0 -> False
2024                 _ -> True
2025
2026     ...
2127     | brick = updateBricks
2128     , twistTime = twistTime
2129     , longTime = longTime
2130     , shortTime = shortTime
2131     , score = score + updateScores
2132     , life = life + maybeExtraLife
2133     , messageTime = messageTime
2134     }
2135     , list
2136 )

```

1 point(s) deduction, too long function, in file [Main.elm](#), lines 2140-2306.

```

2140 calculate ball __ model newList=
2141     let
2142         bricks =
2143             model.brick
2144
2145         shouldDeleteHorizontalBrick =
2146             case Tuple.first (Tuple.second (clearBricks_Horizontal ball bricks)) of
2147                 0 -> False
2148                 _ -> True
2149
2150     ...
2297     , shortTime = shortTime
2298     , transparentTime = transparentTime
2299     , bigTime = bigTime
2300     , score = score + updateScores
2301     , life = life + maybeExtraLife
2302     , messageTime = messageTime
2303     }
2304     , list
2305     )
2306

```

1 point(s) deduction, duplicate code, in file [Main.elm](#), lines 2308-2336.

```
2308 twist : Model -> Ball -> Int -> Bool
2309 twist model ball num =
2310     if (Tuple.second (Tuple.second (clearBricks_Horizontal ball model.brick))) .twist == True
2311         || (Tuple.second (Tuple.second (clearBricks_Vertical ball model.brick))) .twist == True
2312         || (Tuple.second (Tuple.second (clearBricks_Corner ball model.brick))) .twist == True
2313         || Tuple.second (Tuple.second (clearItems ball model.items)) == 2 && num == 1 then
2314         True
2315     else
2316         False
2317
...
2327
2328 shorter : Model -> Ball -> Int -> Bool
2329 shorter model ball num =
2330     if (Tuple.second (Tuple.second (clearBricks_Horizontal ball model.brick))) .short == True
2331         || (Tuple.second (Tuple.second (clearBricks_Vertical ball model.brick))) .short == True
2332         || (Tuple.second (Tuple.second (clearBricks_Corner ball model.brick))) .short == True
2333         || Tuple.second (Tuple.second (clearItems ball model.items)) == 0 && num == 1 then
2334         True
2335     else
2336         False
```

1 point(s) deduction, duplicate code, in file [Main.elm](#), lines 2627-2704.

```
2627 detectBrickHorizontalCollision : Ball -> List Brick -> Int -> Maybe Int
2628 detectBrickHorizontalCollision ball bricks num=
2629     case bricks of
2630     [] ->
2631         Nothing
2632     cell :: rest ->
2633         let
2634             brick_x =
2635                 Tuple.first cell.position
2636             brick_y =
2637
...
2695         ((ball.y - brick_y - 30)^2 + (ball.x - brick_x - 80)^2 <= (ball.r)^2) && (ball.y > brick_y + 30) && (ball.x >
2696             ↪ brick_x + 80) && (ball.verticalSpeed < 0) && (ball.horizontalSpeed < 0)
2697     status3 =
2698         ((ball.y - brick_y - 30)^2 + (ball.x - brick_x)^2 <= (ball.r)^2) && (ball.y > brick_y + 30) && (ball.x < brick_x)
2699         ↪ && (ball.verticalSpeed < 0) && (ball.horizontalSpeed > 0)
2700     status4 =
2701         ((ball.y - brick_y)^2 + (ball.x - brick_x - 80)^2 <= (ball.r)^2) && (ball.y < brick_y) && (ball.x > brick_x + 80)
2702         ↪ && (ball.verticalSpeed > 0) && (ball.horizontalSpeed < 0)
2703     in
2704     if status1 || status2 || status3 || status4 then
2705         Just num
2706     else
2707         detectBrickCornerCollision ball rest (num+1)
```

1 point(s) deduction, should define Type instead of using int, in file [Main.elm](#), lines 2736-2740.

```

2736         0 -> (newbricks,(droppedbricks + 1,twistbricks))
2737         7 -> (newbricks,(droppedbricks + 1,{twistbricks | extralife = True}))
2738         6 -> (newbricks,(droppedbricks + 1,{twistbricks | short = True}))
2739         5 -> (newbricks,(droppedbricks + 1,{twistbricks | long = True}))
2740         4 -> (newbricks,(droppedbricks + 1,{twistbricks | twist = True}))

```

1 point(s) **deduction**, duplicate code, in file [Main.elm](#), lines 2719-2808.

```

2719 clearBricks_Horizontal : Ball -> List Brick -> (List Brick, (Int, Brickfunction))
2720 clearBricks_Horizontal ball bricks =
2721     case detectBrickHorizontalCollision ball bricks 0 of
2722         Nothing ->
2723             (bricks, (0, { twist = False , long = False , short = False , extralife = False }))
2724         Just num ->
2725             let
2726                 before = List.take num bricks
2727                 after = List.drop (num+1) bricks
2728                 self = List.drop num bricks
2729
2730             ...
2799         6 -> (newbricks,(droppedbricks + 1,{twistbricks | short = True}))
2800         5 -> (newbricks,(droppedbricks + 1,{twistbricks | long = True}))
2801         4 -> (newbricks,(droppedbricks + 1,{twistbricks | twist = True}))
2802         _ ->
2803             let
2804                 newbrick = {position = brick.position , status = brick.status - 1}
2805             in
2806                 (newbrick :: newbricks,(droppedbricks + 1, twistbricks))
2807
2808         Nothing -> (newbricks,(droppedbricks , twistbricks))

```

2 point(s) **deduction**, elm-stuff in git repo.

5 point(s) **deduction**, all code in one file.

2 point(s) **bonus**, some comments found.

3 Readme

Not included in this report.

4 Personal work

Not included in this report.