```
TESTING:
p1_graph1
Finding Maximum bandwidth path between source = 1697 and target = 2922.

      Found Maximum bandwidth path WITHOUT heap in 0.119 seconds
      No. of Hops: 71
      Path: 1697 -> 1407 -> 1459 -> 4610 -> 221 -> 907 -> 1912 -> 416 -> 1586
-> 2122 -> 556 -> 2728 -> 3565 -> 2718 -> 786 -> 1860 -> 1772 -> 2307 -> 3366
-> 1581 -> 702 -> 812 -> 39 -> 2191 -> 930 -> 2515 -> 2263 -> 2779 -> 91 ->
2700 -> 616 -> 1142 -> 1958 -> 1256 -> 2020 -> 2008 -> 3066 -> 606 -> 1181 ->
1712 -> 285 -> 1458 -> 2404 -> 2762 -> 3323 -> 348 -> 1731 -> 1049 -> 1396 ->
3164 -> 275 -> 1706 -> 251 -> 446 -> 1091 -> 1151 -> 1038 -> 3277 -> 2575 ->
357 -> 1059 -> 3055 -> 63 -> 86 -> 844 -> 257 -> 229 -> 1760 -> 1951 -> 2019
-> 746 -> 2922
      Maximum Bandwidth: 3427

      Found Maximum bandwidth path WITH heap in 0.048 seconds
      No. of Hops: 32
      Path: 1697 -> 1407 -> 1459 -> 4610 -> 221 -> 907 -> 1912 -> 416 -> 1586
-> 2122 -> 556 -> 2728 -> 3565 -> 4407 -> 4455 -> 3483 -> 303 -> 70 -> 4077 -
> 4840 -> 1788 -> 1663 -> 4015 -> 1199 -> 101 -> 1285 -> 2168 -> 3363 -> 1003
-> 4849 -> 1066 -> 746 -> 2922
      Maximum Bandwidth: 3427

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.067 seconds
      No. of Hops: 79
      Path: 1697 -> 1407 -> 1459 -> 4610 -> 221 -> 579 -> 2453 -> 2557 -> 80
-> 4448 -> 3206 -> 2920 -> 1176 -> 282 -> 3916 -> 4865 -> 930 -> 1208 -> 3118
-> 664 -> 4968 -> 1953 -> 4572 -> 4502 -> 3263 -> 2758 -> 4662 -> 1997 -> 404
-> 3379 -> 1410 -> 2900 -> 4661 -> 910 -> 2279 -> 4352 -> 3399 -> 1857 ->
3971 -> 3742 -> 1675 -> 2175 -> 3088 -> 4766 -> 2038 -> 3196 -> 1337 -> 3499
-> 2727 -> 2887 -> 2500 -> 4115 -> 1448 -> 76 -> 3589 -> 820 -> 4013 -> 2263
-> 2779 -> 2180 -> 3112 -> 2248 -> 3997 -> 2391 -> 137 -> 332 -> 4873 -> 1119
-> 303 -> 3483 -> 2600 -> 308 -> 4372 -> 2588 -> 4243 -> 85 -> 63 -> 2334 ->
746 -> 2922
      Maximum Bandwidth: 3427

Finding Maximum bandwidth path between source = 486 and target = 2398.

      Found Maximum bandwidth path WITHOUT heap in 0.083 seconds
      No. of Hops: 69
      Path: 486 -> 2937 -> 1095 -> 4440 -> 1626 -> 2681 -> 1857 -> 3162 ->
2946 -> 1846 -> 1495 -> 1269 -> 1357 -> 391 -> 3073 -> 3220 -> 54 -> 2232 ->
513 -> 847 -> 1884 -> 402 -> 2611 -> 2504 -> 2920 -> 1552 -> 982 -> 129 ->
1457 -> 479 -> 400 -> 1087 -> 2748 -> 15 -> 2556 -> 1376 -> 857 -> 56 -> 2104
-> 2225 -> 2474 -> 1761 -> 2437 -> 3030 -> 2602 -> 2550 -> 2203 -> 330 ->
1515 -> 1025 -> 1207 -> 1295 -> 1076 -> 1321 -> 2002 -> 1882 -> 3136 -> 1770
-> 2869 -> 828 -> 2755 -> 2290 -> 1140 -> 1151 -> 3866 -> 4207 -> 384 -> 2855
-> 3732 -> 2398
      Maximum Bandwidth: 3798

      Found Maximum bandwidth path WITH heap in 0.017 seconds
      No. of Hops: 40
      Path: 486 -> 2937 -> 1095 -> 4440 -> 1626 -> 2681 -> 1857 -> 3399 ->
4352 -> 3941 -> 3209 -> 1585 -> 3297 -> 2827 -> 1586 -> 2122 -> 556 -> 2728 -
> 3565 -> 4407 -> 4455 -> 3483 -> 2600 -> 308 -> 2259 -> 2990 -> 248 -> 4961
```

```
-> 4307 -> 1059 -> 357 -> 2575 -> 3277 -> 1038 -> 1151 -> 3866 -> 4207 -> 384
-> 2855 -> 3732 -> 2398
      Maximum Bandwidth: 3798

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.01 seconds
      No. of Hops: 51
      Path: 486 -> 2937 -> 1095 -> 4440 -> 1626 -> 4835 -> 4968 -> 1953 ->
4572 -> 4502 -> 3263 -> 2758 -> 4662 -> 1997 -> 404 -> 3379 -> 1410 -> 2900 -
> 4661 -> 910 -> 2279 -> 4352 -> 3399 -> 1857 -> 3971 -> 3742 -> 1675 -> 2175
-> 3088 -> 2036 -> 4654 -> 2740 -> 924 -> 2194 -> 3558 -> 2354 -> 2528 ->
2407 -> 1919 -> 10 -> 4463 -> 48 -> 4014 -> 1709 -> 3443 -> 1151 -> 3866 ->
4207 -> 384 -> 2855 -> 3732 -> 2398
      Maximum Bandwidth: 3798

Finding Maximum bandwidth path between source = 104 and target = 1062.

      Found Maximum bandwidth path WITHOUT heap in 0.073 seconds
      No. of Hops: 42
      Path: 104 -> 4475 -> 2873 -> 141 -> 2253 -> 834 -> 792 -> 517 -> 1522 -
> 1261 -> 60 -> 456 -> 1778 -> 1374 -> 1502 -> 191 -> 264 -> 1260 -> 1085 ->
1461 -> 449 -> 168 -> 876 -> 801 -> 1087 -> 400 -> 479 -> 329 -> 858 -> 428 -
> 721 -> 980 -> 330 -> 1515 -> 1025 -> 1207 -> 1295 -> 618 -> 259 -> 1181 ->
1712 -> 285 -> 1062
      Maximum Bandwidth: 2215

      Found Maximum bandwidth path WITH heap in 0.008 seconds
      No. of Hops: 44
      Path: 104 -> 4475 -> 2873 -> 141 -> 3072 -> 3085 -> 3005 -> 4440 ->
1095 -> 2537 -> 393 -> 1549 -> 2027 -> 4876 -> 4325 -> 2134 -> 4689 -> 4769 -
> 4837 -> 1017 -> 3282 -> 1084 -> 1436 -> 3191 -> 1076 -> 4613 -> 743 -> 1809
-> 1326 -> 903 -> 4751 -> 1524 -> 3399 -> 3240 -> 1183 -> 1622 -> 4768 -> 939
-> 4376 -> 4643 -> 2567 -> 4913 -> 1210 -> 2364 -> 1062
      Maximum Bandwidth: 2215

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.018 seconds
      No. of Hops: 48
      Path: 104 -> 4475 -> 2873 -> 141 -> 3479 -> 4999 -> 3921 -> 3847 ->
2720 -> 2194 -> 924 -> 2740 -> 4654 -> 2036 -> 3088 -> 4766 -> 2038 -> 3196 -
> 1337 -> 3499 -> 2727 -> 2887 -> 2500 -> 4115 -> 1448 -> 76 -> 3589 -> 820 -
> 4013 -> 2263 -> 2779 -> 2180 -> 3112 -> 2248 -> 3381 -> 1394 -> 2087 ->
2913 -> 2340 -> 4997 -> 952 -> 3340 -> 504 -> 491 -> 354 -> 2404 -> 1458 ->
285 -> 1062
      Maximum Bandwidth: 2215

Finding Maximum bandwidth path between source = 3241 and target = 3787.

      Found Maximum bandwidth path WITHOUT heap in 0.148 seconds
      No. of Hops: 16
      Path: 3241 -> 591 -> 478 -> 3900 -> 4697 -> 4352 -> 3399 -> 1409 ->
1708 -> 85 -> 4243 -> 2588 -> 4372 -> 1935 -> 406 -> 533 -> 3787
      Maximum Bandwidth: 4087

      Found Maximum bandwidth path WITH heap in 0.006 seconds
      No. of Hops: 16
      Path: 3241 -> 591 -> 478 -> 3900 -> 4697 -> 4352 -> 3399 -> 1409 ->
1708 -> 85 -> 4243 -> 2588 -> 4372 -> 1935 -> 406 -> 533 -> 3787
      Maximum Bandwidth: 4087
      Found Maximum bandwidth path using Kruskal's Algorithm in 0.013 seconds
      No. of Hops: 46
      Path: 3241 -> 591 -> 478 -> 3900 -> 4697 -> 4352 -> 3399 -> 1857 ->
3971 -> 3742 -> 1675 -> 2175 -> 3088 -> 4766 -> 2038 -> 3196 -> 1337 -> 3499
-> 2727 -> 2887 -> 2500 -> 4115 -> 1448 -> 76 -> 3589 -> 820 -> 4013 -> 2263
-> 2779 -> 2180 -> 3112 -> 2248 -> 3997 -> 2391 -> 137 -> 332 -> 4873 -> 1119
-> 303 -> 3483 -> 2600 -> 308 -> 4372 -> 1935 -> 406 -> 533 -> 3787
      Maximum Bandwidth: 4087

Finding Maximum bandwidth path between source = 2760 and target = 3337.

      Found Maximum bandwidth path WITHOUT heap in 0.099 seconds
      No. of Hops: 60
      Path: 2760 -> 4997 -> 952 -> 3340 -> 504 -> 491 -> 354 -> 2404 -> 3792
-> 2464 -> 3541 -> 515 -> 1497 -> 3183 -> 2 -> 2991 -> 3093 -> 3666 -> 386 ->
1902 -> 2064 -> 3491 -> 1882 -> 2002 -> 1321 -> 1049 -> 1396 -> 3164 -> 275 -
> 3393 -> 257 -> 229 -> 1760 -> 1951 -> 2019 -> 746 -> 2334 -> 63 -> 85 ->
1708 -> 1409 -> 3399 -> 269 -> 882 -> 1269 -> 1495 -> 1846 -> 2946 -> 3221 ->
2354 -> 3558 -> 2194 -> 2720 -> 3157 -> 1689 -> 3551 -> 3338 -> 762 -> 3687 -
> 2427 -> 3337
      Maximum Bandwidth: 4495

      Found Maximum bandwidth path WITH heap in 0.008 seconds
      No. of Hops: 26
      Path: 2760 -> 4997 -> 2340 -> 2913 -> 2087 -> 1394 -> 3381 -> 2248 ->
3997 -> 2009 -> 3190 -> 3147 -> 85 -> 4243 -> 2588 -> 3671 -> 3798 -> 864 ->
812 -> 702 -> 1581 -> 4192 -> 770 -> 3912 -> 3687 -> 2427 -> 3337
      Maximum Bandwidth: 4495

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.009 seconds
      No. of Hops: 62
      Path: 2760 -> 4997 -> 2340 -> 2913 -> 2087 -> 1394 -> 3381 -> 2248 ->
3112 -> 2180 -> 2779 -> 2263 -> 4013 -> 820 -> 3589 -> 76 -> 1448 -> 4115 ->
2500 -> 2887 -> 2727 -> 3499 -> 1337 -> 3196 -> 2038 -> 4766 -> 3088 -> 2036
-> 4654 -> 2740 -> 924 -> 2194 -> 2720 -> 3847 -> 3921 -> 4999 -> 77 -> 2704
-> 1117 -> 1017 -> 440 -> 3676 -> 1718 -> 646 -> 4909 -> 4095 -> 1770 -> 2869
-> 1939 -> 3296 -> 488 -> 1927 -> 3743 -> 1674 -> 3223 -> 2226 -> 3516 ->
4192 -> 770 -> 3912 -> 3687 -> 2427 -> 3337
      Maximum Bandwidth: 4495

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.1044
Average time taken by Dijkstra with heap: 0.0174
Average time taken by Kruskal's Algorithm: 0.0234
=============================================================================
====


p1_graph2
Finding Maximum bandwidth path between source = 1998 and target = 989.

      Found Maximum bandwidth path WITHOUT heap in 0.156 seconds
      No. of Hops: 43
      Path: 1998 -> 3356 -> 4236 -> 3266 -> 3685 -> 1180 -> 3818 -> 1239 ->
137 -> 890 -> 3528 -> 4384 -> 2607 -> 4637 -> 3459 -> 3418 -> 329 -> 139 ->
4899 -> 998 -> 3856 -> 581 -> 83 -> 429 -> 652 -> 2876 -> 2530 -> 2300 ->
```

```
3743 -> 1281 -> 3340 -> 796 -> 1023 -> 2526 -> 3261 -> 3366 -> 268 -> 3918 ->
2610 -> 2457 -> 3049 -> 2940 -> 4333 -> 989
      Maximum Bandwidth: 5976

      Found Maximum bandwidth path WITH heap in 0.098 seconds
      No. of Hops: 69
      Path: 1998 -> 3356 -> 4236 -> 3266 -> 3685 -> 1180 -> 3818 -> 1239 ->
137 -> 890 -> 3528 -> 4384 -> 2607 -> 4637 -> 3459 -> 3418 -> 329 -> 139 ->
4899 -> 998 -> 3856 -> 581 -> 83 -> 443 -> 489 -> 4052 -> 2653 -> 3987 ->
2597 -> 4737 -> 923 -> 4133 -> 4949 -> 29 -> 4837 -> 765 -> 4720 -> 2005 ->
3838 -> 1357 -> 4191 -> 4718 -> 715 -> 3253 -> 918 -> 3179 -> 2284 -> 3311 ->
590 -> 4648 -> 4495 -> 2961 -> 4343 -> 1385 -> 2052 -> 742 -> 4319 -> 2314 ->
1376 -> 4250 -> 209 -> 1865 -> 268 -> 3918 -> 2610 -> 2457 -> 3049 -> 2940 ->
4333 -> 989
      Maximum Bandwidth: 5976

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.638 seconds
      No. of Hops: 46
      Path: 1998 -> 3356 -> 4236 -> 3266 -> 3685 -> 1180 -> 3818 -> 1239 ->
137 -> 890 -> 3528 -> 4384 -> 2607 -> 4637 -> 3459 -> 3418 -> 329 -> 139 ->
4899 -> 4265 -> 4575 -> 1219 -> 3465 -> 2759 -> 414 -> 3163 -> 3168 -> 4171 -
> 1204 -> 2214 -> 974 -> 1373 -> 1445 -> 2728 -> 4145 -> 4308 -> 1790 -> 2372
-> 3236 -> 961 -> 1420 -> 621 -> 2788 -> 4325 -> 2101 -> 4333 -> 989
      Maximum Bandwidth: 5976

Finding Maximum bandwidth path between source = 3201 and target = 2243.

      Found Maximum bandwidth path WITHOUT heap in 0.183 seconds
      No. of Hops: 44
      Path: 3201 -> 2918 -> 4036 -> 2082 -> 19 -> 3811 -> 2533 -> 1520 -> 191
-> 2434 -> 3251 -> 2620 -> 2942 -> 4254 -> 3600 -> 3381 -> 2320 -> 1729 ->
2916 -> 2644 -> 4418 -> 2372 -> 1790 -> 4308 -> 4145 -> 2728 -> 3318 -> 209 -
> 4250 -> 1376 -> 2314 -> 2995 -> 3748 -> 3327 -> 2605 -> 255 -> 453 -> 1328
-> 2221 -> 2709 -> 3580 -> 1809 -> 3449 -> 3194 -> 2243
      Maximum Bandwidth: 5994

      Found Maximum bandwidth path WITH heap in 0.107 seconds
      No. of Hops: 46
      Path: 3201 -> 2918 -> 4036 -> 2082 -> 19 -> 3811 -> 2533 -> 1520 -> 191
-> 2434 -> 1094 -> 1417 -> 2117 -> 4289 -> 1617 -> 3992 -> 2300 -> 2530 ->
2876 -> 652 -> 4580 -> 3848 -> 3897 -> 3209 -> 149 -> 1979 -> 2986 -> 4646 ->
2853 -> 1436 -> 160 -> 345 -> 4410 -> 3898 -> 3393 -> 749 -> 2395 -> 326 ->
4514 -> 4825 -> 2221 -> 2709 -> 3580 -> 1809 -> 3449 -> 3194 -> 2243
      Maximum Bandwidth: 5994

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.57 seconds
      No. of Hops: 93
      Path: 3201 -> 2918 -> 4036 -> 2082 -> 19 -> 3811 -> 2533 -> 1520 -> 191
-> 2434 -> 1094 -> 1417 -> 2117 -> 4289 -> 1617 -> 3992 -> 2300 -> 1495 ->
2921 -> 4160 -> 459 -> 661 -> 3372 -> 2366 -> 4737 -> 2470 -> 451 -> 1455 ->
4329 -> 2644 -> 4418 -> 2372 -> 1790 -> 4308 -> 4145 -> 2728 -> 1445 -> 1373
-> 974 -> 2214 -> 1204 -> 4171 -> 3168 -> 3163 -> 414 -> 2759 -> 3465 -> 1219
-> 1536 -> 399 -> 4902 -> 711 -> 3659 -> 1927 -> 4890 -> 3650 -> 2756 -> 1758
-> 1925 -> 4797 -> 2393 -> 2739 -> 1368 -> 3612 -> 4585 -> 938 -> 1156 -> 416
-> 4982 -> 1708 -> 2338 -> 1069 -> 3154 -> 468 -> 4343 -> 1385 -> 2052 -> 742
-> 4319 -> 2314 -> 2995 -> 3748 -> 3327 -> 2605 -> 255 -> 453 -> 1328 -> 2221
-> 2709 -> 3580 -> 1809 -> 3449 -> 3194 -> 2243
```

```
      Maximum Bandwidth: 5994

Finding Maximum bandwidth path between source = 2239 and target = 39.

      Found Maximum bandwidth path WITHOUT heap in 0.235 seconds
      No. of Hops: 34
      Path: 2239 -> 384 -> 2235 -> 2586 -> 1545 -> 971 -> 1279 -> 333 -> 1489
-> 4483 -> 3563 -> 383 -> 3686 -> 100 -> 1765 -> 3056 -> 2172 -> 1434 -> 495
-> 1943 -> 381 -> 319 -> 3477 -> 2671 -> 1105 -> 796 -> 1023 -> 2526 -> 3261
-> 3366 -> 268 -> 1865 -> 209 -> 3803 -> 39
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path WITH heap in 0.168 seconds
      No. of Hops: 43
      Path: 2239 -> 384 -> 2235 -> 2586 -> 1545 -> 971 -> 1279 -> 333 -> 1489
-> 4483 -> 3563 -> 383 -> 3686 -> 100 -> 1765 -> 4588 -> 181 -> 2686 -> 255 -
> 2605 -> 3454 -> 1883 -> 4208 -> 1570 -> 1377 -> 4843 -> 1023 -> 796 -> 3340
-> 1281 -> 3743 -> 2300 -> 1495 -> 2114 -> 4400 -> 1829 -> 3346 -> 610 ->
1530 -> 1990 -> 1781 -> 1072 -> 4586 -> 39
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.781 seconds
      No. of Hops: 38
      Path: 2239 -> 384 -> 2235 -> 2586 -> 1545 -> 971 -> 1279 -> 333 -> 1489
-> 4483 -> 3563 -> 383 -> 3686 -> 100 -> 1765 -> 4588 -> 1620 -> 460 -> 3926
-> 1885 -> 997 -> 3318 -> 2728 -> 4145 -> 4308 -> 1790 -> 2372 -> 4418 ->
2644 -> 4329 -> 1455 -> 451 -> 2470 -> 4737 -> 2597 -> 3987 -> 2653 -> 4052 -
> 39
      Maximum Bandwidth: 5992

Finding Maximum bandwidth path between source = 3416 and target = 473.

      Found Maximum bandwidth path WITHOUT heap in 0.205 seconds
      No. of Hops: 33
      Path: 3416 -> 4264 -> 502 -> 4631 -> 306 -> 3646 -> 985 -> 2612 -> 2000
-> 1043 -> 2255 -> 2120 -> 2319 -> 1333 -> 1758 -> 2756 -> 3650 -> 173 ->
1557 -> 3833 -> 4043 -> 3597 -> 446 -> 2130 -> 3350 -> 2618 -> 907 -> 4001 ->
2374 -> 2049 -> 4496 -> 2996 -> 934 -> 473
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path WITH heap in 0.135 seconds
      No. of Hops: 29
      Path: 3416 -> 4264 -> 502 -> 4631 -> 306 -> 3646 -> 985 -> 2612 -> 4946
-> 3659 -> 1927 -> 4890 -> 3650 -> 173 -> 1557 -> 3833 -> 4043 -> 3597 -> 446
-> 2130 -> 3350 -> 2618 -> 907 -> 4001 -> 2374 -> 2049 -> 4496 -> 2996 -> 934
-> 473
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.754 seconds
      No. of Hops: 100
      Path: 3416 -> 4264 -> 502 -> 4631 -> 306 -> 3646 -> 985 -> 2612 -> 4946
-> 3659 -> 1927 -> 4890 -> 3650 -> 2756 -> 1758 -> 1333 -> 2319 -> 2120 ->
374 -> 1973 -> 3434 -> 1549 -> 2932 -> 61 -> 3975 -> 3279 -> 1721 -> 1348 ->
2940 -> 4333 -> 2101 -> 4325 -> 2788 -> 621 -> 1420 -> 508 -> 3480 -> 3765 ->
3406 -> 4190 -> 4674 -> 3389 -> 977 -> 3633 -> 4023 -> 1121 -> 1534 -> 4007 -
> 546 -> 2283 -> 1862 -> 432 -> 2490 -> 1318 -> 373 -> 1702 -> 164 -> 2686 ->
255 -> 2605 -> 3327 -> 3748 -> 2995 -> 2314 -> 4319 -> 742 -> 2052 -> 1385 ->
```

```
4343 -> 2961 -> 4495 -> 929 -> 3995 -> 1767 -> 3398 -> 2959 -> 4987 -> 4507 -
> 3714 -> 3466 -> 1530 -> 4419 -> 1182 -> 4848 -> 3121 -> 3653 -> 3677 ->
3267 -> 3149 -> 221 -> 3523 -> 2251 -> 3161 -> 3229 -> 2912 -> 2730 -> 204 ->
4877 -> 2996 -> 934 -> 473
      Maximum Bandwidth: 5992

Finding Maximum bandwidth path between source = 1119 and target = 1030.

      Found Maximum bandwidth path WITHOUT heap in 0.173 seconds
      No. of Hops: 33
      Path: 1119 -> 3670 -> 3257 -> 2579 -> 4902 -> 399 -> 1536 -> 1219 ->
3465 -> 2759 -> 414 -> 3163 -> 3168 -> 4171 -> 1204 -> 2214 -> 974 -> 1373 ->
1445 -> 2728 -> 3318 -> 209 -> 4250 -> 1376 -> 2314 -> 2995 -> 3748 -> 3327 -
> 2605 -> 255 -> 1669 -> 732 -> 657 -> 1030
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path WITH heap in 0.098 seconds
      No. of Hops: 82
      Path: 1119 -> 3670 -> 3257 -> 1578 -> 3640 -> 4946 -> 3659 -> 1927 ->
4890 -> 3650 -> 2756 -> 1758 -> 1925 -> 4797 -> 2393 -> 2739 -> 1368 -> 3612
-> 274 -> 1210 -> 667 -> 3629 -> 1262 -> 772 -> 4841 -> 4726 -> 4250 -> 209 -
> 3318 -> 2728 -> 4145 -> 4308 -> 1790 -> 2372 -> 3236 -> 961 -> 1420 -> 621
-> 2788 -> 4325 -> 2101 -> 4333 -> 2940 -> 1348 -> 1721 -> 3279 -> 3975 -> 61
-> 2932 -> 1549 -> 3434 -> 3738 -> 2388 -> 2045 -> 2720 -> 4821 -> 765 ->
4720 -> 3147 -> 4941 -> 2307 -> 1599 -> 4761 -> 1936 -> 3115 -> 2508 -> 4474
-> 4644 -> 3165 -> 4524 -> 749 -> 2395 -> 326 -> 4514 -> 4825 -> 2221 -> 1328
-> 453 -> 255 -> 1669 -> 732 -> 657 -> 1030
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.702 seconds
      No. of Hops: 37
      Path: 1119 -> 3670 -> 3257 -> 1578 -> 3640 -> 4946 -> 3659 -> 711 ->
4902 -> 399 -> 1536 -> 1219 -> 3465 -> 2759 -> 414 -> 3163 -> 3168 -> 4171 ->
1204 -> 2214 -> 974 -> 1373 -> 1445 -> 2728 -> 3318 -> 209 -> 4250 -> 1376 ->
2314 -> 2995 -> 3748 -> 3327 -> 2605 -> 255 -> 1669 -> 732 -> 657 -> 1030
      Maximum Bandwidth: 5992

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.1904
Average time taken by Dijkstra with heap: 0.121199995
Average time taken by Kruskal's Algorithm: 2.6890001
==============================================================================
====


p2_graph1
Finding Maximum bandwidth path between source = 4047 and target = 802.

      Found Maximum bandwidth path WITHOUT heap in 0.096 seconds
      No. of Hops: 32
      Path: 4047 -> 1012 -> 3483 -> 3742 -> 1933 -> 1528 -> 3639 -> 3022 ->
1607 -> 474 -> 4397 -> 3214 -> 1935 -> 3289 -> 1058 -> 1751 -> 4150 -> 1335 -
> 2437 -> 2910 -> 1624 -> 3400 -> 1491 -> 2499 -> 1592 -> 1176 -> 442 -> 1800
-> 1910 -> 2847 -> 3638 -> 2775 -> 802
      Maximum Bandwidth: 4740

      Found Maximum bandwidth path WITH heap in 0.002 seconds
      No. of Hops: 32
      Path: 4047 -> 1012 -> 3483 -> 3742 -> 1933 -> 1528 -> 3639 -> 3022 ->
1607 -> 474 -> 4397 -> 3214 -> 1935 -> 3289 -> 1058 -> 1751 -> 4150 -> 1335 -
> 2437 -> 2910 -> 1624 -> 3400 -> 1491 -> 2499 -> 1592 -> 1176 -> 442 -> 1800
-> 1910 -> 2847 -> 3638 -> 2775 -> 802
      Maximum Bandwidth: 4740

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.005 seconds
      No. of Hops: 32
      Path: 4047 -> 1012 -> 3483 -> 3742 -> 1933 -> 1528 -> 3639 -> 3022 ->
1607 -> 474 -> 4397 -> 3214 -> 1935 -> 3289 -> 1058 -> 1751 -> 4150 -> 1335 -
> 2437 -> 2910 -> 1624 -> 3400 -> 1491 -> 2499 -> 1592 -> 1176 -> 442 -> 1800
-> 1910 -> 2847 -> 3638 -> 2775 -> 802
      Maximum Bandwidth: 4740

Finding Maximum bandwidth path between source = 616 and target = 1717.

      Found Maximum bandwidth path WITHOUT heap in 0.087 seconds
      No. of Hops: 26
      Path: 616 -> 4547 -> 3838 -> 3416 -> 2464 -> 2891 -> 1697 -> 1495 ->
609 -> 2699 -> 2334 -> 691 -> 1259 -> 1064 -> 1801 -> 2305 -> 2619 -> 277 ->
2396 -> 1470 -> 2568 -> 1165 -> 4444 -> 3560 -> 900 -> 4904 -> 1717
      Maximum Bandwidth: 1766

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 15
      Path: 616 -> 4547 -> 3838 -> 3545 -> 4655 -> 4165 -> 1669 -> 2604 ->
147 -> 4883 -> 199 -> 1392 -> 4713 -> 900 -> 4904 -> 1717
      Maximum Bandwidth: 1766

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.007 seconds
      No. of Hops: 67
      Path: 616 -> 4547 -> 3838 -> 3416 -> 2464 -> 2891 -> 2731 -> 3087 ->
1623 -> 4102 -> 2297 -> 164 -> 317 -> 351 -> 371 -> 4485 -> 4211 -> 23 ->
1373 -> 3365 -> 526 -> 3887 -> 4829 -> 1589 -> 2436 -> 2468 -> 3864 -> 1126 -
> 4250 -> 3237 -> 3535 -> 4539 -> 1627 -> 1069 -> 1480 -> 277 -> 3148 -> 1457
-> 3545 -> 4655 -> 4165 -> 3907 -> 3892 -> 839 -> 1052 -> 3205 -> 796 -> 3091
-> 1255 -> 859 -> 801 -> 756 -> 1379 -> 4844 -> 2351 -> 480 -> 2572 -> 4491 -
> 2140 -> 375 -> 3094 -> 3166 -> 4921 -> 2682 -> 4561 -> 900 -> 4904 -> 1717
      Maximum Bandwidth: 1766

Finding Maximum bandwidth path between source = 4861 and target = 112.

      Found Maximum bandwidth path WITHOUT heap in 0.113 seconds
      No. of Hops: 54
      Path: 4861 -> 1517 -> 3732 -> 2822 -> 4272 -> 1278 -> 4447 -> 346 ->
4982 -> 3779 -> 402 -> 351 -> 371 -> 4485 -> 4211 -> 23 -> 1373 -> 3365 ->
526 -> 686 -> 1013 -> 4417 -> 284 -> 1704 -> 671 -> 572 -> 2235 -> 1799 ->
1728 -> 865 -> 1845 -> 3719 -> 4019 -> 3495 -> 2027 -> 2369 -> 2538 -> 789 ->
1603 -> 662 -> 96 -> 1212 -> 2499 -> 1491 -> 3400 -> 1569 -> 2411 -> 1940 ->
4376 -> 1790 -> 1416 -> 3629 -> 2974 -> 3092 -> 112
      Maximum Bandwidth: 4671

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 22
      Path: 4861 -> 1517 -> 3732 -> 2822 -> 4272 -> 1278 -> 4447 -> 346 ->
4982 -> 3779 -> 1644 -> 4585 -> 1294 -> 3771 -> 2515 -> 1214 -> 3438 -> 799 -
> 823 -> 1161 -> 4797 -> 3092 -> 112
      Maximum Bandwidth: 4671

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 22
      Path: 4861 -> 1517 -> 3732 -> 2822 -> 4272 -> 1278 -> 4447 -> 346 ->
4982 -> 3779 -> 1644 -> 4585 -> 1294 -> 3771 -> 2515 -> 1214 -> 3438 -> 799 -
> 823 -> 1161 -> 4797 -> 3092 -> 112
      Maximum Bandwidth: 4671

Finding Maximum bandwidth path between source = 2078 and target = 886.

      Found Maximum bandwidth path WITHOUT heap in 0.093 seconds
      No. of Hops: 72
      Path: 2078 -> 1309 -> 3768 -> 2013 -> 4056 -> 2576 -> 3821 -> 2073 ->
986 -> 3269 -> 3676 -> 2012 -> 2389 -> 1555 -> 2115 -> 3640 -> 3658 -> 3452 -
> 334 -> 2235 -> 1799 -> 1728 -> 865 -> 1845 -> 3719 -> 4019 -> 3495 -> 2027
-> 2369 -> 2538 -> 1970 -> 3488 -> 354 -> 1887 -> 708 -> 2837 -> 3551 -> 350
-> 1889 -> 318 -> 716 -> 655 -> 2627 -> 3131 -> 2951 -> 639 -> 3363 -> 47 ->
2222 -> 387 -> 1345 -> 1957 -> 471 -> 2045 -> 1676 -> 2670 -> 2254 -> 3430 ->
2178 -> 609 -> 2699 -> 2334 -> 1989 -> 1525 -> 1468 -> 11 -> 3126 -> 765 ->
2818 -> 154 -> 1352 -> 3800 -> 886
      Maximum Bandwidth: 4514

      Found Maximum bandwidth path WITH heap in 0.004 seconds
      No. of Hops: 27
      Path: 2078 -> 1309 -> 3768 -> 2013 -> 4056 -> 2576 -> 3821 -> 2073 ->
986 -> 3269 -> 3676 -> 2012 -> 2389 -> 689 -> 2265 -> 2123 -> 2645 -> 3499 ->
4047 -> 1012 -> 3483 -> 4891 -> 3177 -> 2673 -> 154 -> 1352 -> 3800 -> 886
      Maximum Bandwidth: 4514

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.005 seconds
      No. of Hops: 37
      Path: 2078 -> 1309 -> 3768 -> 2013 -> 4056 -> 2576 -> 3821 -> 1227 ->
2662 -> 1224 -> 1802 -> 2387 -> 1516 -> 958 -> 4160 -> 3525 -> 2429 -> 4321 -
> 3907 -> 4165 -> 4655 -> 3545 -> 1457 -> 3148 -> 277 -> 1480 -> 1069 -> 1627
-> 4539 -> 3535 -> 3237 -> 4250 -> 1126 -> 2818 -> 154 -> 1352 -> 3800 -> 886
      Maximum Bandwidth: 4514

Finding Maximum bandwidth path between source = 1776 and target = 3161.

      Found Maximum bandwidth path WITHOUT heap in 0.1 seconds
      No. of Hops: 39
      Path: 1776 -> 851 -> 1511 -> 3450 -> 889 -> 2453 -> 1464 -> 365 -> 2610
-> 4119 -> 4113 -> 1064 -> 1259 -> 691 -> 2750 -> 2219 -> 2044 -> 676 -> 33 -
> 1182 -> 3763 -> 1539 -> 2373 -> 3546 -> 1850 -> 772 -> 4 -> 423 -> 1069 ->
1480 -> 277 -> 3148 -> 1195 -> 4127 -> 1993 -> 1354 -> 2935 -> 1508 -> 4656 -
> 3161
      Maximum Bandwidth: 3195

      Found Maximum bandwidth path WITH heap in 0.004 seconds
      No. of Hops: 25
      Path: 1776 -> 851 -> 1511 -> 3450 -> 889 -> 2453 -> 1464 -> 365 -> 2610
-> 4119 -> 4113 -> 1064 -> 1259 -> 691 -> 2750 -> 4461 -> 1457 -> 3148 ->
1195 -> 4127 -> 1993 -> 1354 -> 2935 -> 1508 -> 4656 -> 3161
```

```
      Maximum Bandwidth: 3195

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 20
      Path: 1776 -> 851 -> 1511 -> 3450 -> 889 -> 2453 -> 1464 -> 365 -> 2610
-> 4119 -> 4113 -> 1440 -> 2163 -> 4626 -> 4127 -> 1993 -> 1354 -> 2935 ->
1508 -> 4656 -> 3161
      Maximum Bandwidth: 3195

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.0978
Average time taken by Dijkstra with heap: 0.0032000002
Average time taken by Kruskal's Algorithm: 0.0050000004
============================================================================
====


p2_graph2
Finding Maximum bandwidth path between source = 2145 and target = 3046.

      Found Maximum bandwidth path WITHOUT heap in 0.149 seconds
      No. of Hops: 41
      Path: 2145 -> 3285 -> 2476 -> 3395 -> 2898 -> 3783 -> 4231 -> 2158 ->
2672 -> 816 -> 220 -> 2251 -> 2359 -> 2204 -> 3780 -> 4344 -> 4023 -> 3111 ->
1598 -> 2313 -> 2400 -> 415 -> 2995 -> 1520 -> 2116 -> 2514 -> 2489 -> 1144 -
> 2948 -> 489 -> 4679 -> 363 -> 3311 -> 2945 -> 914 -> 436 -> 4752 -> 4442 ->
2899 -> 2368 -> 3141 -> 3046
      Maximum Bandwidth: 5994

      Found Maximum bandwidth path WITH heap in 0.07 seconds
      No. of Hops: 41
      Path: 2145 -> 3285 -> 2476 -> 3395 -> 2898 -> 3783 -> 4231 -> 2158 ->
2672 -> 816 -> 220 -> 2251 -> 2359 -> 2204 -> 3780 -> 4344 -> 4023 -> 4797 ->
2102 -> 4773 -> 3507 -> 1010 -> 2995 -> 1520 -> 2116 -> 2514 -> 2489 -> 1144
-> 2948 -> 489 -> 4679 -> 363 -> 3311 -> 2945 -> 914 -> 436 -> 4752 -> 4442 -
> 2899 -> 2368 -> 3141 -> 3046
      Maximum Bandwidth: 5994

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.57 seconds
      No. of Hops: 41
      Path: 2145 -> 3285 -> 2476 -> 3395 -> 2898 -> 3783 -> 4231 -> 2158 ->
2672 -> 816 -> 220 -> 2251 -> 2359 -> 2204 -> 3780 -> 4344 -> 4023 -> 4797 ->
2102 -> 4773 -> 3507 -> 1010 -> 2995 -> 1520 -> 2116 -> 2514 -> 2489 -> 1144
-> 2948 -> 489 -> 4679 -> 363 -> 3311 -> 2945 -> 914 -> 436 -> 4752 -> 4442 -
> 2899 -> 2368 -> 3141 -> 3046
      Maximum Bandwidth: 5994

Finding Maximum bandwidth path between source = 2245 and target = 2394.

      Found Maximum bandwidth path WITHOUT heap in 0.213 seconds
      No. of Hops: 18
      Path: 2245 -> 3008 -> 3373 -> 3094 -> 1123 -> 1597 -> 3200 -> 2142 ->
3386 -> 2178 -> 2747 -> 2078 -> 3454 -> 2402 -> 141 -> 3509 -> 1672 -> 552 ->
2394
      Maximum Bandwidth: 5991

      Found Maximum bandwidth path WITH heap in 0.145 seconds
      No. of Hops: 30
      Path: 2245 -> 3008 -> 3373 -> 3094 -> 1123 -> 1597 -> 4448 -> 2325 ->
2013 -> 3566 -> 3613 -> 2051 -> 1838 -> 91 -> 1658 -> 455 -> 3280 -> 2335 ->
2228 -> 3597 -> 2898 -> 3783 -> 2365 -> 717 -> 2082 -> 4634 -> 839 -> 3932 ->
1139 -> 552 -> 2394
      Maximum Bandwidth: 5991

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.558 seconds
      No. of Hops: 44
      Path: 2245 -> 3008 -> 3373 -> 3094 -> 1123 -> 1597 -> 4448 -> 2325 ->
2013 -> 3566 -> 2900 -> 434 -> 641 -> 572 -> 182 -> 625 -> 1364 -> 3735 ->
751 -> 2126 -> 2825 -> 2450 -> 3926 -> 1433 -> 3601 -> 3590 -> 453 -> 2938 ->
750 -> 2638 -> 694 -> 3025 -> 4642 -> 1955 -> 332 -> 3172 -> 431 -> 2496 ->
4101 -> 4018 -> 1526 -> 2487 -> 2817 -> 552 -> 2394
      Maximum Bandwidth: 5991

Finding Maximum bandwidth path between source = 3617 and target = 1484.

      Found Maximum bandwidth path WITHOUT heap in 0.153 seconds
      No. of Hops: 30
      Path: 3617 -> 793 -> 2356 -> 2324 -> 3355 -> 2645 -> 540 -> 2354 ->
2014 -> 2317 -> 340 -> 486 -> 2319 -> 554 -> 2441 -> 494 -> 1783 -> 366 ->
1850 -> 962 -> 1091 -> 369 -> 1954 -> 1991 -> 1956 -> 1587 -> 2127 -> 3989 ->
4393 -> 642 -> 1484
      Maximum Bandwidth: 5981

      Found Maximum bandwidth path WITH heap in 0.111 seconds
      No. of Hops: 84
      Path: 3617 -> 793 -> 1160 -> 4181 -> 1243 -> 2975 -> 3223 -> 1526 ->
2193 -> 1193 -> 4840 -> 4511 -> 4270 -> 4338 -> 3246 -> 160 -> 1842 -> 1575 -
> 2478 -> 2397 -> 1486 -> 2846 -> 3303 -> 155 -> 2921 -> 2062 -> 4035 -> 2542
-> 2162 -> 1729 -> 4543 -> 2009 -> 2774 -> 2723 -> 4801 -> 3435 -> 2896 ->
1008 -> 4459 -> 4558 -> 992 -> 1021 -> 1148 -> 3346 -> 4004 -> 503 -> 4770 ->
3337 -> 3367 -> 3342 -> 545 -> 4981 -> 1612 -> 4030 -> 172 -> 2335 -> 2228 ->
9 -> 2068 -> 1544 -> 2618 -> 1791 -> 2843 -> 1607 -> 1370 -> 769 -> 447 ->
1426 -> 1216 -> 3792 -> 3572 -> 1200 -> 3238 -> 369 -> 2615 -> 2028 -> 1777 -
> 1361 -> 2013 -> 4506 -> 4869 -> 3989 -> 4393 -> 642 -> 1484
      Maximum Bandwidth: 5981

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.629 seconds
      No. of Hops: 15
      Path: 3617 -> 793 -> 815 -> 1553 -> 3783 -> 4231 -> 2158 -> 2672 -> 816
-> 3961 -> 1943 -> 2127 -> 3989 -> 4393 -> 642 -> 1484
      Maximum Bandwidth: 5981

Finding Maximum bandwidth path between source = 1338 and target = 1159.

      Found Maximum bandwidth path WITHOUT heap in 0.159 seconds
      No. of Hops: 37
      Path: 1338 -> 994 -> 336 -> 2290 -> 3707 -> 1140 -> 305 -> 418 -> 3651
-> 202 -> 637 -> 2716 -> 2074 -> 1293 -> 2350 -> 339 -> 541 -> 3252 -> 555 ->
2430 -> 322 -> 2962 -> 2741 -> 2906 -> 447 -> 2549 -> 823 -> 2191 -> 1818 ->
2359 -> 3699 -> 3673 -> 304 -> 2490 -> 2830 -> 3445 -> 707 -> 1159
      Maximum Bandwidth: 5991

      Found Maximum bandwidth path WITH heap in 0.072 seconds
      No. of Hops: 23
      Path: 1338 -> 994 -> 336 -> 2290 -> 3707 -> 1140 -> 4790 -> 2893 ->
3863 -> 115 -> 3764 -> 1714 -> 1602 -> 3708 -> 4634 -> 2904 -> 4266 -> 1661 -
> 4486 -> 4851 -> 2125 -> 2730 -> 2948 -> 1159
      Maximum Bandwidth: 5991

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.549 seconds
      No. of Hops: 36
      Path: 1338 -> 994 -> 336 -> 1486 -> 2846 -> 1175 -> 71 -> 4575 -> 247 -
> 4759 -> 967 -> 4356 -> 2064 -> 748 -> 1440 -> 502 -> 3847 -> 1960 -> 879 ->
584 -> 1709 -> 1781 -> 763 -> 2761 -> 2258 -> 3582 -> 4773 -> 3507 -> 1010 ->
2995 -> 1520 -> 2116 -> 2514 -> 2489 -> 1144 -> 2948 -> 1159
      Maximum Bandwidth: 5991

Finding Maximum bandwidth path between source = 2946 and target = 2414.

      Found Maximum bandwidth path WITHOUT heap in 0.2 seconds
      No. of Hops: 49
      Path: 2946 -> 91 -> 1658 -> 553 -> 2884 -> 173 -> 605 -> 685 -> 600 ->
121 -> 2503 -> 372 -> 1554 -> 196 -> 2032 -> 999 -> 1296 -> 2234 -> 505 ->
2439 -> 1257 -> 434 -> 641 -> 572 -> 182 -> 1173 -> 2308 -> 1294 -> 1270 ->
1898 -> 1178 -> 298 -> 661 -> 761 -> 1648 -> 3237 -> 2908 -> 561 -> 1910 ->
2152 -> 1060 -> 3767 -> 2064 -> 2909 -> 186 -> 3369 -> 3317 -> 991 -> 4619 ->
2414
      Maximum Bandwidth: 5990

      Found Maximum bandwidth path WITH heap in 0.143 seconds
      No. of Hops: 54
      Path: 2946 -> 91 -> 4078 -> 4477 -> 412 -> 2590 -> 1497 -> 341 -> 4180
-> 3770 -> 2462 -> 589 -> 3351 -> 3818 -> 3797 -> 3524 -> 900 -> 569 -> 146 -
> 3406 -> 1653 -> 2123 -> 2379 -> 4766 -> 3981 -> 2610 -> 729 -> 4011 -> 2953
-> 2536 -> 420 -> 1419 -> 1787 -> 2013 -> 2325 -> 2392 -> 1957 -> 3699 ->
2359 -> 2251 -> 220 -> 816 -> 4756 -> 4364 -> 565 -> 4704 -> 3651 -> 202 ->
2472 -> 2151 -> 4130 -> 926 -> 4174 -> 4995 -> 2414
      Maximum Bandwidth: 5990

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.875 seconds
      No. of Hops: 74
      Path: 2946 -> 91 -> 1658 -> 1792 -> 2614 -> 1956 -> 3875 -> 865 -> 1166
-> 3887 -> 4433 -> 1620 -> 1905 -> 3442 -> 146 -> 3406 -> 913 -> 1464 -> 1114
-> 431 -> 3172 -> 826 -> 2378 -> 926 -> 3693 -> 3879 -> 2552 -> 65 -> 4735 ->
3603 -> 443 -> 1820 -> 2945 -> 3311 -> 363 -> 4679 -> 489 -> 2948 -> 1144 ->
2489 -> 2514 -> 2116 -> 1520 -> 2995 -> 415 -> 2400 -> 2313 -> 1598 -> 3111 -
> 4023 -> 4797 -> 2102 -> 4773 -> 1710 -> 3744 -> 3665 -> 4850 -> 1211 ->
1655 -> 683 -> 4660 -> 4432 -> 3596 -> 3358 -> 4992 -> 594 -> 1063 -> 4935 ->
1268 -> 186 -> 3369 -> 3317 -> 991 -> 4619 -> 2414
      Maximum Bandwidth: 5990

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.17480001
Average time taken by Dijkstra with heap: 0.1082
Average time taken by Kruskal's Algorithm: 2.6362
============================================================================
====


p3_graph1
Finding Maximum bandwidth path between source = 4996 and target = 33.
```

```
      Found Maximum bandwidth path WITHOUT heap in 0.108 seconds
      No. of Hops: 34
      Path: 4996 -> 711 -> 4752 -> 4426 -> 1014 -> 192 -> 2698 -> 880 -> 495
-> 2439 -> 1531 -> 1721 -> 4407 -> 4833 -> 3721 -> 169 -> 811 -> 974 -> 3452
-> 4959 -> 2065 -> 4452 -> 3195 -> 4510 -> 1996 -> 3470 -> 3877 -> 2427 ->
2254 -> 1476 -> 4248 -> 2638 -> 3870 -> 1400 -> 33
      Maximum Bandwidth: 4605

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 34
      Path: 4996 -> 711 -> 4752 -> 4426 -> 1014 -> 192 -> 2698 -> 880 -> 495
-> 2439 -> 1531 -> 1721 -> 4407 -> 4833 -> 3721 -> 169 -> 811 -> 974 -> 3452
-> 4959 -> 2065 -> 4452 -> 3195 -> 4510 -> 1996 -> 3470 -> 3877 -> 2427 ->
2254 -> 1476 -> 4248 -> 2638 -> 3870 -> 1400 -> 33
      Maximum Bandwidth: 4605

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.006 seconds
      No. of Hops: 34
      Path: 4996 -> 711 -> 4752 -> 4426 -> 1014 -> 192 -> 2698 -> 880 -> 495
-> 2439 -> 1531 -> 1721 -> 4407 -> 4833 -> 3721 -> 169 -> 811 -> 974 -> 3452
-> 4959 -> 2065 -> 4452 -> 3195 -> 4510 -> 1996 -> 3470 -> 3877 -> 2427 ->
2254 -> 1476 -> 4248 -> 2638 -> 3870 -> 1400 -> 33
      Maximum Bandwidth: 4605

Finding Maximum bandwidth path between source = 1734 and target = 1599.

      Found Maximum bandwidth path WITHOUT heap in 0.096 seconds
      No. of Hops: 86
      Path: 1734 -> 1054 -> 392 -> 2831 -> 4924 -> 4702 -> 509 -> 826 -> 3590
-> 1243 -> 1868 -> 2950 -> 2999 -> 2507 -> 553 -> 4334 -> 1118 -> 3391 ->
1677 -> 1443 -> 1634 -> 4263 -> 323 -> 297 -> 2300 -> 171 -> 1767 -> 3577 ->
149 -> 563 -> 2949 -> 2420 -> 2509 -> 4126 -> 4266 -> 3858 -> 1334 -> 570 ->
3156 -> 3456 -> 4128 -> 854 -> 306 -> 1408 -> 2622 -> 2030 -> 646 -> 1336 ->
4145 -> 1803 -> 1668 -> 1657 -> 3395 -> 612 -> 938 -> 3890 -> 4088 -> 4510 ->
1996 -> 3470 -> 3877 -> 2427 -> 2254 -> 1476 -> 4248 -> 2638 -> 3870 -> 1648
-> 3729 -> 2927 -> 2247 -> 4199 -> 1346 -> 850 -> 2368 -> 4586 -> 2259 -> 207
-> 1531 -> 2921 -> 3459 -> 2640 -> 2674 -> 3524 -> 4231 -> 1654 -> 1599
      Maximum Bandwidth: 4743

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 36
      Path: 1734 -> 1054 -> 392 -> 2831 -> 4924 -> 4702 -> 509 -> 826 -> 158
-> 4897 -> 4239 -> 3230 -> 3475 -> 2035 -> 1168 -> 383 -> 3353 -> 2682 ->
4180 -> 1992 -> 249 -> 1281 -> 3780 -> 2941 -> 3137 -> 880 -> 495 -> 2439 ->
1531 -> 2921 -> 3459 -> 2640 -> 2674 -> 3524 -> 4231 -> 1654 -> 1599
      Maximum Bandwidth: 4743

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 36
      Path: 1734 -> 1054 -> 392 -> 2831 -> 4924 -> 4702 -> 509 -> 826 -> 3590
-> 1243 -> 1868 -> 2950 -> 2999 -> 2507 -> 553 -> 4334 -> 1118 -> 3391 ->
2226 -> 1180 -> 4496 -> 3938 -> 1132 -> 157 -> 4615 -> 4961 -> 2897 -> 3611 -
> 1531 -> 2921 -> 3459 -> 2640 -> 2674 -> 3524 -> 4231 -> 1654 -> 1599
      Maximum Bandwidth: 4743

Finding Maximum bandwidth path between source = 410 and target = 4700.
```

```
      Found Maximum bandwidth path WITHOUT heap in 0.067 seconds
      No. of Hops: 38
      Path: 410 -> 676 -> 1586 -> 2310 -> 2500 -> 1728 -> 2465 -> 1501 ->
2225 -> 2741 -> 2232 -> 1493 -> 2631 -> 1822 -> 2051 -> 2541 -> 2306 -> 1991
-> 1269 -> 2240 -> 1192 -> 747 -> 907 -> 2300 -> 674 -> 594 -> 610 -> 1984 ->
85 -> 2281 -> 451 -> 1067 -> 2215 -> 792 -> 1780 -> 356 -> 306 -> 641 -> 4700
      Maximum Bandwidth: 3498

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 39
      Path: 410 -> 676 -> 1586 -> 2310 -> 2500 -> 1728 -> 4645 -> 1313 ->
3149 -> 4821 -> 246 -> 4058 -> 3882 -> 1168 -> 2508 -> 387 -> 3671 -> 4273 ->
874 -> 4838 -> 2561 -> 1260 -> 3283 -> 3002 -> 2987 -> 1827 -> 4839 -> 2928 -
> 4491 -> 328 -> 4381 -> 3928 -> 722 -> 4050 -> 4503 -> 4978 -> 4601 -> 2719
-> 641 -> 4700
      Maximum Bandwidth: 3498

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 42
      Path: 410 -> 676 -> 1586 -> 2310 -> 2500 -> 2317 -> 4025 -> 4945 ->
1542 -> 3812 -> 4506 -> 640 -> 2510 -> 1849 -> 980 -> 2983 -> 2569 -> 2476 ->
2978 -> 3234 -> 320 -> 30 -> 3525 -> 4728 -> 2436 -> 1034 -> 1191 -> 4226 ->
336 -> 1014 -> 192 -> 2698 -> 880 -> 495 -> 2439 -> 1531 -> 1721 -> 4407 ->
4833 -> 3721 -> 169 -> 641 -> 4700
      Maximum Bandwidth: 3498

Finding Maximum bandwidth path between source = 4769 and target = 4265.

      Found Maximum bandwidth path WITHOUT heap in 0.093 seconds
      No. of Hops: 27
      Path: 4769 -> 4398 -> 4190 -> 1045 -> 3018 -> 2038 -> 502 -> 1364 ->
2074 -> 864 -> 1306 -> 169 -> 641 -> 306 -> 356 -> 1780 -> 1590 -> 141 ->
1828 -> 4088 -> 3890 -> 938 -> 612 -> 3395 -> 1657 -> 1668 -> 1803 -> 4265
      Maximum Bandwidth: 4449

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 82
      Path: 4769 -> 4398 -> 4190 -> 1045 -> 3018 -> 2038 -> 4964 -> 417 ->
427 -> 3017 -> 2302 -> 2770 -> 3298 -> 2546 -> 3262 -> 3338 -> 4681 -> 4765 -
> 2591 -> 4121 -> 2632 -> 4665 -> 2530 -> 261 -> 3148 -> 644 -> 3360 -> 3326
-> 2799 -> 3962 -> 2345 -> 1728 -> 2500 -> 2317 -> 4025 -> 4945 -> 1287 ->
2193 -> 94 -> 2804 -> 904 -> 1255 -> 4640 -> 1401 -> 3570 -> 4838 -> 874 ->
4273 -> 91 -> 3841 -> 2453 -> 166 -> 505 -> 1664 -> 4359 -> 2940 -> 4080 ->
4456 -> 3785 -> 3170 -> 465 -> 2099 -> 24 -> 2031 -> 4922 -> 4889 -> 2490 ->
3158 -> 1441 -> 2681 -> 987 -> 2536 -> 536 -> 1064 -> 4806 -> 4038 -> 717 ->
4306 -> 4990 -> 4515 -> 4283 -> 1803 -> 4265
      Maximum Bandwidth: 4449

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 68
      Path: 4769 -> 4398 -> 4190 -> 1045 -> 3018 -> 2038 -> 4964 -> 417 ->
427 -> 3017 -> 2667 -> 3919 -> 1449 -> 2101 -> 2278 -> 3399 -> 4786 -> 3882 -
> 1168 -> 2035 -> 3475 -> 3230 -> 4239 -> 4897 -> 158 -> 826 -> 3590 -> 1243
-> 1868 -> 2950 -> 2999 -> 2507 -> 553 -> 4334 -> 1118 -> 3391 -> 2226 ->
1180 -> 4496 -> 3938 -> 1132 -> 157 -> 4615 -> 4961 -> 2897 -> 3611 -> 1531 -
> 1721 -> 4407 -> 4833 -> 3721 -> 169 -> 811 -> 974 -> 3452 -> 4959 -> 2065 -
> 4452 -> 3195 -> 4510 -> 4088 -> 3890 -> 938 -> 612 -> 3395 -> 1657 -> 1668
-> 1803 -> 4265
      Maximum Bandwidth: 4449

Finding Maximum bandwidth path between source = 3626 and target = 3780.

      Found Maximum bandwidth path WITHOUT heap in 0.076 seconds
      No. of Hops: 37
      Path: 3626 -> 3434 -> 3324 -> 984 -> 2550 -> 59 -> 1815 -> 2776 -> 803
-> 2681 -> 987 -> 2536 -> 1742 -> 2325 -> 1517 -> 678 -> 1343 -> 581 -> 1836
-> 352 -> 901 -> 1879 -> 1754 -> 150 -> 2856 -> 642 -> 2875 -> 194 -> 510 ->
188 -> 101 -> 489 -> 993 -> 39 -> 2401 -> 935 -> 2941 -> 3780
      Maximum Bandwidth: 4023

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 56
      Path: 3626 -> 3434 -> 3324 -> 984 -> 2952 -> 4696 -> 1708 -> 3702 ->
4746 -> 2240 -> 1192 -> 747 -> 4675 -> 4220 -> 509 -> 826 -> 3590 -> 1243 ->
3977 -> 4683 -> 4793 -> 4473 -> 3589 -> 2481 -> 294 -> 2600 -> 4640 -> 1401 -
> 1799 -> 3053 -> 4152 -> 3466 -> 1587 -> 4011 -> 4860 -> 2687 -> 1088 -> 581
-> 1836 -> 352 -> 4494 -> 4480 -> 1509 -> 138 -> 1055 -> 4568 -> 923 -> 4832
-> 482 -> 4400 -> 2007 -> 3228 -> 3761 -> 1992 -> 249 -> 1281 -> 3780
      Maximum Bandwidth: 4023

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 90
      Path: 3626 -> 3434 -> 3324 -> 984 -> 2456 -> 1018 -> 423 -> 4451 ->
3830 -> 549 -> 3680 -> 381 -> 3931 -> 1142 -> 2231 -> 2390 -> 2628 -> 1748 ->
3935 -> 1962 -> 1970 -> 1126 -> 4324 -> 2581 -> 3936 -> 4025 -> 4945 -> 1542
-> 3812 -> 4506 -> 640 -> 2510 -> 1849 -> 980 -> 2983 -> 2569 -> 2476 -> 2978
-> 3234 -> 320 -> 30 -> 3525 -> 4728 -> 2436 -> 1034 -> 1191 -> 4226 -> 336 -
> 1014 -> 192 -> 2698 -> 880 -> 495 -> 2439 -> 1531 -> 3611 -> 2897 -> 4961 -
> 4615 -> 157 -> 1132 -> 3938 -> 4496 -> 1180 -> 2226 -> 3391 -> 1118 -> 4334
-> 553 -> 2507 -> 2999 -> 2950 -> 1868 -> 1243 -> 3590 -> 826 -> 158 -> 4897
-> 4239 -> 3230 -> 3475 -> 2035 -> 1168 -> 383 -> 3353 -> 2682 -> 4180 ->
1992 -> 249 -> 1281 -> 3780
      Maximum Bandwidth: 4023

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.088
Average time taken by Dijkstra with heap: 0.003
Average time taken by Kruskal's Algorithm: 0.0044
==============================================================================
====


p3_graph2
Finding Maximum bandwidth path between source = 1720 and target = 171.

      Found Maximum bandwidth path WITHOUT heap in 0.126 seconds
      No. of Hops: 40
      Path: 1720 -> 3314 -> 2687 -> 1661 -> 1817 -> 1765 -> 94 -> 11 -> 1662
-> 1367 -> 1509 -> 1437 -> 1630 -> 977 -> 1370 -> 1731 -> 370 -> 363 -> 1816
-> 1319 -> 468 -> 1312 -> 1425 -> 1010 -> 1153 -> 1771 -> 1862 -> 12 -> 847 -
> 408 -> 982 -> 1707 -> 488 -> 1582 -> 1123 -> 1460 -> 1068 -> 307 -> 641 ->
422 -> 171
      Maximum Bandwidth: 5983
      Found Maximum bandwidth path WITH heap in 0.085 seconds
      No. of Hops: 54
      Path: 1720 -> 3314 -> 2687 -> 4563 -> 958 -> 1940 -> 2072 -> 775 -> 438
-> 4411 -> 126 -> 1185 -> 1161 -> 1773 -> 3152 -> 2961 -> 2384 -> 82 -> 609 -
> 3917 -> 1184 -> 158 -> 4843 -> 4009 -> 4583 -> 1037 -> 1418 -> 4216 -> 1905
-> 4233 -> 1398 -> 926 -> 784 -> 458 -> 167 -> 2649 -> 1330 -> 2210 -> 3672 -
> 4313 -> 730 -> 235 -> 3566 -> 3079 -> 3536 -> 3716 -> 1288 -> 1200 -> 445 -
> 3350 -> 3761 -> 307 -> 641 -> 422 -> 171
      Maximum Bandwidth: 5983

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.546 seconds
      No. of Hops: 49
      Path: 1720 -> 3314 -> 2687 -> 1661 -> 4074 -> 4869 -> 893 -> 850 -> 285
-> 2543 -> 3818 -> 4444 -> 900 -> 2418 -> 3097 -> 2323 -> 3073 -> 2584 ->
1123 -> 4641 -> 1785 -> 983 -> 3343 -> 4692 -> 1985 -> 3673 -> 956 -> 2650 ->
1141 -> 2823 -> 3876 -> 2400 -> 2512 -> 1071 -> 1526 -> 4503 -> 1433 -> 2126
-> 293 -> 778 -> 157 -> 3677 -> 882 -> 2103 -> 2189 -> 1662 -> 1367 -> 4026 -
> 3961 -> 171
      Maximum Bandwidth: 5983

Finding Maximum bandwidth path between source = 2025 and target = 708.

      Found Maximum bandwidth path WITHOUT heap in 0.161 seconds
      No. of Hops: 11
      Path: 2025 -> 3822 -> 4692 -> 3343 -> 983 -> 1785 -> 4641 -> 1123 ->
2584 -> 554 -> 1482 -> 708
      Maximum Bandwidth: 5988

      Found Maximum bandwidth path WITH heap in 0.085 seconds
      No. of Hops: 11
      Path: 2025 -> 3822 -> 4692 -> 3343 -> 983 -> 1785 -> 4641 -> 1123 ->
2584 -> 554 -> 1482 -> 708
      Maximum Bandwidth: 5988

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.542 seconds
      No. of Hops: 34
      Path: 2025 -> 3822 -> 4692 -> 3343 -> 983 -> 1785 -> 4641 -> 1123 ->
2584 -> 3073 -> 2323 -> 3097 -> 2418 -> 900 -> 4444 -> 3818 -> 2543 -> 285 ->
850 -> 374 -> 1977 -> 3372 -> 1453 -> 547 -> 3854 -> 3792 -> 2375 -> 715 ->
2157 -> 4473 -> 4854 -> 477 -> 1316 -> 1482 -> 708
      Maximum Bandwidth: 5988

Finding Maximum bandwidth path between source = 1579 and target = 439.

      Found Maximum bandwidth path WITHOUT heap in 0.216 seconds
      No. of Hops: 30
      Path: 1579 -> 3403 -> 3489 -> 3082 -> 27 -> 1589 -> 1713 -> 2095 ->
1095 -> 2819 -> 195 -> 644 -> 1075 -> 3411 -> 2006 -> 1710 -> 828 -> 801 ->
3567 -> 3264 -> 3161 -> 1454 -> 2198 -> 2230 -> 4499 -> 2965 -> 4183 -> 1044
-> 266 -> 2499 -> 439
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path WITH heap in 0.148 seconds
      No. of Hops: 38
      Path: 1579 -> 3403 -> 3038 -> 1717 -> 4173 -> 3039 -> 1281 -> 1769 ->
1043 -> 2478 -> 2078 -> 2276 -> 1981 -> 3228 -> 1371 -> 4096 -> 4800 -> 801 -
```

```
> 3567 -> 4438 -> 3076 -> 1458 -> 2375 -> 3792 -> 459 -> 3261 -> 2647 -> 4411
-> 126 -> 4409 -> 1209 -> 3744 -> 4499 -> 2965 -> 4183 -> 1044 -> 266 -> 2499
-> 439
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.542 seconds
      No. of Hops: 23
      Path: 1579 -> 3403 -> 3489 -> 3082 -> 27 -> 1589 -> 1713 -> 2095 ->
1095 -> 2819 -> 4273 -> 3567 -> 3264 -> 3161 -> 1454 -> 2198 -> 2230 -> 4499
-> 2965 -> 4183 -> 1044 -> 266 -> 2499 -> 439
      Maximum Bandwidth: 5992

Finding Maximum bandwidth path between source = 174 and target = 994.

      Found Maximum bandwidth path WITHOUT heap in 0.184 seconds
      No. of Hops: 49
      Path: 174 -> 2696 -> 3988 -> 720 -> 1359 -> 3422 -> 2575 -> 1151 -> 368
-> 1141 -> 2650 -> 956 -> 3673 -> 1985 -> 4692 -> 3343 -> 983 -> 1785 -> 4641
-> 1123 -> 2584 -> 3073 -> 2323 -> 3097 -> 2418 -> 900 -> 4444 -> 3818 ->
2543 -> 285 -> 850 -> 374 -> 1977 -> 3372 -> 1453 -> 547 -> 3854 -> 3792 ->
2375 -> 1458 -> 3076 -> 4438 -> 3567 -> 4273 -> 2819 -> 1095 -> 2095 -> 4424
-> 2529 -> 994
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path WITH heap in 0.118 seconds
      No. of Hops: 52
      Path: 174 -> 2696 -> 3988 -> 720 -> 1359 -> 3422 -> 2575 -> 1151 -> 368
-> 1141 -> 2823 -> 3876 -> 2400 -> 2512 -> 1071 -> 1526 -> 4503 -> 1433 ->
2126 -> 4578 -> 3660 -> 1067 -> 598 -> 2076 -> 4731 -> 1732 -> 3619 -> 1714 -
> 1373 -> 246 -> 1997 -> 424 -> 3121 -> 1725 -> 1349 -> 1523 -> 1937 -> 1727
-> 4798 -> 3054 -> 2184 -> 2375 -> 1458 -> 3076 -> 4438 -> 3567 -> 4273 ->
2819 -> 1095 -> 2095 -> 4424 -> 2529 -> 994
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.588 seconds
      No. of Hops: 49
      Path: 174 -> 2696 -> 3988 -> 720 -> 1359 -> 3422 -> 2575 -> 1151 -> 368
-> 1141 -> 2650 -> 956 -> 3673 -> 1985 -> 4692 -> 3343 -> 983 -> 1785 -> 4641
-> 1123 -> 2584 -> 3073 -> 2323 -> 3097 -> 2418 -> 900 -> 4444 -> 3818 ->
2543 -> 285 -> 850 -> 374 -> 1977 -> 3372 -> 1453 -> 547 -> 3854 -> 3792 ->
2375 -> 1458 -> 3076 -> 4438 -> 3567 -> 4273 -> 2819 -> 1095 -> 2095 -> 4424
-> 2529 -> 994
      Maximum Bandwidth: 5992

Finding Maximum bandwidth path between source = 3378 and target = 2531.

      Found Maximum bandwidth path WITHOUT heap in 0.146 seconds
      No. of Hops: 21
      Path: 3378 -> 1474 -> 4831 -> 1573 -> 2472 -> 318 -> 2609 -> 1725 ->
2518 -> 3757 -> 2487 -> 457 -> 3171 -> 368 -> 1141 -> 2650 -> 2115 -> 790 ->
2060 -> 646 -> 683 -> 2531
      Maximum Bandwidth: 5991

      Found Maximum bandwidth path WITH heap in 0.08 seconds
      No. of Hops: 23
      Path: 3378 -> 1474 -> 3475 -> 2221 -> 4966 -> 4070 -> 2621 -> 4120 ->
3863 -> 4034 -> 4841 -> 3739 -> 1127 -> 2095 -> 1095 -> 2819 -> 195 -> 644 ->
2802 -> 3106 -> 719 -> 2391 -> 683 -> 2531
      Maximum Bandwidth: 5991

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.54 seconds
      No. of Hops: 54
      Path: 3378 -> 1474 -> 3475 -> 2221 -> 4966 -> 3591 -> 4597 -> 4243 ->
4540 -> 3896 -> 1473 -> 2990 -> 619 -> 2041 -> 1755 -> 3422 -> 2575 -> 1151 -
> 368 -> 1141 -> 2823 -> 3876 -> 2400 -> 2512 -> 1071 -> 1526 -> 4503 -> 1433
-> 2126 -> 4578 -> 3660 -> 1067 -> 598 -> 2076 -> 4731 -> 1732 -> 3619 ->
1714 -> 2266 -> 4516 -> 4875 -> 2660 -> 999 -> 2279 -> 1500 -> 2437 -> 445 ->
2863 -> 1583 -> 2163 -> 1051 -> 1541 -> 646 -> 683 -> 2531
      Maximum Bandwidth: 5991

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.1666
Average time taken by Dijkstra with heap: 0.1032
Average time taken by Kruskal's Algorithm: 2.5516
==============================================================================
====


p4_graph1
Finding Maximum bandwidth path between source = 2824 and target = 1146.

      Found Maximum bandwidth path WITHOUT heap in 0.096 seconds
      No. of Hops: 71
      Path: 2824 -> 4945 -> 3229 -> 2068 -> 47 -> 2212 -> 1053 -> 2274 ->
4696 -> 3294 -> 680 -> 539 -> 4943 -> 1475 -> 4669 -> 4501 -> 4899 -> 3877 ->
2695 -> 4897 -> 1542 -> 4698 -> 644 -> 4920 -> 2134 -> 1307 -> 980 -> 4843 ->
3715 -> 1615 -> 781 -> 4643 -> 2624 -> 4971 -> 2087 -> 3050 -> 2668 -> 321 ->
246 -> 3228 -> 3991 -> 4867 -> 701 -> 2617 -> 1828 -> 3769 -> 3622 -> 4303 ->
1215 -> 1999 -> 2592 -> 2838 -> 824 -> 3114 -> 1038 -> 3867 -> 65 -> 1373 ->
3947 -> 197 -> 3555 -> 1889 -> 84 -> 4299 -> 4183 -> 2059 -> 4162 -> 645 ->
365 -> 118 -> 543 -> 1146
      Maximum Bandwidth: 4527

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 71
      Path: 2824 -> 4945 -> 3229 -> 2068 -> 47 -> 2212 -> 1053 -> 2274 ->
4696 -> 3294 -> 680 -> 539 -> 4943 -> 1475 -> 4669 -> 4501 -> 4899 -> 3877 ->
2695 -> 4897 -> 1542 -> 4698 -> 644 -> 4920 -> 2134 -> 1307 -> 980 -> 4843 ->
3715 -> 1615 -> 781 -> 4643 -> 2624 -> 4971 -> 2087 -> 3050 -> 2668 -> 321 ->
246 -> 3228 -> 3991 -> 4867 -> 701 -> 2617 -> 1828 -> 3769 -> 3622 -> 4303 ->
1215 -> 1999 -> 2592 -> 2838 -> 824 -> 3114 -> 1038 -> 3867 -> 65 -> 1373 ->
3947 -> 197 -> 3555 -> 1889 -> 84 -> 4299 -> 4183 -> 2059 -> 4162 -> 645 ->
365 -> 118 -> 543 -> 1146
      Maximum Bandwidth: 4527

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.005 seconds
      No. of Hops: 71
      Path: 2824 -> 4945 -> 3229 -> 2068 -> 47 -> 2212 -> 1053 -> 2274 ->
4696 -> 3294 -> 680 -> 539 -> 4943 -> 1475 -> 4669 -> 4501 -> 4899 -> 3877 ->
2695 -> 4897 -> 1542 -> 4698 -> 644 -> 4920 -> 2134 -> 1307 -> 980 -> 4843 ->
3715 -> 1615 -> 781 -> 4643 -> 2624 -> 4971 -> 2087 -> 3050 -> 2668 -> 321 ->
246 -> 3228 -> 3991 -> 4867 -> 701 -> 2617 -> 1828 -> 3769 -> 3622 -> 4303 ->
```

```
1215 -> 1999 -> 2592 -> 2838 -> 824 -> 3114 -> 1038 -> 3867 -> 65 -> 1373 ->
3947 -> 197 -> 3555 -> 1889 -> 84 -> 4299 -> 4183 -> 2059 -> 4162 -> 645 ->
365 -> 118 -> 543 -> 1146
      Maximum Bandwidth: 4527

Finding Maximum bandwidth path between source = 4867 and target = 145.

      Found Maximum bandwidth path WITHOUT heap in 0.102 seconds
      No. of Hops: 10
      Path: 4867 -> 3991 -> 3228 -> 246 -> 1382 -> 2047 -> 3479 -> 4954 ->
1441 -> 3001 -> 145
      Maximum Bandwidth: 3408

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 10
      Path: 4867 -> 3991 -> 3228 -> 246 -> 1382 -> 2047 -> 3479 -> 4954 ->
1441 -> 3001 -> 145
      Maximum Bandwidth: 3408

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 10
      Path: 4867 -> 3991 -> 3228 -> 246 -> 1382 -> 2047 -> 3479 -> 4954 ->
1441 -> 3001 -> 145
      Maximum Bandwidth: 3408

Finding Maximum bandwidth path between source = 334 and target = 1606.

      Found Maximum bandwidth path WITHOUT heap in 0.095 seconds
      No. of Hops: 86
      Path: 334 -> 4278 -> 3054 -> 346 -> 3811 -> 870 -> 1180 -> 588 -> 2567
-> 3498 -> 3522 -> 3179 -> 2858 -> 1701 -> 4675 -> 2860 -> 4304 -> 4503 ->
734 -> 2816 -> 3038 -> 4079 -> 348 -> 680 -> 3294 -> 4696 -> 2274 -> 1053 ->
2212 -> 47 -> 2068 -> 3229 -> 3588 -> 2161 -> 4741 -> 308 -> 3502 -> 1838 ->
4055 -> 1676 -> 3631 -> 4125 -> 600 -> 1056 -> 2760 -> 1502 -> 648 -> 3726 ->
1548 -> 3395 -> 1482 -> 1333 -> 1081 -> 4776 -> 3284 -> 4883 -> 418 -> 4340 -
> 3032 -> 3768 -> 2305 -> 2106 -> 3802 -> 1599 -> 4744 -> 2117 -> 2790 -> 935
-> 2165 -> 2082 -> 3592 -> 856 -> 2484 -> 1194 -> 3211 -> 4752 -> 4669 ->
4501 -> 4899 -> 573 -> 2320 -> 2105 -> 33 -> 4628 -> 2983 -> 1227 -> 1606
      Maximum Bandwidth: 4470

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 37
      Path: 334 -> 4278 -> 3054 -> 346 -> 3811 -> 870 -> 1180 -> 588 -> 2567
-> 3498 -> 3522 -> 3179 -> 2858 -> 1701 -> 4675 -> 2860 -> 4304 -> 4503 ->
734 -> 2816 -> 3038 -> 4079 -> 348 -> 680 -> 539 -> 4943 -> 1475 -> 4669 ->
4501 -> 4899 -> 573 -> 2320 -> 2105 -> 33 -> 4628 -> 2983 -> 1227 -> 1606
      Maximum Bandwidth: 4470

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 37
      Path: 334 -> 4278 -> 3054 -> 346 -> 3811 -> 870 -> 1180 -> 588 -> 2567
-> 3498 -> 3522 -> 3179 -> 2858 -> 1701 -> 4675 -> 2860 -> 4304 -> 4503 ->
734 -> 2816 -> 3038 -> 4079 -> 348 -> 680 -> 539 -> 4943 -> 1475 -> 4669 ->
4501 -> 4899 -> 573 -> 2320 -> 2105 -> 33 -> 4628 -> 2983 -> 1227 -> 1606
      Maximum Bandwidth: 4470

Finding Maximum bandwidth path between source = 4177 and target = 2629.
```

```
      Found Maximum bandwidth path WITHOUT heap in 0.096 seconds
      No. of Hops: 37
      Path: 4177 -> 2768 -> 2999 -> 3131 -> 4851 -> 4265 -> 4729 -> 269 ->
279 -> 1489 -> 1026 -> 1431 -> 4403 -> 4640 -> 3855 -> 1215 -> 1999 -> 2592 -
> 2838 -> 824 -> 3114 -> 1038 -> 3867 -> 65 -> 1373 -> 2698 -> 3446 -> 3984 -
> 413 -> 12 -> 2807 -> 2722 -> 2856 -> 1484 -> 2503 -> 2897 -> 4017 -> 2629
      Maximum Bandwidth: 2614

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 42
      Path: 4177 -> 2768 -> 2999 -> 3131 -> 4851 -> 4265 -> 4729 -> 269 ->
279 -> 2238 -> 651 -> 1423 -> 2854 -> 1477 -> 4166 -> 4896 -> 448 -> 1369 ->
747 -> 4372 -> 1981 -> 1126 -> 1656 -> 1746 -> 1092 -> 4256 -> 1857 -> 3867 -
> 65 -> 1373 -> 2698 -> 3446 -> 3984 -> 413 -> 12 -> 2807 -> 2722 -> 2856 ->
1484 -> 2503 -> 2897 -> 4017 -> 2629
      Maximum Bandwidth: 2614

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.006 seconds
      No. of Hops: 85
      Path: 4177 -> 2768 -> 2999 -> 3131 -> 4851 -> 4265 -> 4729 -> 269 ->
279 -> 2238 -> 651 -> 1423 -> 2854 -> 1477 -> 4166 -> 4896 -> 448 -> 1369 ->
747 -> 4372 -> 1981 -> 1126 -> 1656 -> 1746 -> 1092 -> 4256 -> 1857 -> 3867 -
> 1038 -> 3114 -> 824 -> 2838 -> 2592 -> 1999 -> 1215 -> 4303 -> 3622 -> 3769
-> 1828 -> 2617 -> 701 -> 4867 -> 3991 -> 3228 -> 246 -> 321 -> 2668 -> 3050
-> 2087 -> 4971 -> 2624 -> 4643 -> 781 -> 1615 -> 3715 -> 4843 -> 980 -> 1307
-> 2134 -> 4920 -> 644 -> 4698 -> 1542 -> 4897 -> 2695 -> 3877 -> 4899 ->
4501 -> 4669 -> 1475 -> 4943 -> 539 -> 680 -> 348 -> 4079 -> 3038 -> 2816 ->
3344 -> 2664 -> 1171 -> 2856 -> 1484 -> 2503 -> 2897 -> 4017 -> 2629
      Maximum Bandwidth: 2614

Finding Maximum bandwidth path between source = 509 and target = 3567.

      Found Maximum bandwidth path WITHOUT heap in 0.098 seconds
      No. of Hops: 71
      Path: 509 -> 3393 -> 4411 -> 178 -> 1994 -> 526 -> 4748 -> 4624 -> 2196
-> 4888 -> 4696 -> 3294 -> 680 -> 348 -> 4079 -> 3038 -> 2816 -> 3344 -> 2664
-> 1171 -> 2856 -> 2722 -> 2807 -> 12 -> 413 -> 3984 -> 3446 -> 2698 -> 1373
-> 65 -> 3867 -> 1038 -> 3114 -> 824 -> 2838 -> 2592 -> 3478 -> 3160 -> 430 -
> 1461 -> 1851 -> 921 -> 4664 -> 4345 -> 3122 -> 2199 -> 2637 -> 274 -> 2236
-> 1248 -> 220 -> 647 -> 4125 -> 3631 -> 1676 -> 4055 -> 1838 -> 3502 -> 308
-> 3856 -> 2691 -> 482 -> 1930 -> 3797 -> 3006 -> 3522 -> 3498 -> 2567 -> 588
-> 1180 -> 870 -> 3567
      Maximum Bandwidth: 4735

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 31
      Path: 509 -> 3393 -> 4411 -> 178 -> 1994 -> 526 -> 4748 -> 4624 -> 2196
-> 4888 -> 4696 -> 3294 -> 680 -> 348 -> 4079 -> 3038 -> 2816 -> 734 -> 4503
-> 4304 -> 2860 -> 4675 -> 1701 -> 2858 -> 3179 -> 3522 -> 3498 -> 2567 ->
588 -> 1180 -> 870 -> 3567
      Maximum Bandwidth: 4735

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.005 seconds
      No. of Hops: 31
      Path: 509 -> 3393 -> 4411 -> 178 -> 1994 -> 526 -> 4748 -> 4624 -> 2196
-> 4888 -> 4696 -> 3294 -> 680 -> 348 -> 4079 -> 3038 -> 2816 -> 734 -> 4503
-> 4304 -> 2860 -> 4675 -> 1701 -> 2858 -> 3179 -> 3522 -> 3498 -> 2567 ->
588 -> 1180 -> 870 -> 3567
      Maximum Bandwidth: 4735

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.09740001
Average time taken by Dijkstra with heap: 0.003
Average time taken by Kruskal's Algorithm: 0.0048
==============================================================================
====


p4_graph2
Finding Maximum bandwidth path between source = 1120 and target = 1859.

      Found Maximum bandwidth path WITHOUT heap in 0.134 seconds
      No. of Hops: 68
      Path: 1120 -> 1038 -> 2794 -> 1313 -> 3297 -> 3476 -> 1927 -> 2480 ->
2998 -> 2581 -> 2446 -> 3287 -> 45 -> 484 -> 1800 -> 3465 -> 302 -> 2878 ->
154 -> 3057 -> 12 -> 2004 -> 3050 -> 1599 -> 1719 -> 3335 -> 1115 -> 3392 ->
2811 -> 2065 -> 1329 -> 1896 -> 2687 -> 769 -> 3415 -> 2143 -> 2219 -> 1318 -
> 2488 -> 1063 -> 853 -> 486 -> 1370 -> 744 -> 1213 -> 2750 -> 2985 -> 1325 -
> 1723 -> 1903 -> 2867 -> 2708 -> 267 -> 899 -> 1813 -> 1523 -> 466 -> 2421 -
> 83 -> 2575 -> 42 -> 1853 -> 2439 -> 253 -> 4168 -> 3381 -> 2669 -> 2665 ->
1859
      Maximum Bandwidth: 5990

      Found Maximum bandwidth path WITH heap in 0.07 seconds
      No. of Hops: 20
      Path: 1120 -> 1038 -> 4310 -> 4972 -> 3004 -> 4841 -> 2072 -> 1744 ->
2764 -> 545 -> 4047 -> 4231 -> 2805 -> 3724 -> 4855 -> 4101 -> 4168 -> 3381 -
> 2669 -> 2665 -> 1859
      Maximum Bandwidth: 5990

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.545 seconds
      No. of Hops: 79
      Path: 1120 -> 3694 -> 1644 -> 4041 -> 4310 -> 4972 -> 4809 -> 3423 ->
510 -> 1715 -> 155 -> 2976 -> 4363 -> 1982 -> 4813 -> 2962 -> 647 -> 408 ->
1130 -> 2844 -> 2071 -> 2625 -> 4318 -> 448 -> 4423 -> 3625 -> 3839 -> 64 ->
3955 -> 600 -> 69 -> 1581 -> 2620 -> 1475 -> 2183 -> 1571 -> 611 -> 1113 ->
3684 -> 1233 -> 1281 -> 893 -> 3792 -> 2199 -> 1886 -> 922 -> 3602 -> 3514 ->
2094 -> 3483 -> 699 -> 2383 -> 656 -> 664 -> 579 -> 4375 -> 946 -> 1868 ->
3940 -> 1663 -> 1021 -> 4680 -> 4108 -> 1670 -> 1556 -> 3177 -> 4670 -> 4208
-> 1009 -> 4969 -> 878 -> 3104 -> 3781 -> 230 -> 253 -> 4168 -> 3381 -> 2669
-> 2665 -> 1859
      Maximum Bandwidth: 5990

Finding Maximum bandwidth path between source = 476 and target = 610.

      Found Maximum bandwidth path WITHOUT heap in 0.187 seconds
      No. of Hops: 8
      Path: 476 -> 16 -> 2071 -> 2625 -> 4318 -> 448 -> 2519 -> 3223 -> 610
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path WITH heap in 0.128 seconds
      No. of Hops: 8
      Path: 476 -> 16 -> 2071 -> 2625 -> 4318 -> 448 -> 2519 -> 3223 -> 610
      Maximum Bandwidth: 5992

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.555 seconds
      No. of Hops: 8
      Path: 476 -> 16 -> 2071 -> 2625 -> 4318 -> 448 -> 2519 -> 3223 -> 610
      Maximum Bandwidth: 5992

Finding Maximum bandwidth path between source = 3077 and target = 1099.

      Found Maximum bandwidth path WITHOUT heap in 0.145 seconds
      No. of Hops: 52
      Path: 3077 -> 684 -> 4618 -> 89 -> 723 -> 742 -> 3666 -> 3009 -> 140 ->
3750 -> 2727 -> 1440 -> 3585 -> 1780 -> 1852 -> 1208 -> 1573 -> 2530 -> 1361
-> 712 -> 932 -> 1096 -> 232 -> 2887 -> 2575 -> 83 -> 2421 -> 466 -> 1523 ->
1813 -> 899 -> 267 -> 2708 -> 2867 -> 1903 -> 1723 -> 1325 -> 2985 -> 2750 ->
1213 -> 744 -> 1370 -> 486 -> 853 -> 1063 -> 2488 -> 1318 -> 2219 -> 2143 ->
3415 -> 769 -> 2485 -> 1099
      Maximum Bandwidth: 5990

      Found Maximum bandwidth path WITH heap in 0.086 seconds
      No. of Hops: 71
      Path: 3077 -> 684 -> 4618 -> 89 -> 723 -> 742 -> 3666 -> 6 -> 4177 ->
2411 -> 2999 -> 4313 -> 2190 -> 540 -> 3733 -> 3603 -> 2004 -> 12 -> 4636 ->
2925 -> 2606 -> 2039 -> 4336 -> 3470 -> 2351 -> 1754 -> 869 -> 441 -> 239 ->
2474 -> 4016 -> 815 -> 1086 -> 1047 -> 1828 -> 289 -> 1238 -> 1866 -> 3811 ->
800 -> 3792 -> 893 -> 3922 -> 3779 -> 2939 -> 1314 -> 1154 -> 4598 -> 599 ->
3533 -> 34 -> 1799 -> 2071 -> 1160 -> 936 -> 1871 -> 3854 -> 4549 -> 4375 ->
579 -> 664 -> 1335 -> 4697 -> 1098 -> 251 -> 1013 -> 1994 -> 3816 -> 4717 ->
2497 -> 968 -> 1099
      Maximum Bandwidth: 5990

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.505 seconds
      No. of Hops: 29
      Path: 3077 -> 684 -> 4618 -> 89 -> 723 -> 742 -> 3666 -> 3009 -> 714 ->
1495 -> 2909 -> 886 -> 3928 -> 3329 -> 3483 -> 2094 -> 3514 -> 3602 -> 4948 -
> 678 -> 393 -> 2580 -> 3264 -> 2582 -> 2219 -> 2143 -> 3415 -> 769 -> 2485 -
> 1099
      Maximum Bandwidth: 5990

Finding Maximum bandwidth path between source = 2995 and target = 938.

      Found Maximum bandwidth path WITHOUT heap in 0.231 seconds
      No. of Hops: 46
      Path: 2995 -> 2666 -> 3988 -> 1400 -> 2801 -> 679 -> 3464 -> 633 ->
3635 -> 2664 -> 351 -> 357 -> 1386 -> 1962 -> 2112 -> 4499 -> 3231 -> 1554 ->
3971 -> 4802 -> 505 -> 959 -> 1503 -> 4731 -> 1728 -> 3525 -> 1016 -> 3420 ->
4259 -> 2936 -> 3535 -> 658 -> 704 -> 2207 -> 2051 -> 852 -> 1146 -> 3081 ->
990 -> 3039 -> 797 -> 3426 -> 433 -> 142 -> 2578 -> 264 -> 938
      Maximum Bandwidth: 5983

      Found Maximum bandwidth path WITH heap in 0.157 seconds
      No. of Hops: 46
      Path: 2995 -> 2666 -> 3988 -> 1400 -> 2801 -> 679 -> 3464 -> 633 ->
3635 -> 2664 -> 351 -> 357 -> 1386 -> 1962 -> 2112 -> 4499 -> 3231 -> 1554 ->
3971 -> 4802 -> 505 -> 959 -> 1503 -> 4731 -> 1728 -> 3525 -> 1016 -> 3420 ->
4259 -> 2936 -> 3535 -> 658 -> 704 -> 2207 -> 2051 -> 852 -> 1146 -> 3081 ->
990 -> 3039 -> 797 -> 3426 -> 433 -> 142 -> 2578 -> 264 -> 938
```

```
      Maximum Bandwidth: 5983

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.548 seconds
      No. of Hops: 46
      Path: 2995 -> 2666 -> 3988 -> 1400 -> 2801 -> 679 -> 3464 -> 633 ->
3635 -> 2664 -> 351 -> 357 -> 1386 -> 1962 -> 2112 -> 4499 -> 3231 -> 1554 ->
3971 -> 4802 -> 505 -> 959 -> 1503 -> 4731 -> 1728 -> 3525 -> 1016 -> 3420 ->
4259 -> 2936 -> 3535 -> 658 -> 704 -> 2207 -> 2051 -> 852 -> 1146 -> 3081 ->
990 -> 3039 -> 797 -> 3426 -> 433 -> 142 -> 2578 -> 264 -> 938
      Maximum Bandwidth: 5983

Finding Maximum bandwidth path between source = 2355 and target = 2850.

      Found Maximum bandwidth path WITHOUT heap in 0.191 seconds
      No. of Hops: 36
      Path: 2355 -> 1890 -> 437 -> 1673 -> 4686 -> 3036 -> 2105 -> 4458 ->
2061 -> 2995 -> 2666 -> 3898 -> 2710 -> 1769 -> 1995 -> 3364 -> 3519 -> 2887
-> 232 -> 1096 -> 156 -> 4073 -> 4209 -> 1805 -> 4054 -> 1970 -> 3689 -> 4229
-> 3908 -> 1154 -> 1314 -> 2939 -> 3779 -> 3922 -> 1842 -> 91 -> 2850
      Maximum Bandwidth: 5983

      Found Maximum bandwidth path WITH heap in 0.125 seconds
      No. of Hops: 40
      Path: 2355 -> 1890 -> 437 -> 1673 -> 4686 -> 3036 -> 2105 -> 4458 ->
2061 -> 2995 -> 2666 -> 3988 -> 2489 -> 2449 -> 4102 -> 305 -> 3715 -> 4096 -
> 2695 -> 4802 -> 3971 -> 1554 -> 3231 -> 4499 -> 4463 -> 1852 -> 1208 ->
2976 -> 155 -> 2396 -> 4818 -> 2285 -> 3908 -> 1154 -> 1314 -> 2939 -> 3779 -
> 3922 -> 1842 -> 91 -> 2850
      Maximum Bandwidth: 5983

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.548 seconds
      No. of Hops: 57
      Path: 2355 -> 1890 -> 437 -> 1673 -> 4686 -> 3036 -> 2105 -> 4458 ->
2061 -> 2995 -> 2666 -> 3988 -> 1400 -> 2801 -> 679 -> 3464 -> 633 -> 3635 ->
2664 -> 351 -> 357 -> 1386 -> 1962 -> 2112 -> 4499 -> 4532 -> 3104 -> 3781 ->
230 -> 253 -> 4168 -> 4101 -> 1151 -> 4352 -> 3820 -> 327 -> 4970 -> 137 ->
2199 -> 1886 -> 922 -> 3602 -> 4948 -> 678 -> 393 -> 2580 -> 2659 -> 4778 ->
2015 -> 4598 -> 1154 -> 1314 -> 2939 -> 3779 -> 3922 -> 1842 -> 91 -> 2850
      Maximum Bandwidth: 5983

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.1776
Average time taken by Dijkstra with heap: 0.1132
Average time taken by Kruskal's Algorithm: 2.5402002
============================================================================
====


p5_graph1
Finding Maximum bandwidth path between source = 2786 and target = 4278.

      Found Maximum bandwidth path WITHOUT heap in 0.087 seconds
      No. of Hops: 29
      Path: 2786 -> 2323 -> 3051 -> 919 -> 1636 -> 3190 -> 916 -> 3134 ->
2080 -> 803 -> 373 -> 2708 -> 2648 -> 1703 -> 20 -> 3018 -> 2551 -> 379 ->
2269 -> 40 -> 1504 -> 363 -> 118 -> 1215 -> 2469 -> 1121 -> 2806 -> 2860 ->
3930 -> 4278
```

```
      Maximum Bandwidth: 4338

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 37
      Path: 2786 -> 2323 -> 3051 -> 919 -> 1636 -> 3190 -> 916 -> 3134 ->
2080 -> 803 -> 373 -> 2708 -> 3525 -> 3309 -> 1851 -> 55 -> 1568 -> 324 ->
2955 -> 4745 -> 4929 -> 2394 -> 3387 -> 1322 -> 1411 -> 4203 -> 2272 -> 2104
-> 207 -> 3281 -> 2943 -> 1386 -> 2068 -> 1121 -> 2806 -> 2860 -> 3930 ->
4278
      Maximum Bandwidth: 4338

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 67
      Path: 2786 -> 2323 -> 3051 -> 919 -> 1636 -> 3190 -> 916 -> 3134 ->
2080 -> 803 -> 373 -> 2708 -> 2648 -> 274 -> 4061 -> 2405 -> 3263 -> 284 ->
2213 -> 3083 -> 3232 -> 3436 -> 1797 -> 4600 -> 2158 -> 1999 -> 445 -> 2276 -
> 4738 -> 808 -> 2128 -> 2054 -> 94 -> 552 -> 4981 -> 769 -> 181 -> 3767 ->
3851 -> 1074 -> 1855 -> 896 -> 4187 -> 4411 -> 2613 -> 3615 -> 4819 -> 3718 -
> 4096 -> 4619 -> 1930 -> 3313 -> 899 -> 1064 -> 2304 -> 4243 -> 2962 -> 1695
-> 4686 -> 826 -> 2943 -> 1386 -> 2068 -> 1121 -> 2806 -> 2860 -> 3930 ->
4278
      Maximum Bandwidth: 4338

Finding Maximum bandwidth path between source = 4632 and target = 4454.

      Found Maximum bandwidth path WITHOUT heap in 0.098 seconds
      No. of Hops: 15
      Path: 4632 -> 4202 -> 2672 -> 194 -> 286 -> 1404 -> 1956 -> 193 -> 3454
-> 3180 -> 4075 -> 4092 -> 3926 -> 1343 -> 4020 -> 4454
      Maximum Bandwidth: 4603

      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 15
      Path: 4632 -> 4202 -> 2672 -> 194 -> 286 -> 615 -> 2152 -> 334 -> 94 ->
1287 -> 39 -> 4851 -> 3709 -> 3112 -> 1787 -> 4454
      Maximum Bandwidth: 4603

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 74
      Path: 4632 -> 4202 -> 2672 -> 2540 -> 2070 -> 803 -> 373 -> 2708 ->
2648 -> 274 -> 4061 -> 2405 -> 3263 -> 284 -> 2213 -> 3083 -> 3232 -> 3436 ->
1797 -> 4600 -> 2158 -> 1999 -> 445 -> 2276 -> 4738 -> 808 -> 2128 -> 2054 ->
94 -> 552 -> 4981 -> 769 -> 181 -> 3767 -> 3851 -> 1074 -> 1855 -> 896 ->
4187 -> 4411 -> 2613 -> 3615 -> 4819 -> 3718 -> 4096 -> 4619 -> 1930 -> 3313
-> 899 -> 4006 -> 4046 -> 1925 -> 878 -> 1018 -> 2418 -> 4731 -> 3475 -> 2136
-> 1732 -> 4464 -> 3628 -> 3437 -> 140 -> 4117 -> 4320 -> 1973 -> 1368 ->
2714 -> 4105 -> 1924 -> 2297 -> 1413 -> 3112 -> 1787 -> 4454
      Maximum Bandwidth: 4603

Finding Maximum bandwidth path between source = 2126 and target = 1254.

      Found Maximum bandwidth path WITHOUT heap in 0.089 seconds
      No. of Hops: 16
      Path: 2126 -> 3835 -> 2695 -> 2070 -> 2540 -> 2672 -> 194 -> 286 ->
1404 -> 510 -> 1247 -> 4397 -> 650 -> 3030 -> 603 -> 125 -> 1254
      Maximum Bandwidth: 3868
      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 27
      Path: 2126 -> 3835 -> 2695 -> 2906 -> 2163 -> 371 -> 3337 -> 599 -> 639
-> 260 -> 4985 -> 1344 -> 2846 -> 866 -> 3497 -> 278 -> 3546 -> 253 -> 2044 -
> 3382 -> 4398 -> 1247 -> 4397 -> 650 -> 3030 -> 603 -> 125 -> 1254
      Maximum Bandwidth: 3868

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 42
      Path: 2126 -> 3835 -> 2695 -> 2906 -> 2163 -> 4872 -> 1855 -> 896 ->
4187 -> 4411 -> 2613 -> 3615 -> 4819 -> 3718 -> 4096 -> 4619 -> 1930 -> 3313
-> 899 -> 1064 -> 2304 -> 4243 -> 2962 -> 1695 -> 4686 -> 934 -> 3692 -> 74 -
> 4361 -> 2618 -> 3943 -> 3546 -> 253 -> 2044 -> 3382 -> 4398 -> 1247 -> 4397
-> 650 -> 3030 -> 603 -> 125 -> 1254
      Maximum Bandwidth: 3868

Finding Maximum bandwidth path between source = 1134 and target = 2605.

      Found Maximum bandwidth path WITHOUT heap in 0.088 seconds
      No. of Hops: 79
      Path: 1134 -> 4206 -> 358 -> 699 -> 2881 -> 2572 -> 1424 -> 2182 -> 437
-> 361 -> 2582 -> 3335 -> 3197 -> 2545 -> 2252 -> 283 -> 3972 -> 3436 -> 3232
-> 3083 -> 3093 -> 1014 -> 2962 -> 3362 -> 2559 -> 2166 -> 490 -> 1433 -> 678
-> 713 -> 335 -> 1209 -> 732 -> 1604 -> 477 -> 2310 -> 471 -> 1758 -> 1885 ->
2714 -> 1368 -> 1973 -> 2130 -> 611 -> 2860 -> 2806 -> 1121 -> 2469 -> 1215 -
> 118 -> 363 -> 1504 -> 40 -> 2269 -> 379 -> 2551 -> 3018 -> 20 -> 1703 ->
2648 -> 2708 -> 373 -> 803 -> 2070 -> 2540 -> 2672 -> 194 -> 286 -> 1404 ->
1956 -> 193 -> 3454 -> 3180 -> 1834 -> 2502 -> 513 -> 2446 -> 1765 -> 2011 ->
2605
      Maximum Bandwidth: 4200

      Found Maximum bandwidth path WITH heap in 0.005 seconds
      No. of Hops: 30
      Path: 1134 -> 4206 -> 358 -> 699 -> 2881 -> 2572 -> 1424 -> 2182 -> 437
-> 361 -> 2582 -> 4458 -> 1628 -> 105 -> 3693 -> 1932 -> 1010 -> 2467 -> 4817
-> 3030 -> 650 -> 4397 -> 1247 -> 510 -> 3106 -> 3514 -> 513 -> 2446 -> 1765
-> 2011 -> 2605
      Maximum Bandwidth: 4200

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.005 seconds
      No. of Hops: 56
      Path: 1134 -> 4206 -> 358 -> 699 -> 2881 -> 2572 -> 4124 -> 1521 -> 695
-> 213 -> 1192 -> 1802 -> 1667 -> 4544 -> 498 -> 1927 -> 3399 -> 4786 -> 4292
-> 347 -> 3473 -> 2031 -> 4011 -> 395 -> 1897 -> 536 -> 1441 -> 3986 -> 2322
-> 637 -> 1429 -> 3564 -> 655 -> 4133 -> 4195 -> 1274 -> 1971 -> 4735 -> 2901
-> 4851 -> 3709 -> 3112 -> 1787 -> 4454 -> 4020 -> 1343 -> 3926 -> 4092 ->
4075 -> 3180 -> 1834 -> 2502 -> 513 -> 2446 -> 1765 -> 2011 -> 2605
      Maximum Bandwidth: 4200

Finding Maximum bandwidth path between source = 2566 and target = 3951.

      Found Maximum bandwidth path WITHOUT heap in 0.102 seconds
      No. of Hops: 23
      Path: 2566 -> 245 -> 1771 -> 3948 -> 1566 -> 414 -> 819 -> 2650 -> 1452
-> 2526 -> 3747 -> 1180 -> 4811 -> 4381 -> 4248 -> 2687 -> 2876 -> 2701 ->
3936 -> 3708 -> 679 -> 589 -> 1529 -> 3951
      Maximum Bandwidth: 4429
      Found Maximum bandwidth path WITH heap in 0.003 seconds
      No. of Hops: 23
      Path: 2566 -> 245 -> 1771 -> 3948 -> 1566 -> 414 -> 819 -> 2650 -> 1452
-> 2526 -> 3747 -> 1180 -> 4811 -> 4381 -> 4248 -> 2687 -> 2876 -> 2701 ->
3936 -> 3708 -> 679 -> 589 -> 1529 -> 3951
      Maximum Bandwidth: 4429

      Found Maximum bandwidth path using Kruskal's Algorithm in 0.004 seconds
      No. of Hops: 73
      Path: 2566 -> 245 -> 1771 -> 3948 -> 1566 -> 414 -> 819 -> 2650 -> 1452
-> 2526 -> 3747 -> 1180 -> 4811 -> 4723 -> 2525 -> 260 -> 639 -> 599 -> 3337
-> 371 -> 3396 -> 3304 -> 4893 -> 1499 -> 476 -> 1622 -> 2938 -> 4388 -> 1110
-> 4600 -> 2158 -> 1999 -> 445 -> 2276 -> 4738 -> 808 -> 2128 -> 2054 -> 94 -
> 552 -> 4981 -> 769 -> 181 -> 3767 -> 3851 -> 1074 -> 1855 -> 896 -> 4187 ->
4411 -> 2613 -> 3615 -> 4819 -> 3718 -> 4096 -> 4619 -> 1930 -> 3313 -> 899 -
> 1064 -> 2304 -> 957 -> 3577 -> 1655 -> 1970 -> 3707 -> 3513 -> 1550 -> 380
-> 1330 -> 591 -> 4118 -> 1529 -> 3951
      Maximum Bandwidth: 4429

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.092800006
Average time taken by Dijkstra with heap: 0.0033999998
Average time taken by Kruskal's Algorithm: 0.0042000003
==============================================================================
====


p5_graph2
Finding Maximum bandwidth path between source = 2581 and target = 1603.

      Found Maximum bandwidth path WITHOUT heap in 0.147 seconds
      No. of Hops: 29
      Path: 2581 -> 3325 -> 3898 -> 2232 -> 120 -> 2395 -> 4159 -> 3553 ->
487 -> 309 -> 1875 -> 1285 -> 164 -> 3966 -> 988 -> 363 -> 2292 -> 3099 ->
882 -> 2878 -> 521 -> 1313 -> 3908 -> 3000 -> 3596 -> 646 -> 2679 -> 3993 ->
3830 -> 1603
      Maximum Bandwidth: 5993

      Found Maximum bandwidth path WITH heap in 0.094 seconds
      No. of Hops: 34
      Path: 2581 -> 3325 -> 3898 -> 2232 -> 120 -> 2395 -> 4159 -> 3553 ->
487 -> 309 -> 1875 -> 1285 -> 164 -> 3966 -> 988 -> 363 -> 2292 -> 1109 ->
4113 -> 2883 -> 4350 -> 1505 -> 2788 -> 1499 -> 1111 -> 3771 -> 248 -> 1449 -
> 1132 -> 2682 -> 4315 -> 481 -> 2425 -> 4793 -> 1603
      Maximum Bandwidth: 5993

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.545 seconds
      No. of Hops: 79
      Path: 2581 -> 3325 -> 3898 -> 2232 -> 120 -> 2395 -> 4159 -> 2608 ->
1328 -> 1850 -> 350 -> 369 -> 875 -> 3337 -> 1523 -> 1628 -> 2782 -> 379 ->
429 -> 211 -> 1652 -> 4164 -> 3612 -> 4002 -> 4420 -> 2342 -> 1321 -> 4996 ->
2819 -> 2747 -> 3063 -> 2704 -> 499 -> 791 -> 4337 -> 2463 -> 602 -> 760 ->
1071 -> 4900 -> 4616 -> 2844 -> 4122 -> 2007 -> 4094 -> 935 -> 2587 -> 1485 -
> 3052 -> 1760 -> 3897 -> 1963 -> 4721 -> 686 -> 2944 -> 402 -> 934 -> 1673 -
> 1286 -> 1166 -> 2848 -> 430 -> 2810 -> 3802 -> 358 -> 2513 -> 1501 -> 4415
```

```
-> 4145 -> 2494 -> 3142 -> 3914 -> 4237 -> 2206 -> 2682 -> 4315 -> 481 ->
2425 -> 4793 -> 1603
      Maximum Bandwidth: 5993

Finding Maximum bandwidth path between source = 3012 and target = 1169.

      Found Maximum bandwidth path WITHOUT heap in 0.233 seconds
      No. of Hops: 52
      Path: 3012 -> 4889 -> 3690 -> 3450 -> 197 -> 26 -> 616 -> 1230 -> 2410
-> 2009 -> 71 -> 796 -> 0 -> 704 -> 984 -> 369 -> 350 -> 1850 -> 344 -> 952 -
> 1092 -> 1752 -> 2925 -> 294 -> 1002 -> 119 -> 68 -> 3103 -> 586 -> 2538 ->
93 -> 2617 -> 605 -> 3140 -> 21 -> 3232 -> 2374 -> 1438 -> 2174 -> 2604 ->
2587 -> 1485 -> 917 -> 4178 -> 823 -> 415 -> 2254 -> 3560 -> 4224 -> 2456 ->
438 -> 3973 -> 1169
      Maximum Bandwidth: 5991

      Found Maximum bandwidth path WITH heap in 0.169 seconds
      No. of Hops: 91
      Path: 3012 -> 4889 -> 3690 -> 3450 -> 197 -> 26 -> 616 -> 1230 -> 2410
-> 2009 -> 71 -> 796 -> 3530 -> 4778 -> 2505 -> 1042 -> 997 -> 4799 -> 782 ->
2640 -> 2878 -> 4736 -> 461 -> 1016 -> 242 -> 1236 -> 1973 -> 2383 -> 3388 ->
3608 -> 2395 -> 4745 -> 4777 -> 3706 -> 156 -> 4095 -> 4219 -> 3796 -> 4300 -
> 2093 -> 1935 -> 2397 -> 1706 -> 2018 -> 3070 -> 2273 -> 630 -> 684 -> 3492
-> 2102 -> 3294 -> 4209 -> 3350 -> 3856 -> 292 -> 1903 -> 4909 -> 3688 ->
3396 -> 253 -> 2501 -> 1904 -> 61 -> 4580 -> 4718 -> 3913 -> 2313 -> 2104 ->
874 -> 3104 -> 974 -> 3949 -> 2198 -> 2947 -> 2683 -> 533 -> 2957 -> 4781 ->
2906 -> 1418 -> 2976 -> 2764 -> 4552 -> 191 -> 372 -> 3745 -> 3959 -> 4434 ->
3506 -> 4901 -> 3973 -> 1169
      Maximum Bandwidth: 5991

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.566 seconds
      No. of Hops: 45
      Path: 3012 -> 4889 -> 3690 -> 3450 -> 197 -> 26 -> 616 -> 1230 -> 2410
-> 2009 -> 71 -> 796 -> 0 -> 114 -> 391 -> 4546 -> 2504 -> 4569 -> 984 -> 369
-> 875 -> 3337 -> 1523 -> 1706 -> 2397 -> 2174 -> 2604 -> 2587 -> 935 -> 4094
-> 2007 -> 4122 -> 2844 -> 1928 -> 2191 -> 270 -> 2218 -> 4939 -> 2425 -> 481
-> 4315 -> 2682 -> 2300 -> 4911 -> 3973 -> 1169
      Maximum Bandwidth: 5991

Finding Maximum bandwidth path between source = 2581 and target = 1134.

      Found Maximum bandwidth path WITHOUT heap in 0.211 seconds
      No. of Hops: 55
      Path: 2581 -> 3325 -> 3898 -> 2232 -> 120 -> 2395 -> 4159 -> 3553 ->
487 -> 309 -> 1875 -> 1285 -> 1229 -> 2810 -> 430 -> 2848 -> 1166 -> 1286 ->
1673 -> 934 -> 1422 -> 910 -> 3245 -> 3749 -> 2532 -> 1585 -> 2455 -> 153 ->
734 -> 1670 -> 3877 -> 1604 -> 1132 -> 2682 -> 2300 -> 3639 -> 630 -> 684 ->
3492 -> 2102 -> 3674 -> 3488 -> 3029 -> 3234 -> 484 -> 2757 -> 3801 -> 3334 -
> 4203 -> 3151 -> 417 -> 1319 -> 1909 -> 534 -> 2005 -> 1134
      Maximum Bandwidth: 5989

      Found Maximum bandwidth path WITH heap in 0.133 seconds
      No. of Hops: 30
      Path: 2581 -> 3325 -> 3898 -> 2232 -> 813 -> 4808 -> 3714 -> 1747 ->
3254 -> 532 -> 1059 -> 2204 -> 2604 -> 2587 -> 1485 -> 917 -> 4178 -> 823 ->
415 -> 2254 -> 3560 -> 4224 -> 2456 -> 438 -> 3973 -> 4901 -> 1319 -> 1909 ->
534 -> 2005 -> 1134
```

```
      Maximum Bandwidth: 5989

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.529 seconds
      No. of Hops: 63
      Path: 2581 -> 3325 -> 3898 -> 2232 -> 120 -> 2395 -> 4159 -> 2608 ->
1328 -> 1850 -> 350 -> 369 -> 984 -> 4569 -> 2504 -> 375 -> 4026 -> 354 ->
4955 -> 1947 -> 201 -> 4122 -> 2007 -> 4094 -> 935 -> 2587 -> 1485 -> 3052 ->
1760 -> 3897 -> 1963 -> 4721 -> 686 -> 2944 -> 402 -> 934 -> 1673 -> 1286 ->
1166 -> 2848 -> 430 -> 2810 -> 1229 -> 2124 -> 3471 -> 4135 -> 1501 -> 4415 -
> 4145 -> 2494 -> 3142 -> 3914 -> 4237 -> 2206 -> 2682 -> 2300 -> 4911 ->
3973 -> 4901 -> 1319 -> 1909 -> 534 -> 2005 -> 1134
      Maximum Bandwidth: 5989

Finding Maximum bandwidth path between source = 726 and target = 3866.

      Found Maximum bandwidth path WITHOUT heap in 0.169 seconds
      No. of Hops: 25
      Path: 726 -> 1716 -> 4064 -> 1178 -> 788 -> 3505 -> 1286 -> 1166 ->
2848 -> 430 -> 2810 -> 1229 -> 4219 -> 4095 -> 4938 -> 832 -> 3485 -> 4684 ->
4983 -> 3812 -> 2736 -> 1011 -> 2166 -> 732 -> 621 -> 3866
      Maximum Bandwidth: 5993

      Found Maximum bandwidth path WITH heap in 0.098 seconds
      No. of Hops: 25
      Path: 726 -> 1716 -> 4064 -> 1178 -> 788 -> 3505 -> 1286 -> 1166 ->
2848 -> 430 -> 2810 -> 1229 -> 4219 -> 4095 -> 4938 -> 832 -> 3485 -> 4684 ->
4983 -> 3812 -> 2736 -> 1011 -> 2166 -> 732 -> 621 -> 3866
      Maximum Bandwidth: 5993

      Found Maximum bandwidth path using Kruskal's Algorithm in 2.55 seconds
      No. of Hops: 59
      Path: 726 -> 1716 -> 4064 -> 1178 -> 788 -> 3505 -> 1286 -> 1673 -> 934
-> 402 -> 2944 -> 686 -> 4721 -> 1963 -> 3897 -> 1760 -> 3052 -> 1485 -> 2587
-> 935 -> 4094 -> 2007 -> 4122 -> 2844 -> 1928 -> 2191 -> 270 -> 2218 -> 4939
-> 2425 -> 481 -> 4315 -> 2682 -> 2206 -> 4237 -> 3914 -> 3142 -> 2494 ->
4145 -> 4415 -> 1501 -> 2513 -> 358 -> 3802 -> 2810 -> 1229 -> 4219 -> 4095 -
> 4938 -> 832 -> 3485 -> 4684 -> 4983 -> 3812 -> 2736 -> 1011 -> 2166 -> 732
-> 621 -> 3866
      Maximum Bandwidth: 5993

Finding Maximum bandwidth path between source = 978 and target = 359.

      Found Maximum bandwidth path WITHOUT heap in 0.211 seconds
      No. of Hops: 30
      Path: 978 -> 3202 -> 4370 -> 867 -> 1436 -> 1389 -> 1338 -> 2256 -> 951
-> 2221 -> 1267 -> 2472 -> 1608 -> 22 -> 1437 -> 684 -> 1593 -> 1401 -> 2024
-> 938 -> 2537 -> 1127 -> 1851 -> 1024 -> 778 -> 471 -> 11 -> 1702 -> 3799 ->
1089 -> 359
      Maximum Bandwidth: 5987

      Found Maximum bandwidth path WITH heap in 0.176 seconds
      No. of Hops: 26
      Path: 978 -> 3202 -> 4370 -> 867 -> 4848 -> 3612 -> 1701 -> 760 -> 602
-> 2463 -> 3479 -> 484 -> 3234 -> 4545 -> 859 -> 1149 -> 1778 -> 2384 -> 3572
-> 1397 -> 4073 -> 3645 -> 1266 -> 4098 -> 3777 -> 4117 -> 359
      Maximum Bandwidth: 5987
      Found Maximum bandwidth path using Kruskal's Algorithm in 2.634 seconds
      No. of Hops: 10
      Path: 978 -> 3202 -> 4370 -> 3815 -> 329 -> 1981 -> 2482 -> 2724 ->
2407 -> 4487 -> 359
      Maximum Bandwidth: 5987

EXECUTION SUMMARY:
Average time taken by Dijkstra with no heap: 0.1942
Average time taken by Dijkstra with heap: 0.134
Average time taken by Kruskal's Algorithm: 2.5648003
=============================================================================
====


OVERALL SUMMARY:
=============================================================================
====
Average time taken by Dijkstra with no heap for sparse graph: 0.09608
Average time taken by Dijkstra with no heap for dense graph: 0.18072

Average time taken by Dijkstra with heap for sparse graph: 0.006
Average time taken by Dijkstra with heap for dense graph: 0.115959994

Average time taken by Kruskal's Algorithm for sparse graph: 0.00836
Average time taken by Kruskal's Algorithm for dense graph: 2.5963602
=============================================================================
====
```